Digital Signal Processing

Third Edition

About the Author

S Salivahanan is the Principal of SSN College of Engineering, Chennai. He obtained his BE in Electronics and Communication Engineering from PSG College of Technology, Coimbatore; ME in Communication Systems from NIT, Tiruchirappalli; and PhD in the area of Microwave Integrated Circuits from Madurai Kamaraj University. He has three and a half decades of teaching, research, administration and industrial experience both in India and abroad. He has also taught at NIT, Tiruchirappalli; AC College of Engineering and Technology, Karaikudi; RV College of Engineering, Bangalore; Dayananda Sagar College of Engineering, Bangalore; Mepco Schlenk Engineering College, Sivakasi; and Bannari Amman Institute of Technology, Sathyamangalam. He served as a mentor for the MS degree under the distance learning programme offered by Birla Institute of Technology and Science (BITS), Pilani.

Dr Salivahanan has industrial experience as Scientist/Engineer at Space Applications Centre, ISRO, Ahmedabad; Telecommunication Engineer at State Organisation of Electricity, Iraq; and Electronics Engineer at Electric Dar Establishment, Kingdom of Saudi Arabia. He is the author of 28 popular books which include all time best sellers such as *Basic Electrical, Electronics and Computer Engineering, Electronic Devices and Circuits*, and *Linear Integrated Circuits,* all published by McGraw Hill Education (India). *Digital Signal Processing* by McGraw Hill Education (India) and McGraw Hill Education International has also been translated into Mandarin, the largest spoken variant of the Chinese language. He has also authored *Digital Circuits and Design* and also published several papers at national and international levels.

Professor Salivahanan is the recipient of the Bharatiya Vidya Bhavan National Award for Best Engineering College Principal for 2011 from ISTE, and the IEEE Outstanding Branch Counsellor and Advisor Award in the Asia-Pacific region for 1996-97. He was the Chairman of IEEE Madras Section for two years (2008, 2009) and Syndicate Member of Anna University.

He is a Senior Member of IEEE, Fellow of IETE, Fellow of Institution of Engineers (India), Life Member of ISTE and Life Member of Society for EMC Engineers. He is also a member of IEEE Societies in Microwave Theory and Techniques, Communications, Signal Processing, and Aerospace and Electronics.

Digital Signal Processing

Third Edition

S Salivahanan

Principal
SSN College of Engineering
Chennai

McGraw Hill Education (India) Private Limited
NEW DELHI

McGraw Hill Education Offices
New Delhi New York St Louis San Francisco Auckland Bogotá Caracas
Kuala Lumpur Lisbon London Madrid Mexico City Milan Montreal
San Juan Santiago Singapore Sydney Tokyo Toronto

McGraw Hill Education (India) Private Limited

Published by McGraw Hill Education (India) Private Limited
P-24, Green Park Extension, New Delhi 110016

Digital Signal Processing, 3e

Print Edition
ISBN (13) : 978-9-33-290283-1
ISBN (10) : 9-33-290283-6

Ebook Edition
ISBN (13) : 978-9-33290284-8
ISBN (10) : 9-33-290284-4

Managing Director: *Kaushik Bellani*

Head—Higher Education Publishing and Marketing: *Vibha Mahajan*
Senior Publishing Manager—SEM & Tech Ed: *Shalini Jha*
Editorial Executive: *Koyel Ghosh*
Manager—Production Systems: *Satinder S Baveja*
Assistant Manager—Editorial Services: *Sohini Mukherjee*
Assistant Manager—Production: *Anjali Razdan*

Assistant General Manager: Marketing—Higher Education: *Vijay Sarathi*
Asst. Product Manager—SEM & Tech Ed: *Tina Jajoriya*
Senior Graphic Designer (Cover): *Meenu Raghav*

General Manager—Production: *Rajender P Ghansela*
Manager—Production: *Reji Kumar*

Typeset at Print-O-World, 2579, Mandir Lane, Shadipur, New Delhi 110 008, and printed at

Cover Printer :

R

Foreword

It is my pleasant privilege to write a foreword for *Digital Signal Processing* authored by Prof. S Salivahanan, Principal, SSN College of Engineering, Kalavakkam, Chennai. The author has used his vast experience in preparing the manuscript to enable the reader to understand the concepts of digital signal processing in the simplest manner. There has been an increase in the demand for a suitable textbook on analog and digital signal processing technologies. The contents of the book are well organized and written in simple language with numerous worked-out examples and exercise problems with answers. The level of presentation is suitable for self-study. MATLAB-based programs for solving digital signal processing problems are the need of the day and this book satisfies the requirement completely.

Having been a teacher in the field of Electronics and Communication Engineering for the past four decades, I believe that this book will serve as a useful text for the subjects of Signals and Systems as well as Digital Signal Processing in the undergraduate courses and as a ready reference for the Advanced Signal Processing course at the postgraduate level.

I strongly recommend this book to every Electronics, Electrical, Computer Science, Information Technology, Communication Systems and Instrumentation Engineering student and all research engineers. A few copies of this book in the libraries of all science and engineering colleges and polytechnics will be found quite useful.

V Palanisamy
Former Principal
Government College of Engineering
Salem, Tamil Nadu

Contents

Preface

Research and development over the past four decades has led to significant advances in the field of Digital Signal Processing. A single textbook covering the basics of linear continuous-time and discrete-time systems and their applications along with MATLAB programs is the need of the day. The author has made a sincere attempt to meet these requirements.

Target Audience

The author has developed this book from his lecture notes prepared for teaching the undergraduate and postgraduate-level courses over the past several years. This book is suitable as a text for the subjects of Signals and Systems, and Digital Signal Processing in BE, AMIE and Grade IETE degree programs, and for the subject of Advanced Digital Signal Processing in the ME degree program. It will also serve as a useful reference to those preparing for competitive examinations.

About the Book

The various concepts of the subject are arranged logically and explained in a simple reader-friendly language. For proper understanding of the subject, a large number of problems with their step-by-step solutions are provided for every concept. Illustrative examples are discussed to emphasize the conceptual clarity, thereby presenting typical applications. Solutions to university problems have been included in many chapters in this edition. A set of questions and exercises at the end of each chapter will help readers test their understanding of the subject.

Salient Features

- Digital Signal Processing presented with an application-based approach
- Dedicated chapters on Digital Signal Processors, Discrete Time Random Signal Processing, and Power Spectral Estimation
- Expanded and in-depth coverage on Filters
- Modification of topics according to latest updates with addition of Hilbert Transformer, Invertible Systems, and Comparative tables
- Use of MATLAB in visual exposition of text
- Rich pedagogy:
  - Diagrams: 480
  - Solved Examples: 310
  - Review Questions: 640
  - MATLAB Problems: 100

Chapter Organization

The book is divided into 17 chapters. **Chapter 1** introduces classification of signals and systems. A section on continuous-time and discrete-time signals is newly added and state-variable technique has been elaborated with examples in this chapter. **Chapter 2** is devoted to the Fourier analysis of periodic and aperiodic continuous-time signals and systems. **Chapter 3** focuses on the application of Laplace transforms to system analysis. **Chapter 4** is concerned with the evaluation of z-transforms and inverse z-transforms. **Chapter 5** discusses linear time-invariant systems. Additional sections such as discrete convolutions, solution of linear constant coefficient difference equation, frequency domain representation of discrete-time signals and systems have been included in this chapter.

Chapter 6 concentrates on discrete and Fast Fourier Transforms. **Chapter 7** explains finite impulse response (FIR) filters. **Chapter 8** discusses infinite impulse response (IIR) filters. **Chapter 9** deals with realization of digital linear systems. **Chapter 10** includes effects of finite word length in digital filters. **Chapter 11** describes multirate digital signal processing. **Chapter 12** deals with the discrete-time random signal processing.

Chapter 13 covers spectral estimation. Also, optimum digital filters are discussed in the chapter. **Chapter 14** contains adaptive filters. **Chapter 15** presents the applications of digital signal processing. Application of DSP in Biomedical Engineering and Wireless Communication is added in this chapter. **Chapter 16** discusses signal processors. Finally, **Chapter 17** elaborates on MATLAB programs with additional programs.

Appendices A to C present various tables like those of the sinc function, important formulae, series, and Chebyshev polynomials.

Online Learning Center

The website for this book can be accessed at https://www.mhhe.com/salivahanan/dsp3e and contains the following material:

For Instructors

- Solution Manual
- Power Point lecture slides

Acknowledgements

The author extends his sincere thanks to the management of SSN College of Engineering, Chennai for the constant encouragement. He wishes to express his gratitude to Dr V Palanisamy, former Principal, Government Engineering College, Salem, for writing a foreword for this book. Thanks are due to his colleagues and students for their valuable suggestions which have improved the book considerably over the previous edition. He is also thankful to Mr J Venkateshwaran of Mepco Schlenk Engineering College for efficiently word processing the manuscript of the first edition and Mr R Gopalakrishnan, Mr A Chakkarabani and Mr K Rajan of SSN College of Engineering for word processing the additional manuscript of this edition.

A special note of thanks goes to all those reviewers whose comments and suggestions have greatly improved this edition. Their names are given below.

Rajib Kar	*National Institute of Technology (NIT), Durgapur, West Bengal*
Rashmita Mishra	*B P Poddar Institute of Management and Technology, Kolkata, West Bengal*
S Ravi	*Madurai Institute of Technology, Madurai, Tamil Nadu*

The author is grateful to the editorial and production teams including Mrs Vibha Mahajan, Mr Ebi John Amos, Ms Koyel Ghosh, Ms Sohini Mukherjee and Ms Anjali Razdan of McGraw Hill Higher Education (India) for their initiation and support in bringing out this revised edition in a short span of time.

The author is greatly thankful to his wife, Mrs Kalavathy and sons, Santhosh Kanna and Subadesh Kanna, for their spirit of self-denial and enormous patience during the preparation of the revised edition.

Readers are welcome to give constructive suggestions for the improvement of the book.

S Salivahanan

Publisher's Note

McGraw-Hill Education (India) invites suggestions and comments from you, all of which can be sent to *info.india@mheducation.com* (kindly mention the title and author name in the subject line).

Piracy-related issues may also be reported.

Chapter 1

Classification of Signals and Systems

INTRODUCTION 1.1

Signals play a major role in our life. In general, a signal can be a function of time, distance, position, temperature, pressure, etc., and it represents some variable of interest associated with a system. For example, in an electrical system the associated signals are electric current and voltage. In a mechanical system, the associated signals may be force, speed, torque, etc. In addition to these, some examples of signals that we encounter in our daily life are speech, music, picture and video signals. A signal can be represented in a number of ways. Most of the signals that we come across are generated naturally. However, there are some signals that are generated synthetically. In general, a signal carries information, and the objective of signal processing is to extract this information.

Signal processing is a method of extracting information from the signal which in turn depends on the type of signal and the nature of information it carries. Thus signal processing is concerned with representing signals in mathematical terms and extracting the information by carrying out the algorithmic operations on the signal. Mathematically, a signal can be represented in terms of basis functions in the domain of the original independent variable or it can be represented in terms of basic functions in a transformed domain. Similarly, the information contained in the signal can also be extracted either in the original domain or in the transformed domain.

A system may be defined as an integrated unit composed of diverse, interacting structures to perform a desired task. The task may vary such as filtering of noise in a communication receiver, detection of range of a target in a radar system, or monitoring the steam pressure in a boiler. The function of a system is to process a given input sequence to generate an output sequence.

It is said that the origin of digital signal processing techniques can be traced to the seventeenth century when finite difference methods, numerical integration methods, and numerical interpolation methods were developed to solve physical problems involving continuous variables and functions. There has been tremendous growth since then and today digital signal processing techniques are applied in almost every field. The main reasons for such wide applications are due to the numerous advantages of digital signal processing techniques.

1.1.1 Advantages of Digital Signal Processing (DSP) over Analog Signal Processing (ASP)

Digital circuits do not depend on precise values of digital signals for their operation. Digital circuits are less sensitive to changes in component values. They are also less sensitive to variations in temperature, ageing and other external parameters. Digital processing is stable, reliable, flexible, predictable and repeatable.

In a digital processor, the signals and system coefficients are represented as binary words. This enables one to choose any accuracy by increasing or decreasing the number of bits in the binary word.

Digital processing of a signal facilitates the sharing of a single processor among a number of signals by time-sharing. This reduces the processing cost, size, weight and maintenance per signal. Also DSP can save both filtered and unfiltered data for further use.

Digital implementation of a system allows easy adjustment of the processor characteristics during processing. Adjustments in the processor characteristics can be easily done by periodically changing the coefficients of the algorithm representing the processor characteristics. Such adjustments are often needed in adaptive filters.

Digital processing of signals also has a major advantage which is not possible with the analog techniques. With digital filters, linear phase characteristics can be achieved. Digital filters can be made to work over a wide range of frequencies by a mere change in the sampling frequency. Also multirate processing is possible only in the digital domain. Digital circuits can be connected in cascade without any loading problems, whereas this cannot be easily done with analog circuits.

Storage of digital data is very easy. A signal can be stored on various storage media such as magnetic tapes, disks and optical disks without any loss. On the other hand, stored analog signals deteriorate rapidly as time progresses and cannot be recovered in their original form.

For processing very low frequency signals like seismic signals, the analog circuits require inductors and capacitors of very large size, whereas digital processing is more suited for such applications.

1.1.2 Disadvantages of DSP over ASP

Though the advantages are many, there are some drawbacks associated with processing a signal in the digital domain. Digital processing needs 'pre' and 'post' processing devices like analog-to-digital and digital-to-analog converters and associated reconstruction filters. This increases the complexity of the digital system. Also, digital techniques suffer from frequency limitations. For reconstructing a signal from its sample, the sampling frequency must be at least twice the highest frequency component present in that signal. The available frequency range of operation of a digital signal processor is primarily determined by the sample-and-hold circuit and the analog-to-digital converter, and as a result the frequency range is limited by the technology available at that time. The highest sampling frequency is presently around 1 GHz reported by K. Poulton et al., in 1987. However, such high sampling frequencies are not used since the resolution of the A/D converter decreases with an increase in the speed of the converter. A variety of analog processing algorithms can be implemented using passive circuits employing inductors, capacitors and resistors that do not need any power, whereas a DSP chip containing over four lakh transistors dissipates more power around 1 watt. Moreover, active devices are less reliable than passive components. But the advantages of the digital processing techniques outweigh the disadvantages in many applications. Also, the cost of DSP hardware is decreasing continuously. Consequently, the applications of digital signal processing are increasing rapidly.

Applications of DSP

Some selected applications of digital signal processing that are often encountered in daily life are listed as follows:

1. *Telecommunication* Echo cancellation in telephone networks, adaptive equalisation, ADPCM transcoders, telephone dialing application, modems, line repeaters, channel multiplexing, data communication, data encryption, video conferencing, cellular phone and FAX.
2. *Military* Radar signal processing, sonar signal processing, navigation, secure communications and missile guidance.
3. *Consumer electronics* Digital Audio/TV, electronic music synthesiser, educational toys, FM stereo applications and sound recording applications.
4. *Instrumentation and control* Spectrum analysis, position and rate control, noise reduction, data compression, digital filter, PLL, function generator, servo control, robot control and process control.
5. *Image processing* Image representation, image compression, image enhancement, image restoration, image reconstruction, image analysis and recognition, pattern recognition, robotic vision, satellite weather map and animation.
6. *Speech processing* Speech analysis methods are used in automatic speech recognition, speaker verification and speaker identification. Speech synthesis techniques includes conversion of written text into speech, digital audio and equalisation.
7. *Medicine* Medical diagnostic instrumentation such as computerised tomography (CT), X-ray scanning, magnetic resonance imaging, spectrum analysis of ECG and EEG signals to detect various disorders in heart and brain, scanners, patient monitoring and X-ray storage/enhancement.
8. *Seismology* DSP techniques are employed in geophysical exploration for oil and gas, detection of underground nuclear explosion and earthquake monitoring.
9. *Signal filtering* Removal of unwanted background noise, removal of interference, separation of frequency bands and shaping of the signal spectrum.

CONTINUOUS-TIME AND DISCRETE-TIME SIGNALS 1.2

Signals can be classified based on their nature and characteristics in the time domain. They are broadly classified as (i) *continuous-time signals* and (ii) *discrete-time signals*. A continuous-time signal is a mathematically continuous function and the function is defined continuously in the time domain. On the other hand, a discrete-time signal is specified only at certain time instants. The amplitude of the discrete-time signal between two time instants is not defined. Figure 1.1 shows typical continuous-time and discrete-time signals.

Discrete-time Signals—Sequences

A discrete-time signal has a value defined only at discrete points in time and a discrete-time system operates on and produces discrete-time signals. A discrete-time signal is a sequence which is a function defined on the positive and negative integers, that is, $x(n) = \{x(n)\} = \left\{\begin{matrix} ...x(-1), x(0), x(1),... \\ \uparrow \end{matrix}\right\}$ where the up-arrow represents the sample at $n = 0$. Here n is an integer indicating the sample numbers as counted from a chosen time origin, i.e. $n = 0$. The negative values of n correspond to negative time. The function of

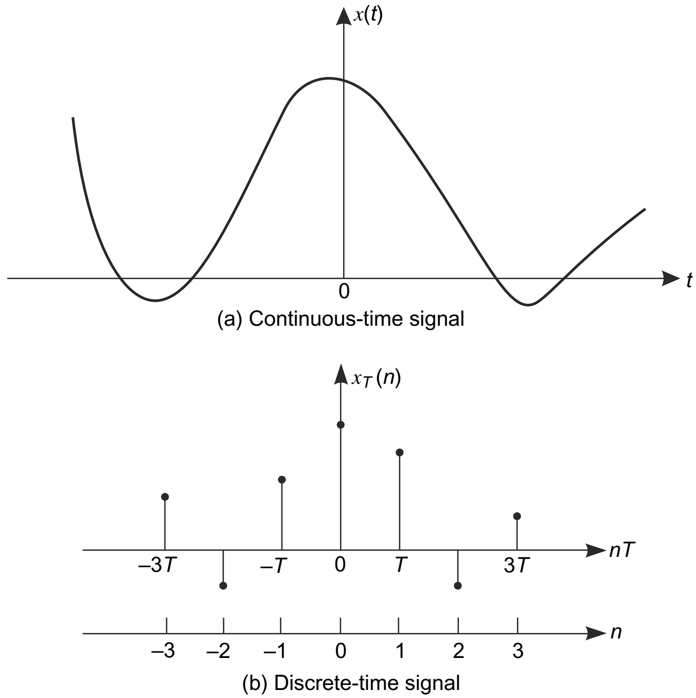


Fig. 1.1 *Continuous-Time and Discrete-Time Signals*

n is referred to as a sequence of samples, or sequence in short. If a continuous-time signal $x(t)$ is sampled every T seconds, a sequence $x(nT)$ results. In general, the sequence values are called samples and the interval between them is called the sample interval, T. For convenience, the sample interval T is taken as 1 second and hence $x(n)$ represents the sequence. The important sequences are the unit sample sequence, the unit step response, the exponential sequence and the sinusoidal sequence.

1.2.1 Unit-impulse Function

The unit-impulse function is defined as

$$\delta(t) = 0, t \neq 0 \tag{1.1}$$

and

$$\int_{-\infty}^{\infty} \delta(t)dt = 1 \tag{1.2}$$

The Eqs. 1.1 and 1.2 indicate that the area of the impulse function is unity and this area is confined to an infinitesimal interval on the t-axis and concentrated at $t = 0$. The unit impulse function is very useful in continuous-time system analysis. It is used to generate the system response providing fundamental information about the system characteristics. In discrete-time domain, the unit-impulse signal is called a unit-sample signal.

It is defined as

$$\delta(n) = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \tag{1.3}$$

Similarly, the shifted unit-impulse sequence $\delta[n - k]$ is defined as

$$\delta[n-k] = \begin{cases} 1, & n = k \\ 0, & n \neq k \end{cases} \tag{1.4}$$

which is shown in Fig. 1.2(b).

The shifted unit-impulse sequence $\delta[n + k]$ is defined as

$$\delta[n+k] = \begin{cases} 1, & n = -k \\ 0, & n \neq -k \end{cases} \tag{1.5}$$

which is shown in Fig. 1.2(b).

1.2.2 Unit-step Function

The integral of the impulse function $\delta(t)$ gives,

$$\int_{-\infty}^{t} \delta(t)\,\mathrm{d}t = \begin{cases} 1, & t > 0 \\ 0, & t < 0 \end{cases} \tag{1.6}$$

Since, the area of the impulse function is all concentrated at $t = 0$ for any value of $t < 0$ the integral becomes zero and for $t > 0$, from Eq. 1.2, the value of the integral is unity. The integral of impulse function is called the unit-step function, which is represented as

$$u(t) = \begin{cases} 1, & t > 0 \\ 0, & t < 0 \end{cases} \tag{1.7}$$

The value at $t = 0$ is taken to be finite and in most cases it is unspecified. The discrete-time unit-step signal shown in Fig. 1.2(c) is given by

$$u(n) = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \tag{1.8}$$

Similarly, the shifted unit-step sequence $u[n - k]$ is defined as

$$u[n-k] = \begin{cases} 1, & n \geq k \\ 0, & n < k \end{cases} \tag{1.9}$$

which is shown in Fig. 1.2(d).

The shifted unit-step sequence $u[n + k]$ is defined as

$$u[n+k] = \begin{cases} 1, & n \geq -k \\ 0, & n < -k \end{cases} \tag{1.10}$$

which is shown in Fig. 1.2(d).

1.2.3 Unit-ramp Function

The unit-ramp funtion $r(t)$ can be obtained by integrating the unit-impulse function twice or integrating the unit-step function once, i.e.

$$r(t) = \int_{-\infty}^{t} \int_{-\infty}^{\alpha} \delta(\tau)\, \mathrm{d}\tau \mathrm{d}\alpha \tag{1.11}$$

$$= \int_{-\infty}^{t} u(\alpha)\, \mathrm{d}\alpha = \int^{t} 1 \cdot \mathrm{d}\alpha$$

That is,

$$r(t) = \begin{cases} 0, & t < 0 \\ t, & t > 0 \end{cases} \tag{1.12}$$

A ramp signal starts at $t = 0$ and increases linearly with time, t. The unit-ramp signal in discrete-time domain, shown in Fig. 1.2(e) is given by

$$r(n) = \begin{cases} 0, & n < 0 \\ n, & n \geq 0 \end{cases} \tag{1.13}$$

1.2.4 Unit-pulse Function

A unit-pulse function, $\Pi(t)$, is obtained from unit-step signals as shown in Fig. 1.2(f) and is given by

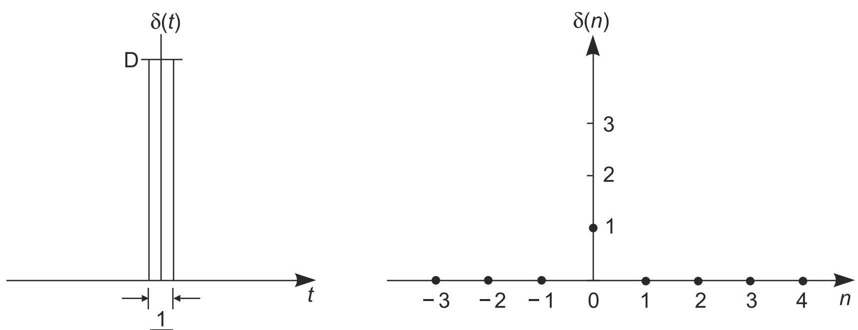
$$\Pi(t) = u\left(t + \frac{1}{2}\right) - u\left(t - \frac{1}{2}\right) \tag{1.14}$$

The signals $u\left(t + \frac{1}{2}\right)$ and $u\left(t - \frac{1}{2}\right)$ are the unit-step signals shifted by $\frac{1}{2}$ units in the time axis towards the left and right, respectively.

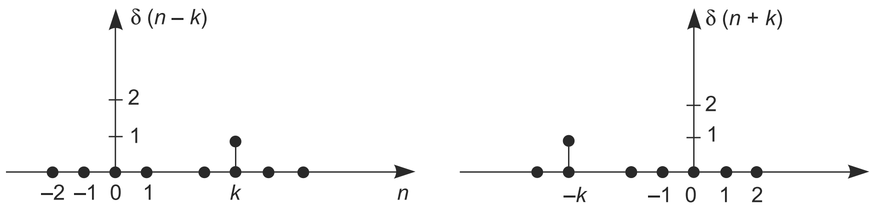
Figure 1.2 shows the graphical representation of all the above functions.

1.2.5 Complex Exponential Signal

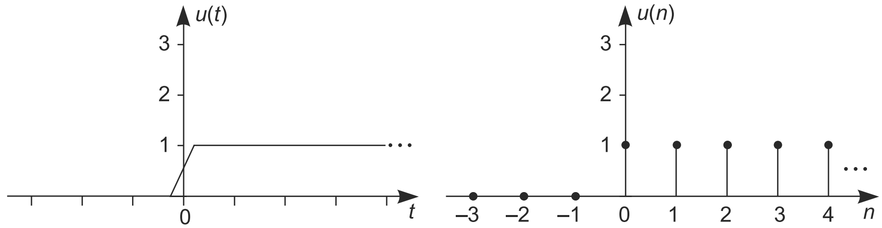
The complex exponential signal is defined as $x(t) = e^{st}$ where $s = \sigma + j\omega$, a complex number.



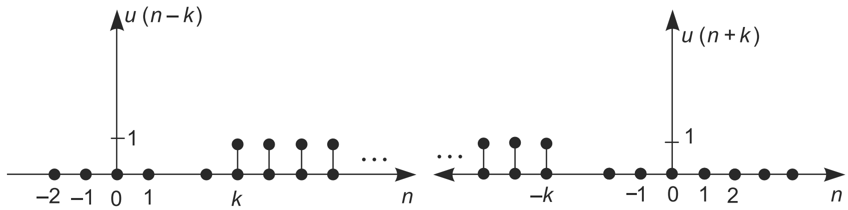
(a) Unit-impulse function



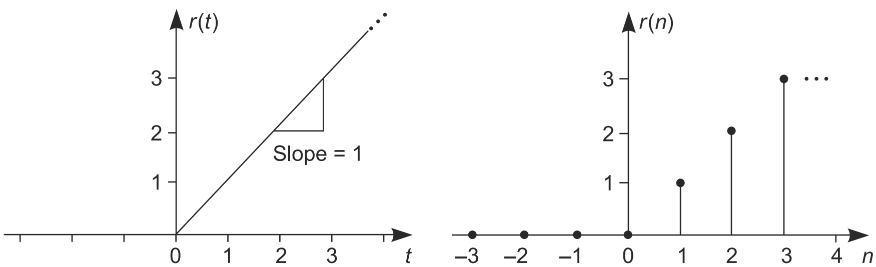
(b) Shifted unit-impulse sequence



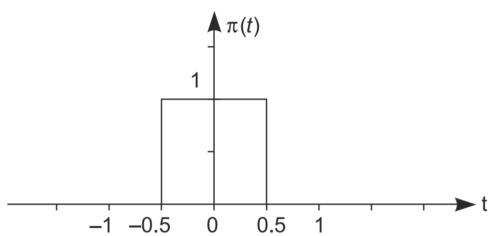
(c) Unit-step function



(d) Shifted unit-step sequence



(e) Unit-ramp function



(f) Unit-pulse function

Fig. 1.2 *(a) Unit-impulse Function (b) Shifted Unit-impulse Sequence (c) Unit-step Function (d) Shifted Unit-step sequence (e) Unit-ramp Function (f) Unit-pulse Function*

$$x(t) = e^{st} = e^{(\sigma + j\omega)t}$$
$$= e^{\sigma t} (\cos \omega t + j \sin \omega t) \quad (1.15)$$

Then this signal $x(t)$ is known as a general complex exponential signal whose real part $e^{\sigma t} \cos \omega t$ and imaginary part $e^{\sigma t} \sin \omega t$ are exponentially increasing ($\sigma > 0$) or decreasing $\sigma < 0$ as shown in Fig. 1.3(a) and (b) respectively.

If $s = \sigma$, then $x(t) = e^{\sigma t}$ which is a real exponential signal. As shown in Fig. 1.4, if ($\sigma > 0$), then $x(t)$ is a growing exponential; and if ($\sigma < 0$), then $x(t)$ is a decaying exponential. When $\sigma = 0$, $x(t) = 1$, is a constant.

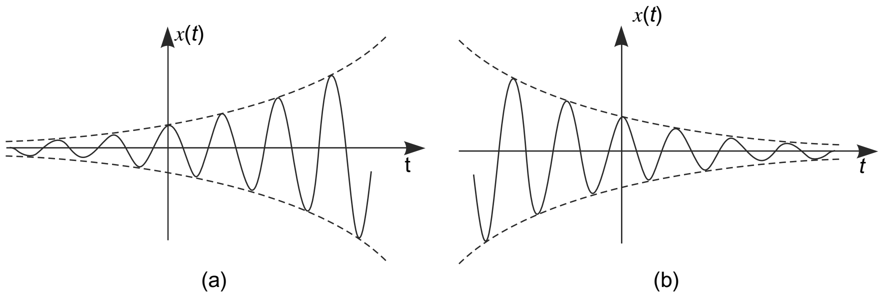


Fig. 1.3 *(a) Exponentially Increasing Sinusoidal Signal and (b) Exponentially Decreasing Sinusoidal Signal*

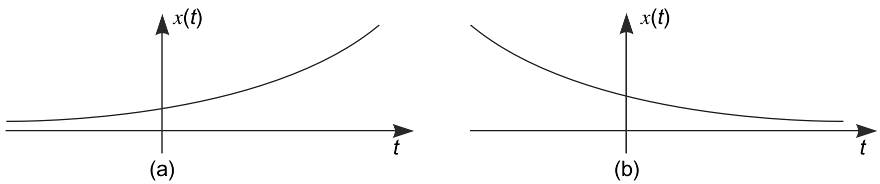


Fig. 1.4 *Continuous-time Real Exponential Signals (a) ($\sigma > 0$) and (b) ($\sigma < 0$)*

1.2.6 Complex Exponential Sequence

Let $x(n) = Ca^n$ be the complex exponential sequence, where 'C' and 'a' are in general complex numbers expressed as

$$C = Ae^{j\theta}$$
$$a = re^{j\omega_0}$$

Now,
$$x(n) = Ae^{j\theta} (re^{j\omega_0})^n \quad (1.16)$$
$$= Ar^n e^{j\left(\omega_0^{n+\theta}\right)}$$

For $r = 1$,

$$x(n) = Ae^{j(\omega_0 n + \theta)}$$
$$x(n) = A(\cos(\omega_0 n + \theta) + j\sin(\omega_0 n + \theta)) \quad (1.17)$$

Thus $x(n)$ is a complex signal whose real part is $A \cos(\omega_0 n + \theta)$ and imaginary part is $A \sin(\omega_0 n + \theta)$.

Figures 1.5 (a)–(c) show the graphical representation of complex exponential sequences for various cases.

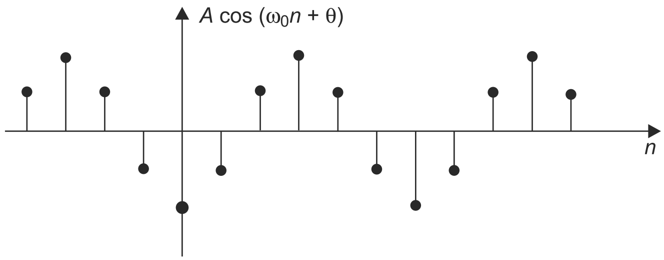


Fig. 1.5(a) *Real Part of Discrete-time Complex Exponential for r = 1*

When $r < 1$, we get a complex signal whose real and imaginary parts are damped sinusoidal signals.

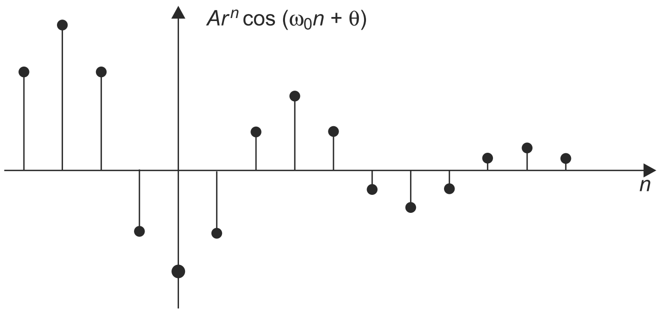


Fig. 1.5(b) *Real Part of Discrete-time Complex Exponential for r < 1*

When $r > 1$, we get a complex signal whose real and imaginary parts are growing sinusoids.

If $x(n) = Ar^n$, we get real discrete time exponential sequence. Depending on the value of r, we get different cases to consider.

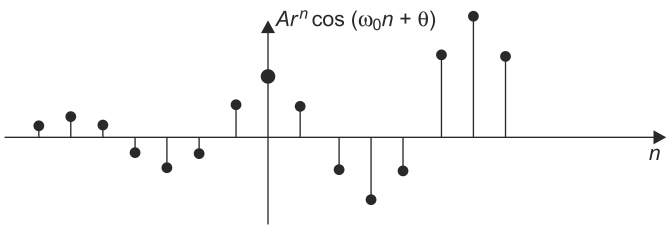


Fig. 1.5(c) *Real Part of Discrete-time Complex Exponential for r > 1*

Case (i) If $r = 0$, $x(n) = 0$

Case (ii) If $r > 1$, the magnitude of real exponential sequence grows exponentially as shown in Fig. 1.6(a).

Case (iii) If $0 < r < 1$, the magnitude of real exponential sequence decays exponentially as shown in Fig. 1.6(b).

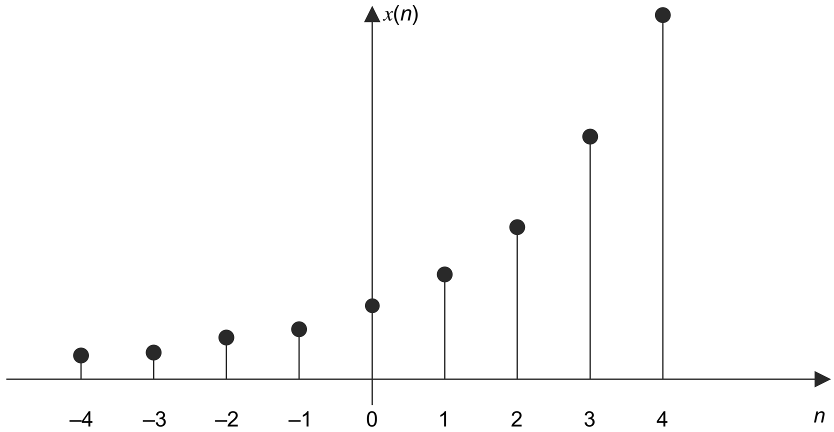


Fig. 1.6(a) *Discrete-time Real Exponential Sequence for* $r > 1$

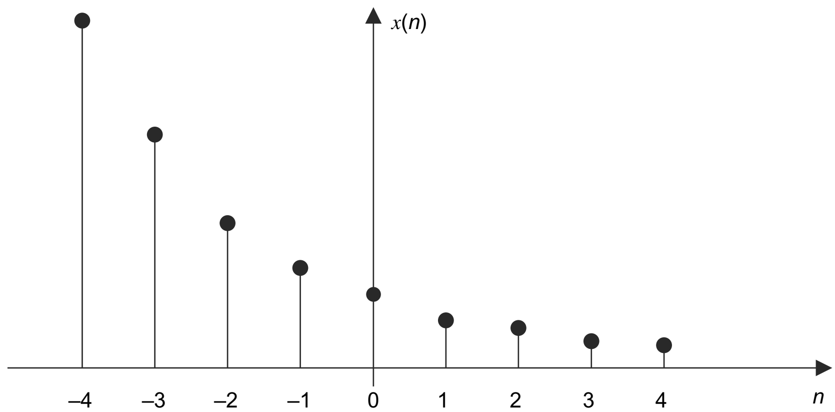


Fig. 1.6(b) *Discrete-time Real Exponential Sequence for* $0 < r < 1$

Case (iv) If $r < -1$, the sequence grows exponentially and alternates between positive and negative values as shown in Fig. 1.6(c).

Case (v) If $-1 < r < 0$, the sequence decays exponentially and alternates between positive and negative values as shown in Fig. 1.6(d).

Case (vi) If $r = -1$, the signal alternates between A and $-A$ as shown in Fig. 1.6(e).

1.2.7 Sinusoidal Signal

A continuous-time sinusoidal signal can be represented as

$$x(t) = A \cos (\omega_0 t + \theta) \tag{1.18}$$

where A is the amplitude, ω_0 is the radian frequency in radians per second, and θ is the phase angle in radians. The sinusoidal signal $x(t)$ is shown in Fig. 1.7, which is periodic with fundamental period

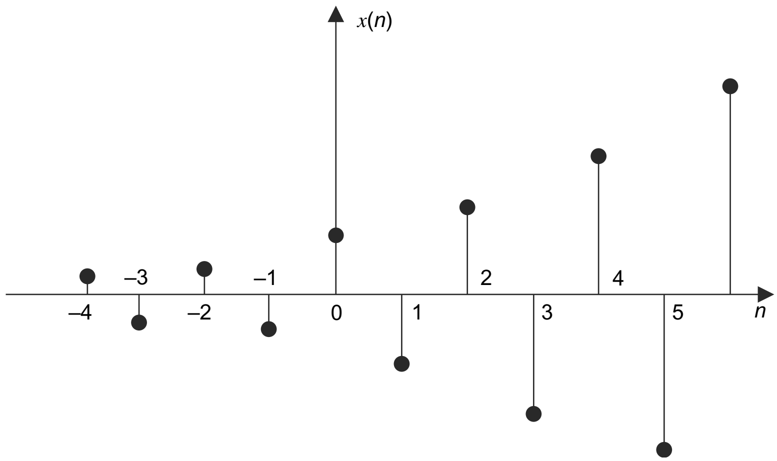


Fig. 1.6(c) *Discrete-time Real Exponential Sequence for* $r < -1$

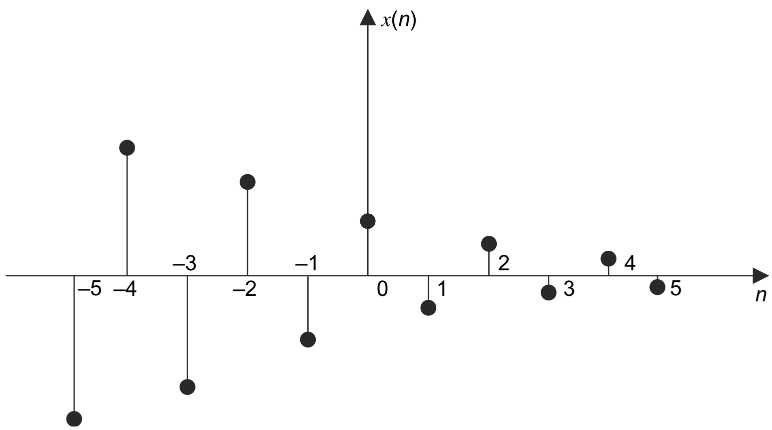


Fig. 1.6(d) *Discrete-time Real Exponential Sequence for* $-1 < r < 0$

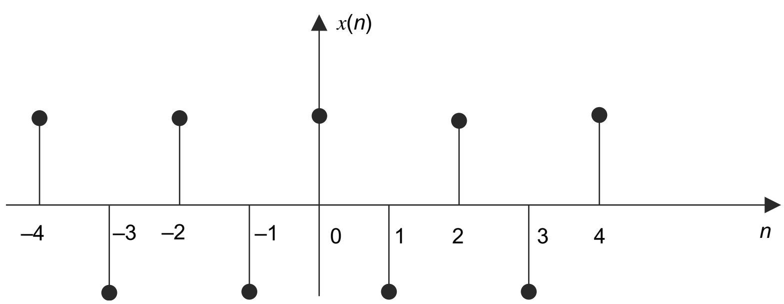


Fig. 1.6(e) *Discrete-time Real Exponential Sequence for* $r = -1$

$$T_0 = \frac{2\pi}{\omega_0}$$

The reciprocal of the fundamental period T_0 is called the fundamental frequency f_0.

$$f_0 = \frac{1}{T_0}$$

Therefore, $\omega_0 = 2\pi f_0$ which is called the fundamental angular frequency. Using Euler's formula, we have the real part as

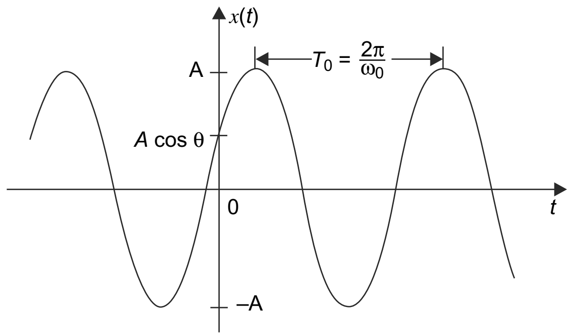


Fig. 1.7 *Continuous-time Sinusoidal Signal*

$$A\,\mathrm{Re}\{e^{j(\omega_0 t + \theta)}\} = A\cos(\omega_0 t + \theta) \tag{1.19}$$

and imaginary part as

$$A\,\mathrm{Im}\{e^{j(\omega_0 t + \theta)}\} = A\sin(\omega_0 t + \theta) \tag{1.20}$$

Example 1.1 *Find the following summation:*

(a) $\sum_{n=-\infty}^{\infty} e^{2n}\delta(n-2)$ (b) $\sum_{n=-5}^{5} \sin 2n\,\delta(n+7)$

Solution

(a) $\sum_{n=-\infty}^{\infty} e^{2n}\delta(n-2) = e^{2n}\Big|_{n=2} = e^4$

(b) $\sum_{n=-5}^{5} \sin 2n\delta(n+7) = 0$

1.2.8 Properties of $\delta(t)$

1. $\int_{-\infty}^{\infty} \delta(t)dt = 1$
2. $\int_{-\infty}^{\infty} x(t)\delta(t)dt = x(0)$
3. $\int_{-\infty}^{\infty} x(t)\delta(t-t_0)dt = x(t_0)$
4. $\int_{-\infty}^{\infty} x(\lambda)\delta(t-\lambda)d\lambda = x(t)$
5. $\delta(at) = \frac{1}{|a|}\delta(t)$
6. $\delta(-t) = \delta(t)$
7. $x(t)\,\delta(t) = x(0)$
8. $\delta(t) = \frac{du(t)}{dt}$
9. $u(t) = \int_{-\infty}^{t} \delta(\tau)d\tau$
10. $x(t)\,\delta(t-t_0) = x(t_0)$
11. $x(t_0)\,\delta(t-t_0) = x(t)$
12. $\int_{t_1}^{t_2} x(t)\delta^n(t-t_0)dt = (-1)^n x^n(t_0)$

Proof

$$\frac{d}{dt}\left[x(t)\delta(t-t_0)\right] = x(t)\dot{\delta}(t-t_0) + \dot{x}(t)\delta(t-t_0)$$

$$= x(t)\dot{\delta}(t-t_0) + \dot{x}(t_0),\ t_1 < t_0 < t_2$$

Integrating, we get

$$\int_{t_1}^{t_2} \frac{d}{dt}\left[x(t)\delta(t-t_0)\right]dt = \int_{t_1}^{t_2}\left[x(t)\dot{\delta}(t-t_0)\right]dt + \int_{t_1}^{t_2} \dot{x}(t_0)dt$$

$$\left[x(t)\delta(t-t_0)\right]_{t_1}^{t_2} = \int_{t_1}^{t_2} x(t)\dot{\delta}(t-t_0)dt + \dot{x}(t_0)$$

LHS = 0.
Therefore,

$$\int_{t_1}^{t_2} x(t)\dot{\delta}(t-t_0)dt + \dot{x}(t_0) = 0$$

i.e., $$\int_{t_1}^{t_2} x(t)\dot{\delta}(t-t_0)\,dt = -\dot{x}(t_0)$$

Similarly, $$\int_{t_1}^{t_2} x(t)\ddot{\delta}(t-t_0)dt = -\ddot{x}(t_0)$$

Hence, $$\int_{t_1}^{t_2} x(t)\delta^n(t-t_0)dt = (-1)^n x^n(t_0)$$

CLASSIFICATION OF SIGNALS 1.3

Both continuous-time and discrete-time signals are further classified as

(i) Deterministic and non-deterministic signals
(ii) Periodic and aperiodic signals
(iii) Even and odd signals
(iv) Causal and noncausal signals
(v) Energy and power signals

1.3.1 Deterministic and Non-deterministic Signals

Deterministic signals are functions that are completely specified in time. The nature and amplitude of such a signal at any time can be predicted. The pattern of the signal is regular and can be characterised mathematically. Examples of deterministic signals are

(i) $x(t) = \alpha t$ This is a ramp whose amplitude increases linearly with time t and slope α.
(ii) $x(t) = A \sin \omega t$. The amplitude of this signal varies sinusoidally with time and its maximum amplitude is A.

(iii) $x(n) = \begin{cases} 1, & n \geq 0 \\ 0, & \text{otherwise} \end{cases}$ This is a discrete-time signal whose amplitude is 1 for the sampling instants $n \geq 0$ and for all other samples, the amplitude is zero.

For all the signals given above, the amplitude at any time instant can be predicted in advance. Contrary to this, a non-deterministic signal is one whose occurrence is random in nature and its pattern is quite irregular. A typical example of a non-deterministic signal is **thermal noise** in an electrical circuit. The behaviour of such a signal is probabilistic in nature and can be analysed only stochastically. Another example which can be easily understood is the number of accidents in an year. One cannot exactly predict what would be the figure in a particular year and this varies randomly. Non-deterministic signals are also called **random signals**.

1.3.2 Periodic and Aperiodic Signals

A continuous-time signal is said to be periodic if it exhibits periodicity, i.e.

$$x(t + T) = x(t), \quad -\infty < t < \infty \tag{1.21}$$

where T is the period of the signal. The smallest value of T that satisfies Eq. 1.21 is called the fundamental period, T_0, of the signal. A periodic signal has a definite pattern that repeats over and over, with a repetition period of T_0. For a discrete-time signal, the condition for periodicity can be written as,

$$x(n + N_0) = x(n), \quad -\infty < n < \infty \tag{1.22}$$

where N_0 is the sampling period measured in units of number of sample spacings. Periodic signals can be in general, expressed as

(i) Continuous-Time Periodic Signals

$$x_p(t) = \sum_{i=-\infty}^{\infty} X(t - iT_0) \tag{1.23}$$

where

$$X(t) = \begin{Bmatrix} x(t), & t_1 \leq t < t_1 + T_0 \\ 0, & \text{elsewhere} \end{Bmatrix} \tag{1.24}$$

(ii) Discrete-Time Periodic Signals

$$x_p(n) = \left[\sum_{i=-\infty}^{\infty} X(n - iN_0)\, T \right] \tag{1.25}$$

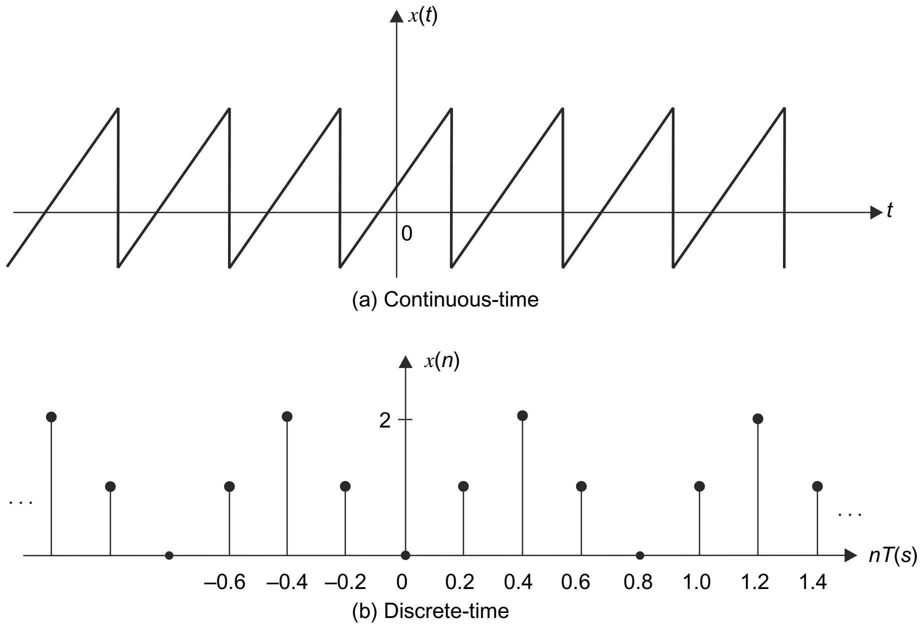
where

$$X(n) = \begin{Bmatrix} x(n), & n_1 \leq n < (n_1 + N_0) \\ 0, & \text{elsewhere} \end{Bmatrix} \tag{1.26}$$

and T is the sampling period in seconds.

A signal which does not satisfy either Eq. 1.21 or Eq. 1.22 is called an aperiodic signal. Some examples of period signals are shown in Fig. 1.8. Some periodic signals can be simply modelled using a single equation. For example,

$$x(t) = A \sin\left(\frac{2\pi t}{T_0} - \phi\right) \tag{1.27}$$



$$x_p(n) = \left[\sum_{i=-\infty}^{\infty} X[n - iN_0]\, T \right]$$

where

$$X[n] = \begin{cases} -n & -2 \le n < 0 \\ n, & 0 \le n < 2 \\ 0, & \text{otherwise} \end{cases}$$

$$N_0 = 4 \quad \text{and} \quad T = 0.2$$

Fig. 1.8 *Some Examples of Periodic Signals*

is a continuous time sinusoidal signal which is valid for all t. The constant A represents the maximum amplitude and ϕ represents the phase shift of the sinusoidal signal. Now, the fundamental period T_0 of the above sinusoidal signal is given by

$$T_0 = \frac{2\pi}{\omega_0} \tag{1.28}$$

Similarly, a periodic discrete time sinusoidal signal is represented as

$$x(n) = \left[A \sin\left(\frac{2\pi n}{N_0} - \beta \right) T \right] \tag{1.29}$$

The term β represents the delay and T represents the sampling period. Even though, the signal $x(n)$ has a sinusoidal envelope it is not necessarily periodic. For this sequence to be periodic with period $N_0 (N_0 > 0)$, ω_0 should satisfy the condition,

$$\frac{\omega_0}{2\pi} = \frac{m}{N_0}$$

where m is a positive integer. Then the fundamental period of the sequence $x(n)$ is given by,

$$N_0 = m\left(\frac{2\pi}{\omega_0}\right) \tag{1.30}$$

The sum of two or more periodic continuous-time signals need not be periodic. They will be periodic if and only if the ratio of their fundamental periods is rational. In order to determine whether the sum of two or more periodic signals is periodic or not, the following steps may be used.

(i) Determine the fundamental period of the individual signals in the sum signal.

(ii) Find the ratio of the fundamental period of the first signal with the fundamental periods of every other signal.

(iii) If all these ratios are rational, then the sum signal is also periodic.

In the case of discrete-time signals, the sum of a number of periodic signals is always periodic because the ratio of individual periods is always the ratio of integers, which is rational.

Example 1.2 *Determine which of the following signals are periodic.*

(a) $x_1(t) = \sin 15\pi t$ (b) $x_2(t) = \sin 20\pi t$ (c) $x_3(t) = \sin \sqrt{2}\pi t$

(d) $x_4(t) = \sin 5\pi t$ (e) $x_5(t) = x_1(t) + x_2(t)$ (f) $x_6(t) = x_2(t) + x_4(t)$

(g) $x_7(t) = 20\cos\left[10\pi t + \frac{\pi}{6}\right]$

Solution

(a) $x_1(t) = \sin 15\pi t$ is periodic.

The fundamental period is $T_0 = \frac{2\pi}{\omega} = \frac{2\pi}{15\pi} = 0.1333333333...$ seconds

(b) $x_2(t) = \sin 20\pi t$ is periodic.

The fundamental period is $T_0 = \frac{2\pi}{\omega} = \frac{2\pi}{20\pi} = 0.1$ seconds

(c) $x_3(t) = \sin\sqrt{2}\pi t$ is periodic.

The fundamental period is $T_0 = \frac{2\pi}{\omega} = \frac{2\pi}{\sqrt{2}\pi} = 1.41421356...$ seconds

(d) $x_4(t) = \sin 5\pi t$ is periodic.

The fundamental period is $T_0 = \frac{2\pi}{\omega} = \frac{2\pi}{5\pi} = 0.4$ seconds

(e) $x_5(t) = x_1(t) + x_2(t)$

The fundamental period of $x_1(t) = T_{01} = \frac{2\pi}{\omega} = \frac{2\pi}{15\pi} = \frac{2}{15}$ seconds and the fundamental period of $x_2(t) = T_{02} = \frac{2\pi}{\omega} = \frac{2\pi}{20\pi} = \frac{1}{10}$ seconds. The ratio of fundamental frequencies, $\frac{T_{01}}{T_{02}} = \frac{4}{3}$, cannot be expressed as a ratio of integers. Hence, $x_5(t)$ is periodic.

(f) $x_6(t) = x_2(t) + x_4(t)$

The fundamental period of $x_2(t) = T_{02} = 0.1$ seconds and the fundamental period of $x_4(t) = T_{04} = 0.4$ seconds. The ratio of fundamental frequencies, $\frac{T_{02}}{T_{04}} = \frac{0.1}{0.4} = \frac{1}{4}$, can be expressed as a ratio of integers. Hence, $x_5(t)$ is periodic.

(g) $x_7(t) = 20\cos\left[10\pi t + \frac{\pi}{6}\right]$ is periodic with period $T_0 = \frac{2\pi}{\omega} = \frac{2\pi}{10\pi} = \frac{1}{5} = 0.2$ seconds.

Example 1.3 *Show that the complex exponential signal $x(t) = e^{j\omega_0 t}$ is periodic and its fundamental period is $2\pi/\omega_0$.*

Solution

We know that $x(t)$ will be periodic if $e^{j\omega_0(t+T)} = e^{j\omega_0}$

Since $e^{j\omega_0(t+T)} = e^{j\omega_0 t}e^{j\omega_0 T}$, we must have $e^{j\omega_0 T} = 1$.

If $\omega_0 = 0$, then $x(t) = 1$, which is periodic for any value of T.

If $\omega_0 \neq 0$, $\omega_0 T = m2\pi$ or $T = m\frac{2\pi}{\omega_0}$ (where m is a positive integer).

Thus, if $m = 1$, the fundamental period T_0 of $x(t)$ is given by $2\pi/\omega_0$.

Example 1.4 *Given the sinusoidal signal $x(t) = \cos 5t$.*

(a) *Determine the value of sampling interval T_s, such that $x[n] = x(nT_s)$ is a periodic sequence.*

(b) *Determine the fundamental period of $x[n] = x(nT_s)$ if $T_s = 0.1\pi$ seconds.*

Solution

(a) The fundamental period of $x(t)$ is $T_0 = 2\pi/\omega_0 = 2\pi/5$. Therefore, $x[n] = x(nT_s)$ is periodic if
$$\frac{T_s}{T_0} = \frac{T_s}{2\pi/5} = \frac{m}{N_0}$$
where m and N_0 are positive integers. Thus, the required value of T_s is given by
$$T_s = \frac{m}{N_0}T_0 = \frac{m}{N_0}\frac{2\pi}{5}$$

(b) Substituting $T_s = 0.1\pi = \pi/10$, we have
$$\frac{T_s}{T_0} = \frac{\pi/10}{2\pi/5} = \frac{5}{20} = \frac{1}{4}$$
Thus, $x[n] = x(nT_s)$ is periodic. Therefore, $N_0 = m\frac{T_0}{T_s} = 4m$

The smallest positive integer N_0 is obtained with $m = 1$. Thus, the fundamental period of $x[n] = x(0.1\pi n)$ is $N_0 = 4$.

Example 1.5 *Check whether each of the following signals is periodic or not. If a signal is periodic, find its fundamental period.*

(a) $x(t) = \cos t + \sin\sqrt{2}\,t$

(b) $x(t) = \sin^2 t$

(c) $x(t) = e^{j[(\pi/4)t-1]}$

(d) $x[n] = e^{j(\pi/4)n}$

(e) $x[n] = \cos\frac{1}{2}n$

(f) $x[n] = \cos\frac{\pi}{3}n + \sin\frac{\pi}{4}n$

(g) $x[n] = \cos^2\frac{\pi}{8}n$

(h) $x[n] = \cos(0.01\pi n)$

(i) $x[n] = \sin\left(\frac{6\pi}{7}n + 1\right)$

(j) $x(t) = 2\cos(10t + 1) - \sin(4t - 1)$

Solution

(a) Given $x(t)=\cos t+\sin\sqrt{2}t = x_1(t)+x_2(t)$

Here $x_1(t) = \cos t = \cos \omega_1 t$

The fundamental period of $x_1(t)$ is $T_1=\frac{2\pi}{\omega_1}=2\pi$

$x_2(t)=\sin\sqrt{2}t=\sin\omega_2 t$

The fundamental period of $x_2(t)$ is $T_2=\frac{2\pi}{\omega_2}=\frac{2\pi}{\sqrt{2}}=\sqrt{2}\pi$

The signal $x(t)$ is periodic, if $\frac{T_1}{T_2}$ is a rational number.

However, here $\frac{T_1}{T_2}=\sqrt{2}$, which is an irrational number.

Hence $x(t)$ is non-periodic.

(b) Given $x(t)=\sin^2 t=\frac{1}{2}-\frac{1}{2}\cos 2t=x_1(t)+x_2(t)$

$x_1(t)=\frac{1}{2}$, which is a dc signal; $T_1=\infty$

$x_2(t)$ is periodic if $T_2=\frac{2\pi}{\omega_2}=\frac{2\pi}{2}=\pi$

If a periodic signal rides over a dc signal, the resultant signal will also be periodic. Therefore, $x(t)$ is periodic with fundamental period. $T_0 = \pi$

(c) Given $x(t)=e^{j[(\pi/4)t-1]}=e^{-j}\,e^{j(\pi/4)t}=e^{-j}e^{j\omega_0 t}$

The term e^{-j} is not a function of time, t. Therefore, it can be considered that $x(t)$ is a complex exponential multiplied to a constant, i.e., $x(t)=k.e^{j\omega_0 t}$

Here, $\omega_0=\frac{\pi}{4}$

Therefore, $x(t)$ is periodic with fundamental period $T_0=\frac{2\pi}{\omega_0}=8$.

(d) Given $x(n)=e^{j(\pi/4)n}=e^{j\Omega_0 n}$

Here, $\Omega_0=\frac{\pi}{4}$

$\frac{\Omega_0}{2\pi}=\frac{1}{8}$ is a rational number and hence $x(n)$ is periodic.

Therefore, the fundamental period is $N_0 = 8$

(e) Given $x(n)=\cos\frac{1}{2}n=\cos\Omega_0 n$

Here, $\Omega_0=\frac{1}{2}$

$\frac{\Omega_0}{2\pi}=\frac{1}{4\pi}$ is not a rational number

Therefore, $x(n)$ is non-periodic.

(f) Given $x(n)=\cos\frac{\pi}{3}n=+\sin\frac{\pi}{4}n=x_1(n)+x_2(n)$

Here $x_1(n) = \cos\frac{\pi}{3}n = \cos\Omega_1 n$ where $\Omega_1 = \frac{\pi}{3}$

$x_2(n) = \sin\frac{\pi}{4}n = \sin\Omega_2 n$ where $\Omega_2 = \frac{\pi}{4}$

$\frac{\Omega_1}{2\pi} = \frac{1}{6}$ is a rational number; $x_1(n)$ is periodic with fundamental period $N_1 = 6$

$\frac{\Omega_2}{2\pi} = \frac{1}{8}$ is a rational number; $x_2(n)$ is periodic with fundamental period $N_2 = 8$

$\frac{N_1}{N_2} = \frac{6}{8} = \frac{3}{4}$; $4N_1 = 3N_2$

Hence $x(n)$ is periodic if $N_0 = 4N_1 = 3N_2 = 24$

(g) Given $x(n) = \cos^2\frac{\pi}{8}n = \frac{1}{2} + \frac{1}{2}\cos\frac{\pi}{4}n = x_1(n) + x_2(n)$

Here, $x_1(n) = \frac{1}{2} = \frac{1}{2}(1)^n$ is periodic with fundamental period $N_1 = 1$

$x_2(n) = \frac{1}{2}\cos\frac{\pi}{4}n = \frac{1}{2}\cos\Omega_2 n$ where $\Omega_2 = \frac{\pi}{4}$

$\frac{\Omega_2}{2\pi} = \frac{1}{8}$, which is a rational number. Therefore, $x_2(n)$ is periodic with fundamental period $N_2 = 8$.

Hence, $x(n)$ is periodic with fundamental period $N_0 = 8\ N_1 = N_2 = 8$

(h) Given $x(n) = \cos(0.01\ \pi n)$

Here, $\Omega_0 = 0.01\ \pi$

$\frac{\Omega_0}{2\pi} = \frac{0.01\pi}{2\pi} = \frac{1}{200}$ is a rational number.

Therefore, $x(n)$ is periodic with fundamental period $N_0 = 200$.

(i) Given $x(n) = \sin\left(\frac{6\pi}{7}n + 1\right) = \sin(\Omega_0 n + \phi)$

Here, $\Omega_0 = \frac{6\pi}{7}$

$\frac{\Omega_0}{2\pi} = \frac{\frac{6\pi}{7}}{2\pi} = \frac{3}{7}$ is a rational number.

Therefore, $x(n)$ is periodic with fundamental period $N_0 = 7$.

(j) Given $x(t) = 2\cos(10t + 1) - \sin(4t - 1)$

Let $x(t) = x_1(t) + x_2(t)$

Here, $x(t) = 2\cos(10t + 1)$

Therefore, the fundamental period of $x_1(t)$ is $T_1 = \frac{2\pi}{\omega_1} = \frac{2\pi}{10} = \frac{\pi}{5}$

$x_2(t) = -\sin(4t - 1)$

Therefore, the fundamental period of $x_2(t)$ is $T_2 = \frac{2\pi}{\omega_2} = \frac{2\pi}{4} = \frac{\pi}{2}$

Here $\frac{T_1}{T_2} = \frac{\frac{\pi}{5}}{\frac{\pi}{2}} = \frac{2}{5}$ is a rational number.

Therefore, $x(t)$ is periodic with period $5T_1$ or $2T_2$.

Hence, $T_0 = \pi$.

Example 1.6 *Determine the fundamental period of*

(a) $x_a(t)=\cos\left(\frac{\pi}{3}\right)t+\sin\left(\frac{\pi}{4}\right)t$ (b) $x_b(t)=\cos\left(2t+\frac{\pi}{3}\right)$

Solution

(a) $x_a(t)=\cos\left(\frac{\pi}{3}\right)t+\sin\left(\frac{\pi}{5}\right)t=x_1(t)+x_2(t)$

For $x_1(t), \omega_1 = \frac{\pi}{3}$

Therefore, the fundamental period of $x_1(t)$ is $T_1 = \frac{2\pi}{\omega_1} = \frac{2\pi}{\left(\frac{\pi}{3}\right)} = 6$

For $x_2(t)$, $\omega_2 = \frac{\pi}{4}$

Therefore, the fundamental period of $x_2(t)$ is $T_2 = \frac{2\pi}{\omega_2} = \frac{2\pi}{\left(\frac{\pi}{4}\right)} = 8$

Here, the fundamental period of $x(t)$ is $\frac{T_1}{T_2} = \frac{6}{8} = \frac{3}{4}$

Hence, $x(t)$ will be periodic, if $T_0 = 4T_1 = 3T_2 = 24$

(b) $x_b(t)=\cos\left(2t+\frac{\pi}{3}\right)=\cos(\omega t+\varphi)$

Here, $\omega = 2$

Therefore, the fundamental period for which the signal to be periodic is $T_0 = \frac{2\pi}{\omega} = \frac{2\pi}{2} = \pi$

1.3.3 Even and Odd Signals

If a signal exhibits symmetry in the time domain about the origin, it is called an *even signal*. The signal must be identical to its reflection about the origin. Mathematically, an even signal satisfies the following relation.

For a continuous-time signal, $x(t) = x(-t)$ (1.31a)

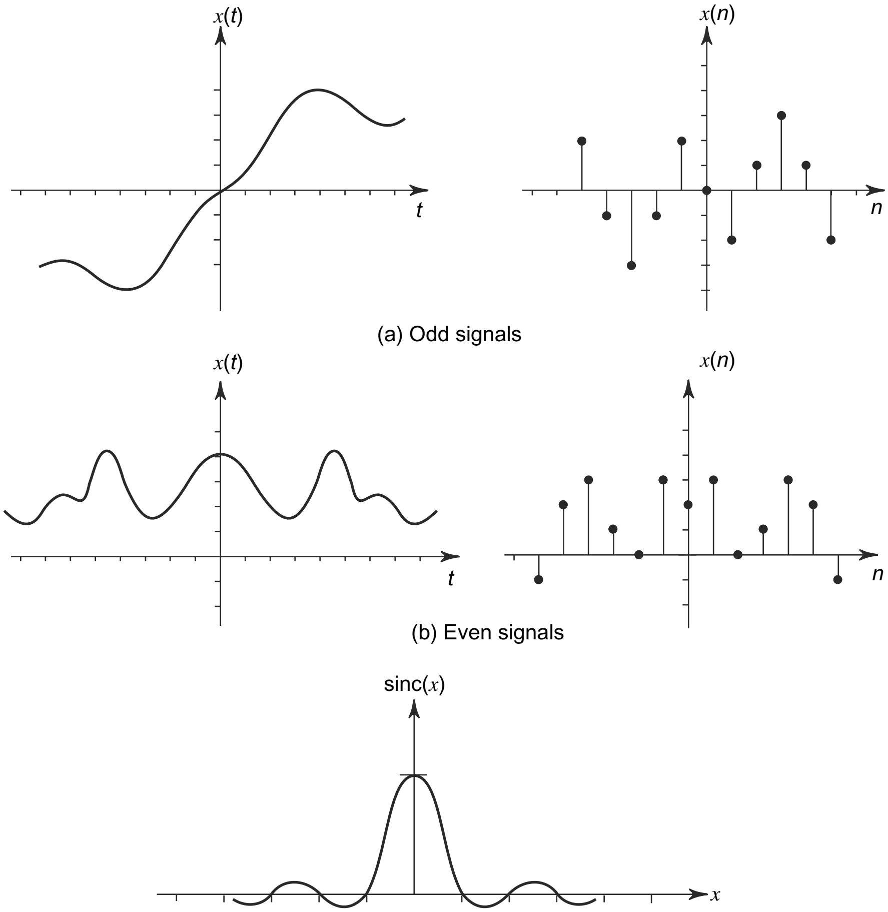
For a discrete-time signal, $x(n) = x(-n)$ (1.31b)

An *odd signal* exhibits anti-symmetry. The signal is not identical to its reflection about the origin, but to its negative. An odd signal satisfies the following relation.

For a continuous-time signal, $x(t) = -x(-t)$ (1.32a)

For a discrete-time signal, $x(n) = -x(-n)$ (1.32b)

$x_1(t) = \sin \omega t$ and $x_2(t) = \cos \omega t$ are good examples of odd and even signals, respectively. Figure 1.9 shows the typical odd and even signals. An even signal which often occurs in the analysis of signals is the sinc function. The sinc function may be expressed in the following two ways according to our convenience:



(a) Odd signals

(b) Even signals

(c) sinc (x) function

Fig. 1.9 *Typical Examples for (a) Odd Signal, (b) Even Signal and (c) the* sinc (*x*) *Function*

(i) $\text{sinc}(x) = \dfrac{\sin x}{x}$ and (ii) $\text{sinc}(x) = \dfrac{\sin(\pi x)}{\pi x}$. In Chapter 2, the first expression is used.

The area under the sinc function is unity. The sinc function is shown in Fig. 1.9(c). The positive portions of the sinc function have angles of $\pm n\pi$ where n is an even integer, and the negative portions of the sinc function have angles of $\pm m\pi$ where m is odd. It can be seen from Fig. 1.9(c) that the sinc function exhibits symmetry about $x = 0$.

A signal can be expressed as a sum of two components, namely, the even component of the signal and the odd component of the signal. The even and odd components can be obtained from the signal itself, as given below.

$$x(t) = x_{\text{even}}(t) + x_{\text{odd}}(t) \tag{1.33}$$

where

$$x_{\text{even}}(t) = \frac{1}{2}[x(t) + x(-t)] \text{ and } x_{\text{odd}}(t) = \frac{1}{2}[x(t) - x(-t)]$$

Example 1.7 *Determine the even and odd components of $x(t) = e^{jt}$.*

Solution Let $x_e(t)$ and $x_0(t)$ be the even and odd components of e^{jt}, respectively.

$$e^{jt} = x_e(t) + x_0(t)$$

Using Euler's formula, we obtain

$$e^{jt} = \cos t + j \sin t$$

Comparing the two forms of e^{jt}, we can straightaway write $x_e(t) = \cos t$ and $x_0(t) = j \sin t$.

Alternatively,

$$x_e(t) = \frac{x(t)+x(-t)}{2} = \frac{1}{2}(e^{jt} + e^{-jt}) = \cos t$$

$$x_0(t) = \frac{x(t)+x(-t)}{2} = \frac{1}{2}(e^{jt} - e^{-jt}) = j \sin t$$

Example 1.8 *Show that the product of two even signals or of two odd signals is an even signal and that the product of an even and an odd signal is an odd signal.*

Solution Let $x(t) = x_1(t)\, x_2(t)$.

If $x_1(t)$ and $x_2(t)$ are both even, then

$$x(-t) = x_1(-t)\,x_2(-t) = x_1(t)\,x_2(t) = x(t)$$

And $x(t)$ is even.

If $x_1(t)$ and $x_2(t)$ are both odd, then

$$x(-t) = x_1(-t)\,x_2(-t) = [-x_1(t)]\,[-x_2(t)] = x_1(t)x_2(t) = x(t)$$

and $x(t)$ is even.

If $x_1(t)$ is even and $x_2(t)$ is odd, then

$$x(-t) = x_1(-t)\,x_2(-t) = x_1(t)[-x_2(t)] = -x_1(t)x_2(t) = -x(t)$$

and $x(t)$ is odd. Here, the variable 't' represents either a continuous or a discrete variable.

Example 1.9 *Determine the even and odd components of $x(t) = \cos t + \sin t$*

Solution Given

$$x(t) = \cos t + \sin t$$

$$x(-t) = \cos(-t) + \sin(-t) = \cos t - \sin t$$

The even component is

$$x_e(t) = \frac{x(t)+x(-t)}{2} = \frac{(\cos t + \sin t) + (\cos t - \sin t)}{2} = \cos t$$

The odd component is

$$x_0(t) = \frac{x(t)-x(-t)}{2} = \frac{(\cos t + \sin t) - (\cos t - \sin t)}{2} = \sin t$$

1.3.4 Causal and Non-causal Signals

A continuous time signal is said to be causal if its amplitude is zero for negative time,

i.e., $x(t) = 0$ for $t < 0$

For a discrete time signal, the condition for causality is

$x(n) = 0$ for $n < 0$

$x(t) = u(t)$, the unit step function is a good example for a causal signal.

A signal is said to be anticausal if its amplitude is zero for positive time,

i.e., for a continuous time signal, $x(t) = 0$ for $t > 0$

for a discrete time signal, $x(n) = 0$ for $n > 0$

A signal which is neither causal nor anticausal is called a non-causal signal.

1.3.5 Energy and Power Signals

Signals can also be classified as those having finite energy or finite average power. However, there are some signals which can neither be classified as energy signals nor power signals. Consider a voltage source $\upsilon(t)$, across a unit resistance R, conducting a current $i(t)$. The instantaneous power dissipated by the resistor is

$$p(t) = \upsilon(t)i(t) = \frac{\upsilon^2(t)}{R} = i^2(t)R$$

Since $R = 1$ ohm, we have

$$p(t) = \upsilon^2(t) = i^2(t) \tag{1.34}$$

The total energy and the average power are defined as the limits

$$E = \lim_{T\to\infty} \int_{-T}^{T} i^2(t)\,dt, \text{ joule} \tag{1.35}$$

and

$$P = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} i^2(t)\,dt, \text{ watt} \tag{1.36}$$

The total energy and the average power normalised to unit resistance of any arbitrary signal $x(t)$ can be defined as

$$E = \lim_{T\to\infty} \int_{-T}^{T} |x(t)|^2\,dt, \text{ joule} \tag{1.37}$$

and

$$P = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} |x(t)|^2\,dt, \text{ watt} \tag{1.38}$$

The ***energy signal*** is one which has finite energy and zero average power, i.e. $x(t)$ is an energy signal if $0 < E < \infty$, and $P = 0$. The ***power signal*** is one which has finite average power and infinite energy, i.e. $0 < P < \infty$, and $E = \infty$. If the signal does not satisfy any of these two conditions, then it is neither an energy nor a power signal.

Example 1.10 *Determine the signal energy and signal power for (a) $f(t) = e^{-3|t|}$ and (b) $f(t) = e^{-3t}$.*

Solution

(a) $$E_\infty = \int_{-\infty}^{\infty} \left[e^{-3|t|}\right]^2 dt = \int_{-\infty}^{0} e^{6t}\,dt + \int_{0}^{\infty} e^{-6t}\,dt = 2\int_{0}^{\infty} e^{-6t}\,dt = -\frac{2}{6}\left[e^{-6t}\right]_{t=0}^{\infty} = -\frac{2}{6}[0-1] = \frac{1}{3}$$

The signal power $P_\infty = 0$ since E_∞ is finite. Hence, the signal $f(t)$ is an energy signal.

(b) $$E_T = \int_{-T}^{T} \left(e^{-3t}\right)^2 dt = \int_{-T}^{T} e^{-6t} dt = -\frac{1}{6}\left[e^{-6T} - e^{6T}\right]$$

As $T \rightarrow \infty$, E_T approaches infinity.

Its average power is

$$P_\infty = \underset{T\to\infty}{Lt} \frac{1}{2T} E_T = \underset{T\to\infty}{Lt} \frac{e^{6T} - e^{-6T}}{12T} = \infty \text{ [Using L'Hospital's rule]}$$

Hence, e^{-3t} is neither an energy signal nor a power signal.

Example 1.11 *Compute the signal energy for $x(t) = e^{-4t}u(t)$.*

Solution Since the value of the given function is zero between $-T$ to 0, the limits must be applied between 0 to T instead of $-T$ to T.

$$E_x = \underset{T\to\infty}{Lt} \int_{-T}^{T} |x(t)|^2 dt$$

$$= \underset{T\to\infty}{Lt} \int_{0}^{T} \left|e^{-4t}\right|^2 dt = \underset{T\to\infty}{Lt} \int_{0}^{T} e^{-8t} dt = \underset{T\to\infty}{Lt} \left[\frac{e^{-8t}}{-8}\right]_0^T = \underset{T\to\infty}{Lt} \left[\frac{e^{-8T}}{-8} + \frac{1}{8}\right] = \frac{1}{8}$$

$$P_x = \underset{T\to\infty}{Lt} \frac{1}{2T} \int_{-T}^{T} |x(t)|^2 dt$$

$$= \underset{T\to\infty}{Lt} \frac{1}{2T} \left[\frac{e^{-8t}}{-8}\right]_{-T}^{T} = \underset{T\to\infty}{Lt} \frac{\left(e^{-8T} - 1\right)}{-16T} = 0$$

Here, as the signal energy E_x is finite and the signal power $P_x = 0$, the signal is an energy signal.

Example 1.12 *Determine whether the following signals are energy signals, power signals, or neither.*

(a) $x(t) = e^{-at}\, u(t),\ a > 0$ (b) $x(t) = A \cos(\omega_0 t + \phi)$ (c) $x(t) = tu(t)$

(d) $x(n) = (-0.5)^n\, u(n)$ (e) $x(n) = 2e^{j3n}$

Solution

(a) $$E = \int_{-\infty}^{\infty} \left|x(t)\right|^2 dt = \int_0^{\infty} e^{-2at} dt = \frac{1}{2a} < \infty$$

Here, the signal has finite energy.

$$P = \frac{1}{T_0}\int_0^{T_0} [x(t)]^2\, dt = \frac{\omega_0}{2\pi}\int_0^{2\pi/\omega_0} e^{-2at}\, dt = \frac{a}{2\pi}\int_0^{2\pi/a} e^{-2at}\, dt = 0$$

Here, the signal has zero average power.
Since the signal has finite energy and zero average power, $x(t)$ is an energy signal.

(b) The sinusoidal signal $x(t)$ is periodic with $T_0 = 2\pi/\omega_0$. All periodic signals have infinite energy. Therefore, only the average power needs to be calculated. The average power of $x(t)$ is

$$P = \frac{1}{T_0}\int_0^{T_0}[x(t)]^2\,dt = \frac{\omega_0}{2\pi}\int_0^{2\pi/\omega_0} A^2\cos^2(\omega_0 t+\phi)\,dt$$

$$= \frac{A^2\omega_0}{2\pi}\int_0^{2\pi/\omega_0}\frac{1}{2}\left[1+\cos(2\omega_0 t+2\phi)\right]dt = \frac{A^2}{2} < \infty$$

Thus, $x(t)$ is a power signal. In general, the periodic signals are power signals.

(c) $$E = \lim_{T\to\infty}\int_{-T/2}^{T/2}\left|x(t)\right|^2 dt = \int_0^{T/2} t^2 dt = \lim_{T\to\infty}\int_0^{T/2} t^2 dt = \lim_{T\to\infty}\frac{(T/2)^3}{3} = \infty$$

$$P = \lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}\left|x(t)\right|^2 dt = \lim_{T\to\infty}\frac{1}{T}\int_0^{T/2} t^2 dt = \lim_{T\to\infty}\frac{1}{T}\frac{(T/2)^3}{3} = \infty$$

Thus, $x(t)$ is neither an energy signal nor a power signal.

(d) By definition,

$$E = \sum_{n=-\infty}^{\infty}|x(n)|^2 = \sum_{n=0}^{\infty}0.25^n = \frac{1}{1-0.25} = \frac{4}{3} < \infty$$

Thus, $x(n)$ is an energy signal.

(e) Since $\left|x(n)\right| = \left|2e^{j3n}\right| = 2\left|e^{j3n}\right| = 2,$

$$P = \lim_{N\to\infty}\frac{1}{2N+1}\sum_{n=-N}^{N}|x(n)|^2 = \lim_{N\to\infty}\frac{1}{2N+1}\sum_{n=-N}^{N}2^2 = \lim_{N\to\infty}\frac{1}{2N+1}4(2N+1) = 4 < \infty$$

Thus, $x(n)$ is a power signal.

Example 1.13 *Determine the power and rms value of the following signals:*

(a) $x_1(t) = 5\cos\left(5t+\frac{\pi}{3}\right)$ and (b) $x_2(t) = 10\cos 5t.\cos 10t$

Solution

(a) $x_1(t) = 5\cos\left(5t+\frac{\pi}{3}\right)$

The fundamental period of $x_1(t)$ is $T_0 = \frac{2\pi}{5}$

Therefore, Power $$P = \frac{1}{T_0}\int_0^{T_0}|x_1(t)|^2\,\mathrm{d}t = \frac{1}{\left(\frac{2\pi}{5}\right)}\int_0^{\frac{2\pi}{5}}\left\{5^2\cos^2\left(5t+\frac{\pi}{3}\right)\right\}\mathrm{d}t$$

$$= \frac{5}{2\pi}\left[25\int_0^{\frac{2\pi}{5}}\left(\frac{1+\cos\left(10t+\frac{2\pi}{3}\right)}{2}\right)\mathrm{d}t\right]$$

$$= \frac{125}{2\pi}\left[\frac{1}{2}t\right]_0^{\frac{2\pi}{5}} + \frac{125}{2\pi}\left[\frac{\sin\left(10t + \frac{2\pi}{3}\right)}{20}\right]_0^{\frac{2\pi}{5}}$$

$$= \frac{125}{2\pi}\left[\frac{\frac{2\pi}{5}}{2}\right] + \frac{125}{2\pi}\left[\frac{\sin\left(\frac{20\pi}{5} + \frac{2\pi}{3}\right) - \sin\left(\frac{2\pi}{3}\right)}{20}\right]$$

$$= 12.5 + \frac{125}{2\pi}[0] = 12.5 \text{ watts}$$

$$P_{rms} = \sqrt{P} = \sqrt{12.5} = 3.535 \text{ watts}$$

(b) $x_2(t) = 10 \cos 5t \,.\, \cos 10t = 10 \cos 15t - 10 \cos 5t$

The fundamental period of the signal $x(t)$ is

$$T_1 = \frac{2\pi}{15} \text{ and } T_2 = \frac{2\pi}{5}$$

Therefore, $\dfrac{T_1}{T_2} = \dfrac{\frac{2\pi}{15}}{\frac{2\pi}{5}} = \dfrac{1}{3}$

$$T_0 = 3T_1 \text{ or } T_2$$

Hence $\quad T_0 = \dfrac{2\pi}{5}$

Power, $\quad P = \dfrac{1}{\left(\frac{2\pi}{5}\right)} \displaystyle\int_0^{\frac{2\pi}{5}} (10 \cos 15t - 10 \cos 5t)^2 \, dt$

$$= \frac{500}{2\pi} \int_0^{\frac{2\pi}{5}} \left[\cos^2 15t - \cos^2 5t - \cos 15t \cos 5t\right] dt$$

$$= \frac{500}{2\pi} \int_0^{\frac{2\pi}{5}} \left\{\left(\frac{1 + \cos 30t}{2}\right) + \left[\frac{1 + \cos 10t}{2}\right] - 2\left[\frac{\cos 20t - \cos 10t}{2}\right]\right\} dt$$

$$= \frac{500}{2\pi}\left[\frac{1}{2}t\right]_0^{\frac{2\pi}{5}} + \frac{500}{2\pi}\left[\frac{1}{2}t\right]_0^{\frac{2\pi}{5}} + 0$$

$$= \frac{500}{2\pi}\left[\frac{2\pi}{10}\right] + \frac{500}{2\pi}\left[\frac{2\pi}{10}\right] = 100 \text{ watts}$$

Hence, $\quad P_{rms} = \sqrt{P} = \sqrt{100} = 10 \text{ watts}$

Example 1.14 *Determine whether* $x(t) = \text{rect}\left(\frac{t}{10}\right)\cos\omega_0 t$ *is an energy signal or a power signal.*

Solution

(i) rect (t) is sketched in Fig. E1.14(a).

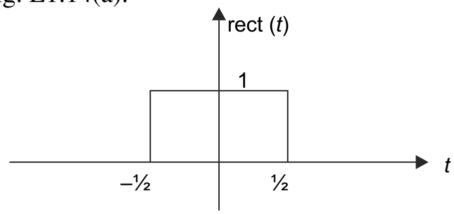


Fig. E1.14(a)

(ii) $\text{rect}\left(\frac{t}{10}\right)$ is sketched in Fig. E1.14(b).

(iii) $\text{rect}\left(\frac{t}{10}\right)\cos\omega_0 t$ is sketched in Fig. E1.14(c).

$$
\begin{aligned}
E &= \int_{-\infty}^{\infty} |\,x(t)\,|^2\, dt \\
&= \int_{-5}^{5} \left|\cos^2 \omega_0 t\right| dt \\
&= \int_{-5}^{5} \left(\frac{1+\cos 2\omega_0 t}{2}\right) dt \\
&= \int_{-5}^{5} \frac{1}{2}\, dt + \int_{-5}^{5} \frac{\cos 2\omega_0 t}{2}\, dt \\
&= \frac{1}{2}[\,t\,]_{-5}^{5} + \left[\frac{\sin 2\omega_0 t}{4}\right]_{-5}^{5} \\
&= 0 + \frac{1}{4}\left[\sin 10\omega_0 + \sin 10\omega_0\right] \\
&= \frac{1}{2}\sin 10\omega_0, \text{ which is less than } \infty
\end{aligned}
$$

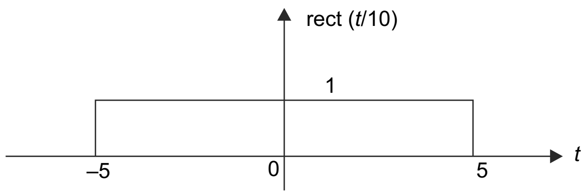


Fig. E1.14(b)

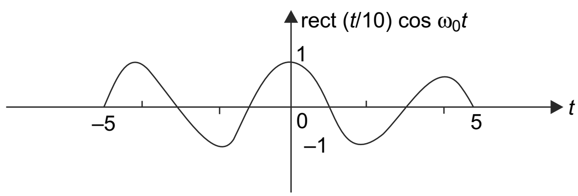


Fig. E1.14(c)

$$P = \lim_{T\to\infty} \frac{1}{T}\int_{-5}^{5} \cos^2 \omega_0 t\, dt = \frac{1}{T}\left(\frac{1}{4}\sin 5\omega_0\right) = 0$$

Since the signal has finite energy and zero average power, $x(t)$ is an energy signal.

1.3.6 Power Density Spectrum of Periodic Signals

The average power of a discrete-time signal with period N is given by

$$P_x = \frac{1}{N}\sum_{n=0}^{N-1} |\,x(n)\,|^2 \tag{1.39}$$

To derive an expression for P_x in terms of the Fourier coefficient $X(k)$, the following two equations can be used for synthesis and analysis of discrete-time periodic signals.

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

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

Therefore,

$$P_x = \frac{1}{N} \sum_{n=0}^{N-1} x(n) x^*(n) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) \left(\sum_{k=0}^{N-1} X(k) e^{-j2\pi kn/N} \right)$$

$$= \sum_{k=0}^{N-1} X(k) \left[\frac{1}{N} \sum_{n=0}^{N-1} x(k) e^{-j2\pi kn/N} \right] = \sum_{k=0}^{N-1} |X(k)|^2 = \frac{1}{N} \sum_{n=0}^{N-1} |x(n)|^2$$

which is a *Parseval's relation* for discrete-time periodic signals. The average power in the signal is the sum of the powers of the individual frequency components.

1.3.7 Energy Density Spectrum of Aperiodic Signals

We know that the energy of a discrete-time signal $x(n)$ is defined as

$$E_x = \sum_{n=-\infty}^{\infty} |x(n)|^2 \tag{1.40}$$

Expressing the energy E_x in terms of the spectral characteristic $X(\omega)$, we have

$$E_x = \sum_{n=-\infty}^{\infty} x(n) x^*(n) = \sum_{n=-\infty}^{\infty} x(n) \left[\frac{1}{2\pi} \int_{-\pi}^{\pi} X^*(\omega) e^{-j\omega n} \mathrm{d}\omega \right]$$

$$= \frac{1}{2\pi} \int_{-\pi}^{\pi} X^*(\omega) \left[\sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n} \right] \mathrm{d}\omega = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(\omega)|^2 \, \mathrm{d}\omega$$

Therefore, the energy relation between $x(n)$ and $X(\omega)$ is

$$E = \sum_{n=-\infty}^{\infty} |x(n)|^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(\omega)|^2 \, \mathrm{d}\omega$$

This is Parseval's relation for discrete-time aperiodic signals with finite energy.

The spectrum $X(\omega)$ may be a complex-valued function of frequency. Hence

$$X(\omega) = |X(\omega)| e^{j\phi(\omega)}$$

where $|X(\omega)|$ is the magnitude spectrum and $\phi(\omega) = \angle X(\omega)$ is the phase spectrum.

For continuous-time signals,

$$S_{xx}(\omega) = |X(\omega)|^2 \tag{1.41}$$

represents the distribution of energy as a function of frequency, and it is called the *energy density spectrum* of $x(n)$. It is clear that $S_{xx}(\omega)$ does not have any phase information.

If the signal $x(n)$ is real, then

$$X^*(\omega) = X(-\omega)$$

or equivalently,

$$|X(-\omega)| = |X(\omega)| \qquad \text{(even symmetry)}$$

and

$$\angle X(-\omega) = -\angle X(\omega) \qquad \text{(odd symmetry)}$$

Therefore, $S_{xx}(-\omega) = S_{xx}(\omega)$ (even symmetry)

1.3.8 Singularity Functions

Singularity functions are an important classification of non-periodic signals. They can be used to represent more complicated signals. The unit-impulse function, sometimes referred to as delta function, is the basic singularity function and all other singularity functions can be derived by repeated integration or differentiation of the delta function. The other commonly used singularity functions are the unit-step and unit-ramp functions.

The advantage of the singularity function is that any arbitrary signal that is made up of straight-line segments can be represented in terms of step and ramp functions.

SIMPLE MANIPULATIONS OF DISCRETE-TIME SIGNALS 1.4

When a signal is processed, the signal undergoes many manipulations involving both the independent and the dependent variable. Some of these manipulations include (i) shifting the signal in the time domain, (ii) folding the signal, and (iii) scaling in the time-domain. A brief introduction of these manipulations here, will help the reader in the following chapters.

1.4.1 Transformation of the Independent Variable

Shifting In the case of discrete-time signals, the independent variable is the time, n. A signal $x(n)$ may be shifted in time, i.e., the signal can be either advanced in the time axis or delayed in the time axis. The shifted signal is represented by $x(n-k)$, where k is an integer. If 'k' is positive, the signal is delayed by k units of time and if k is negative, the time shift results in an advance of signal by k units of time. However, advancing the signal in the time axis is not possible always. If the signal is available in a magnetic disk or other storage units, then the signal can be delayed or advanced as one wishes. But in real time, advancing a signal is not possible since such an operation involves samples that have not been generated. As a result, in real-time signal processing applications, the operation of advancing the time base of the signal is physically unrealisable.

Folding This operation is done by replacing the independent variable n by $-n$. This results in folding of the signal about the origin, i.e. $n = 0$. Folding is also known as the reflection of the signal about the time origin $n = 0$. Folding of a signal is done while convoluting the signal with another.

Time Scaling This involves replacing the independent variable n by kn, where k is an integer. This process is also called as *down sampling*. If $x(n)$ is the discrete-time signal obtained by sampling the analog signal, $x(t)$, then $x(n) = x(nT)$, where T is the sampling period. If time-scaling is done, then the time-scaled signal, $y[n] = x(kn) = x(knT)$. This implies that the sampling rate is changed from $1/T$ to $1/kT$. This decreases the sampling rate by a factor of k. Down-sampling operations are discussed in detail in Chapter 11 of this book. The folding and time scaling operations are shown in Figs. 1.11(a) and (b).

1.4.2 Representation of Signals

In the signal given by $x(at + b)$, i.e., $x(a(t + b/a))$, a is a scaling factor and b/a is a pure shift version in the time domain.

If b/a is positive, then the signal $x(t)$ is shifted to left.
If b/a is negative, then the signal $x(t)$ is shifted to right.
If a is positive, then the signal $x(t)$ will have positive slope.
If a is negative, then the signal $x(t)$ will have negative slope.
If a is less than 0, then the signal $x(t)$ is reflected or reversed through the origin.
If $|a| < 1$, $x(t)$ is expanded, and if $|a| > 1$, $x(t)$ is compressed.

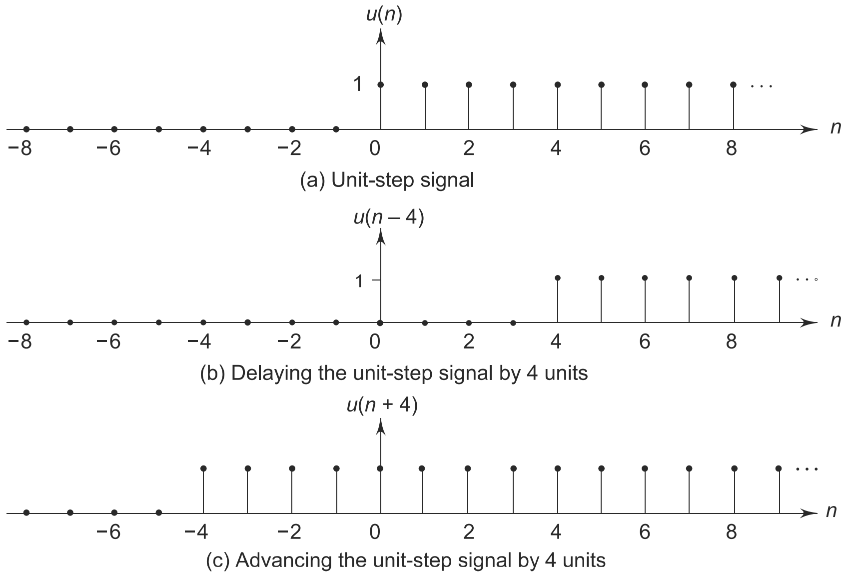


Fig. 1.10 *Graphical Representation of a Signal, and its Delayed and Advanced Versions*

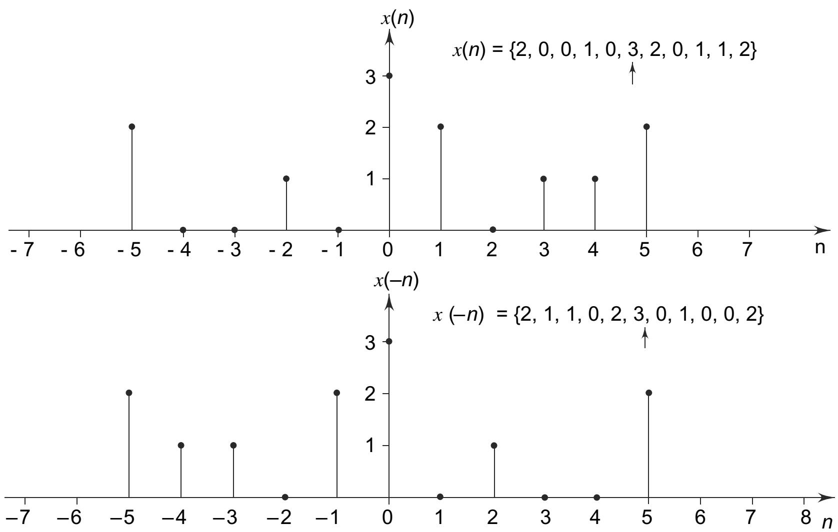


Fig. 1.11(a) *Illustrations of Folding*

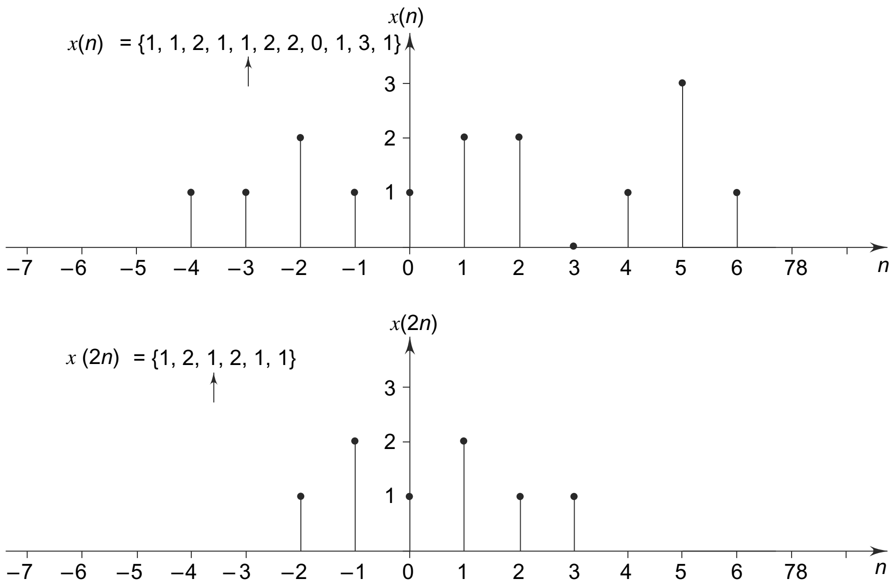


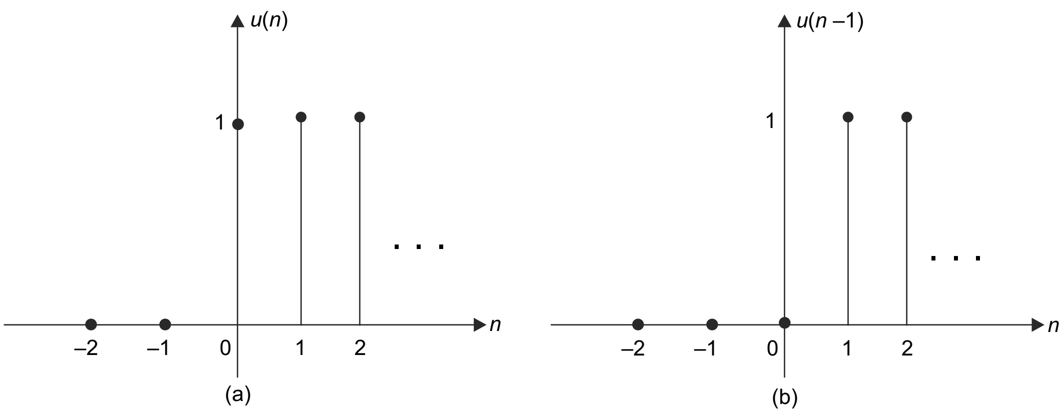
Fig. 1.11(b) *Illustrations of Time Scaling Operations*

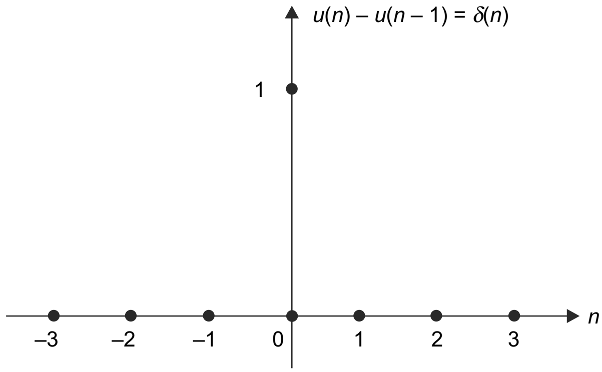
Example 1.15 *Draw the graphical representation of unit step sequence u(n) and shifted unit step sequence u(n – 1) and sketch the signal.*

$$x(n) = u(n) - u(n-1)$$

Solution The graphical representation of unit step sequence $u(n)$ and shifted unit step sequence $u(n-1)$ are shown in Fig. E1.15(a) and Fig. E1.15(b) respectively.

The resultant signal $x(n) = u(n) - u(n-1)$ is obtained by adding both the signals as shown in Fig. E.1.15(c)



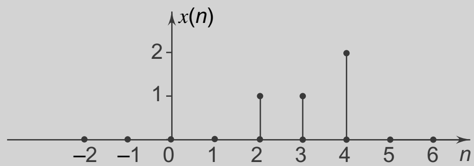


(c)

Fig. E1.15

Example 1.16 *A discrete-time signal x(n) is shown in Fig. E1.16(a). Sketch and label each of the following signals.*

(a) $x(2n)$ (b) $x(-n)$ (c) $x(n + 2)$ (d) $x(n - 2)$ and (e) $x(-n + 2)$



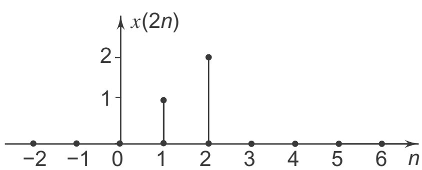
(a)

Fig. E1.16

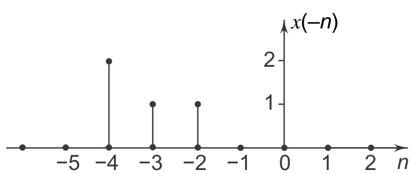
Solution

(a) $x(2n)$ is sketched in Fig. E1.16(b). Then $x(2n) = (0, 0, \underset{\uparrow}{0}, 1, 2, 0, 0, ...)$

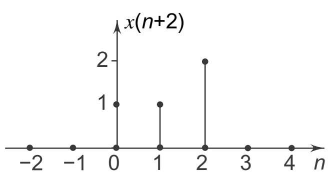
(b) $x(-n)$ is sketched in Fig. E1.16(c). Then $x(-n) = (0, 2, 1, 1, 0, 0, 0, 0, ...)$



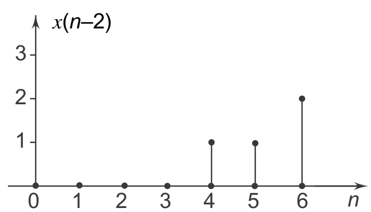
(b)



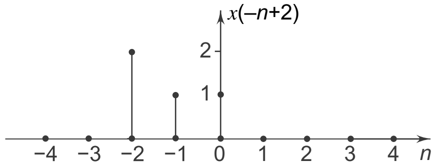
(c)



(d)



(e)



(f)

Fig. E1.16

(c) $x(n+2)$ is sketched in Fig. E1.16(d). Then $x(n+2) = (0, 0, \underset{\uparrow}{1}, 1, 2, 0, 0)$

(d) $x(n-2)$ is sketched in Fig. E1.16(e). Then $x(n-2) = (\underset{\uparrow}{0}, 0, 0, 0, 1, 1, 2)$

(e) $x(-n+2)$ is sketched in Fig. E1.16(f). Then $x(-n+2) = (0, 0, 2, 1, \underset{\uparrow}{1}, 0, 0, 0, 0)$

Example 1.17 *If* $x(n) = \{0, 2, -1, 0, \underset{\uparrow}{2}, 1, 1, 0, -1\}$, *what are (a) x(n – 3), (b) x(–n), and (c) x(1 – n)?*

Solution $x(n) = \{0, 2, -1, 0, 2, 1, 1, 0, -1\}$ is sketched as shown in Fig. E1.17(a).

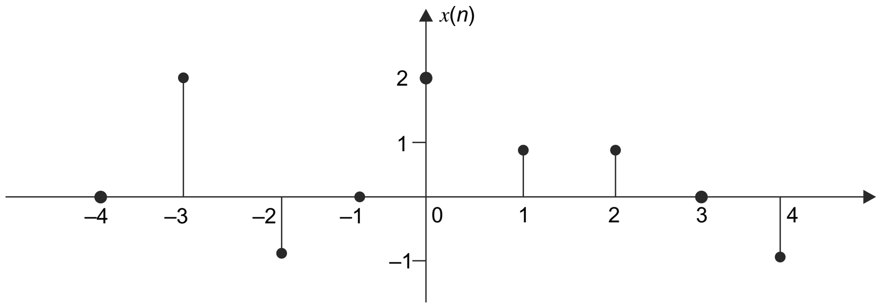


Fig. E1.17(a)

(a) $x(n-3)$ is sketched as shown in Fig. E1.17(b).

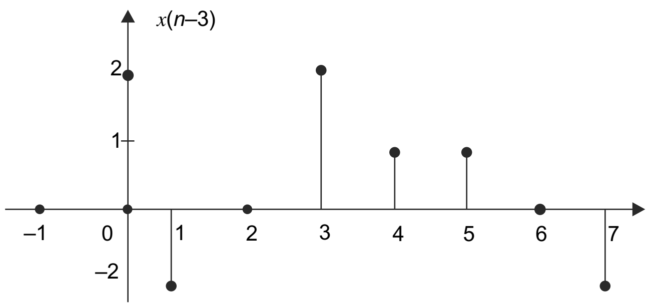


Fig. E1.17(b)

Then, $x(n-3) = \{0, \underset{\uparrow}{2}, -1, 0, 2, 1, 1, 0, -1\}$

(b) $x(-n)$ is sketched in Fig. E1.17(c).

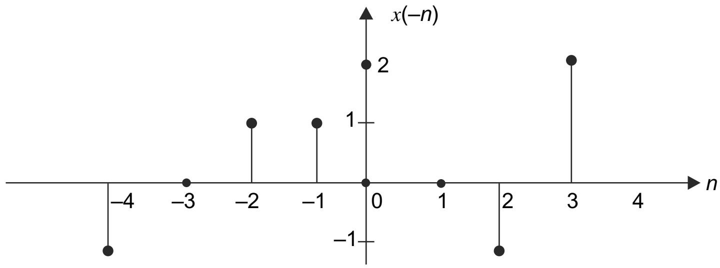


Fig. E1.17(c)

Then, $x(-n) = \{-1, 0, 1, 1, 2, 0, -1, 2, 0\}$
$\uparrow$

(c) $x(1-n)$ is sketched in Fig. E1.17(d).

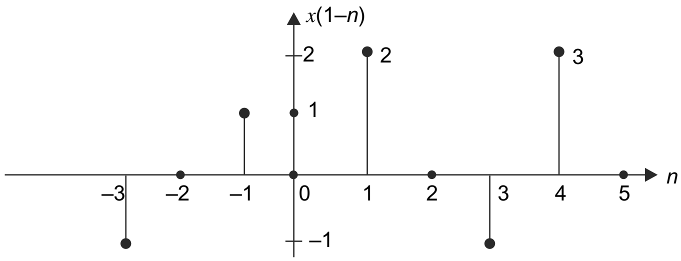


Fig. E1.17(c)

Then, $x(1-n) = \{-1, 0, 1, 1, 2, 0, -1, 2, 0\}$
$\uparrow$

Example 1.18 *Using the discrete-time signals $x_1(n)$ and $x_2(n)$ shown in Fig. E1.13 (a) and (b) represent each of the following signals by sketches with sequence of numbers.*

(a) $y_1(n) = x_1(n) + x_2(n)$ (b) $y_2(n) = x_1(n)x_2(n)$

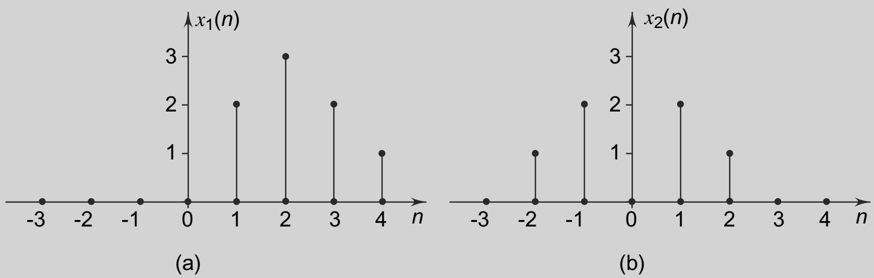
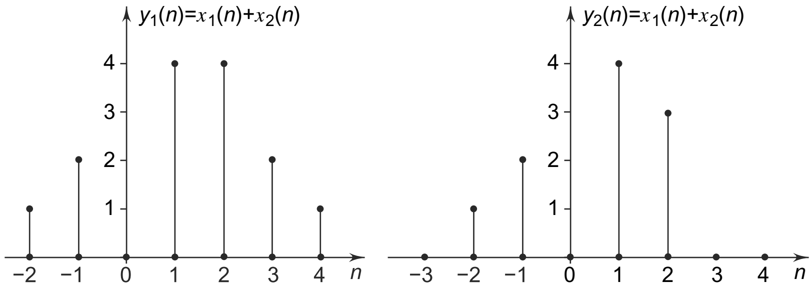


Fig. E1.18

Solution

(a) $y_1(n)$ is sketched in Fig. E1.18 (c). Then $y_1(n) = \{0, 1, 2, 4, 4, 2, 1\}$

(b) $y_2(n)$ is sketched in Fig. E1.18 (d). Then $y_2(n) = \{0, 0, 0, 3, 4, 3, 0, 0\}$



(c) (d)

Fig. E1.18

Example 1.19 *Sketch the following signals.*

(a) $x(t) = \Pi(2t + 3)$ (b) $x(t) = 2\Pi(t - 1/4)$ (c) $x(t) = \cos(20\pi t - 5\pi)$
(d) $x(t) = r(-0.5t + 2)$ (e) $x(t) = -2r(t)$

Solution

(a) $\Pi(2t + 3) = \Pi(2(t + 3/2))$
Here, the signal shown in Fig. E1.19(a) is shifted to left, with centre at −3/2. Since $a = 2$, i.e. $|a| > 1$, the signal is compressed. The signal width becomes 1/2 with unity amplitude.

(b) $x(t) = 2\Pi(t - 1/4)$
Here, the signal shown in Fig. E1.19(b) is shifted to the right, with centre at 1/4. Since $a = 1$, the signal width is 1 and amplitude is 2.

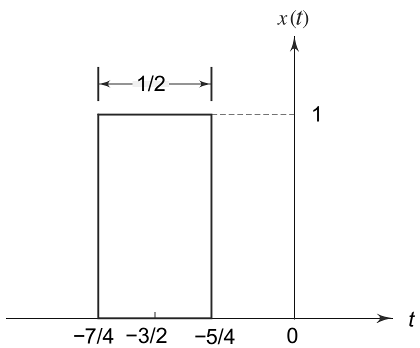


Fig. E1.19(a)

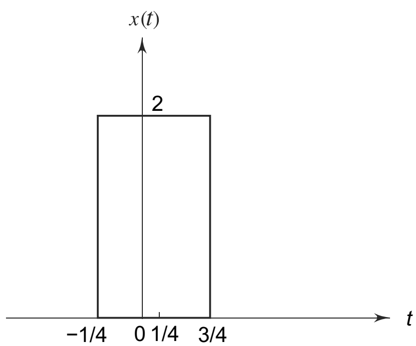


Fig. E1.19(b)

(c) $x(t) = \cos(20\pi t - 5\pi) = \cos\left(20\pi\left(t - \frac{5\pi}{20\pi}\right)\right) = \cos\left(20\pi\left(t - \frac{1}{4}\right)\right)$

Here the signal $x(t)$ shown in Fig. E1.19(c) is shifted by quarter cycle to the right.

(d) $x(t) = r(-0.5t + 2)$

$$= r\left(-0.5\left(t - \frac{2}{0.5}\right)\right) = r(-0.5(t-4))$$

The given ramp signal is reflected through the origin and shifted to right at $t = 4$.

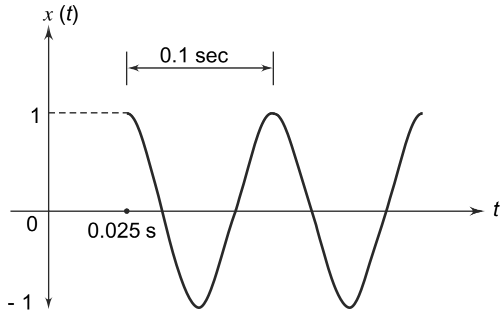


Fig. E1.19(c)

The signal is expanded by $\frac{1}{0.5} = 2$. When $t = 0$, the magnitude of the signal $x(t) = 2$, shown in Fig. E1.19(d).

(e) $x(t) = -2r(t)$

The given $r(t)$ is amplitude scaled by a factor of –2 as shown in Fig. E1.19 (e).

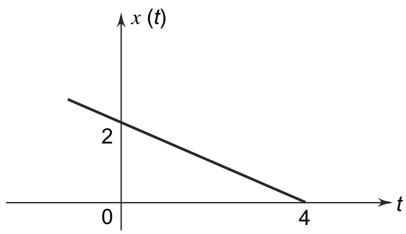


Fig. E1.19(d)

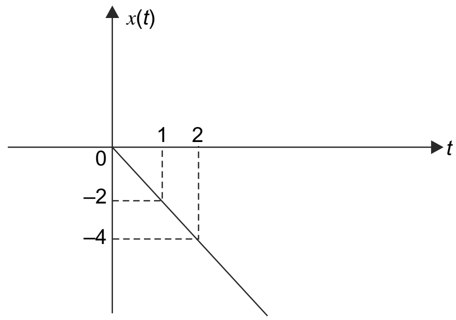


Fig. E1.19(e)

Example 1.20 *Write down the corresponding equation for the signal shown in Fig. E1.20.*

Solution Representation through addition of two unit step functions

The signal $x(t)$ can be obtained by adding both the pulses, i.e.

$$x(t) = 2[u(t) - u(t-2)] + [u(t-3) - u(t-5)]$$

Representation through multiplication of two unit step functions

$$x(t) = 2[u(t)\,u(-t+2)] + [u(t-3)\,u(-t+5)]$$

$$= 2(u(t)\,u(2-t) + u(t-3)\,u(5-t))$$

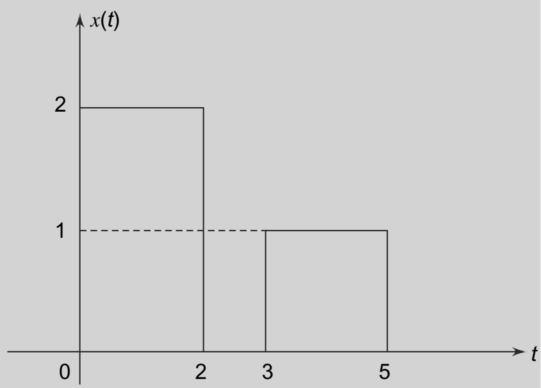


Fig. E1.20

Example 1.21 *For the signal x(t) shown in Fig. E1.21(a), sketch x(2t + 3).*

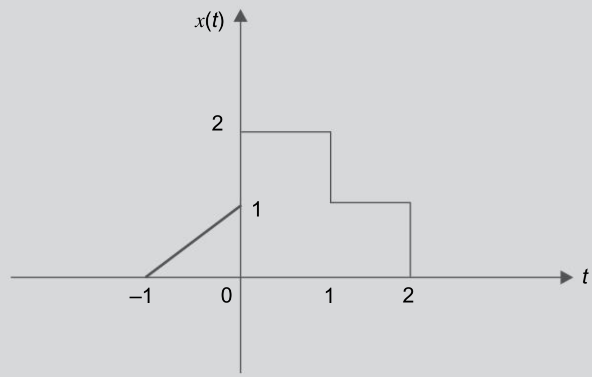


Fig. E1.21(a)

Solution The signal $x(2t + 3)$ can be obtained by first shifting $x(t)$ to the left by 3 units and then scaling by 2 units as shown in Figs. E.1.21(b) and E. 1.21(c).

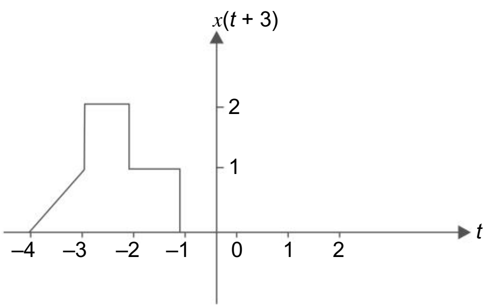


Fig. E1.21(b)

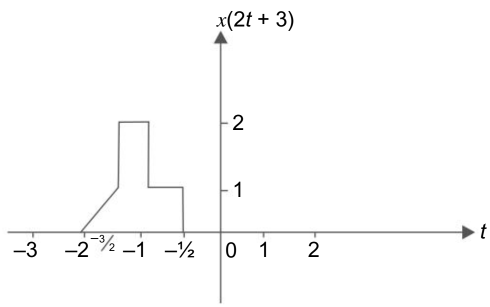


Fig. E1.21(c)

AMPLITUDE AND PHASE SPECTRA 1.5

Let us consider a cosine signal of peak amplitude A, frequency f and phase shift Φ, in order to introduce the concept of amplitude and phase spectra, i.e.,

$$x(t) = A\cos(2\pi ft + \Phi)$$

The amplitude and phase of this signal can be plotted as a function of frequency. The amplitude of the signal as a function of frequency is referred to as *amplitude spectrum* and the phase of the signal as a function of frequency is referred to as *phase spectrum* of the signal. The amplitude and phase spectra together is called the *frequency spectrum* of the signal. The units of the amplitude spectrum depends on the signal. For example, the unit of the amplitude spectrum of a voltage signal is measured in volts, and the unit of the amplitude spectrum of a current signal is measured in amperes. The unit of the phase spectrum is usually radians. The frequency spectrum drawn for positive values of frequencies alone is called a *single-sided spectrum*.

The cosine signal can also be expressed in phasor form as the sum of the two counter rotating phasors with complex-conjugate magnitudes, i.e.

$$x(t) = A\left[\frac{e^{j(2\pi ft+\phi)} + e^{-j(2\pi ft+\phi)}}{2}\right] \tag{1.42}$$

From this, the amplitude spectrum for the signal $x(t)$ consists of two components of amplitude, viz. $A/2$ at frequency 'f' and $A/2$ at frequency '$-f$'. Similarly, the phase spectrum also consists of two phase components one at 'f' and the other at '$-f$'. The frequency spectrum of the signal, in this case, is called a *double-sided spectrum*. The following example illustrates the single-sided and double-sided frequency spectra of a signal.

Example 1.22 *Sketch the single-sided and double-sided amplitude and phase spectra of the signal.*

$$x(t) = 8\sin\left(20\pi t - \frac{\pi}{6}\right), \quad -\infty < t < \infty$$

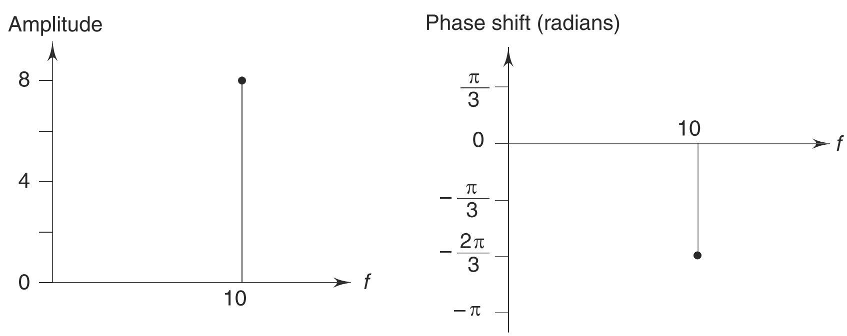
Solution The single-sided spectra is plotted by expressing $x(t)$ as the real part of the rotating phasor. Using the trigonometric identity, $\cos\left(u - \frac{\pi}{2}\right) = \sin u$, the given signal is converted into a form as in Eq. (1.27), i.e.

$$x(t) = 8\sin\left(20\pi t - \frac{\pi}{6}\right) = 8\cos\left(20\pi t - \frac{\pi}{6} - \frac{\pi}{2}\right)$$

$$= 8\cos\left(20\pi t - \frac{2\pi}{3}\right)$$

The single-sided amplitude and phase spectra are shown in Fig. E1.22(a). The signal has an amplitude of 8 units at $f = 10$ Hz and a phase angle of $-\frac{2\pi}{3}$ radians at $f = 10$ Hz. To plot the double-sided spectrum, the signal is converted into the form as in Eq. (1.42). Therefore,

$$x(t) = 4e^{j\left(20\pi t - \frac{2x}{2}\right)} + 4e^{-j\left(20\pi t - \frac{2x}{2}\right)}$$



(a)

The double-sided amplitude and phase spectra are shown in Fig. E1.22(b). The signal has two components at $f = 10$ Hz and $f = -10$ Hz. The amplitude of these components are 4 units each and the phase of these components are $-\frac{2\pi}{3}$ and $\frac{2\pi}{3}$ and radians, respectively.

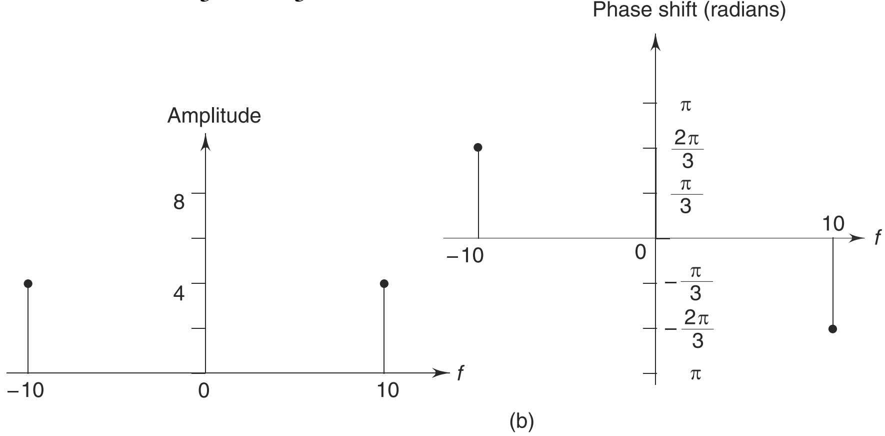


Fig. E1.22 *Amplitude and Phase Spectra (a) Single-sided, (b) Double-sided*

CLASSIFICATION OF SYSTEMS 1.6

As with signals, systems are also broadly classified into continuous-time and discrete-time systems. In a continuous-time system, the associated signals are also continuous, i.e. the input and output of the system are both continuous-time signals. On the other hand, a discrete-time system handles discrete-time signals. Here, both the input and output signals are discrete-time signals.

Both continuous and discrete-time systems are further classified into the following types.

(i) Static and dynamic systems
(ii) Linear and non-linear systems
(iii) Time-variant and time-invariant systems
(iv) Causal and non-causal systems
(v) Stable and unstable systems, and
(vi) Invertible systems

1.6.1 Static and Dynamic Systems

The output of a static system at any specific time depends on the input at that particular time. It does not depend on past or future values of the input. Hence, a static system can be considered as a system with no memory or energy storage elements. A simple resistive network is an example of a static system. The input/output relation of such systems does not involve integrals or derivatives.

The output of a dynamic system, on the other hand at any specified time depends on the inputs at that specific time and at other times. Such systems have memory or energy storage elements. The equation characterising a dynamic system will always be a differential equation for continuous-time system or a difference equation for a discrete-time system. Any electrical circuit consisting of a capacitor or an inductor is an example of a dynamic system. The following equations characterise dynamic systems.

(i) $\frac{d^2y(t)}{dt^2} + \frac{dy(t)}{dt} + 3y(t) = \frac{dx(t)}{dt} + x(t)$

(ii) $y(n-1) + 2y(n) = 4x(n) - x(n-1)$

1.6.2 Linear and Non-linear Systems

A linear system is one in which the principle of superposition holds true (see Fig. 1.12). For a system with two inputs $x_1(t)$ and $x_2(t)$, the superposition is defined as follows.

$$H[a_1x_1(t) + a_2x_2(t)] = a_1H[x_1(t)] + a_2H[x_2(t)] \quad (1.43)$$

where, a_1 and a_2 are the weights added to the inputs, and $H[x(t)] = y(t)$ is the response of the continuous-time system to the input $x(t)$. Thus, a *linear system* is defined as one whose response to the sum of the weighted inputs is same as the sum of the weighted responses. If a system does not satisfy Eq. (1.43), then the system is *non-linear*. For a discrete-time system, the condition for linearity is given by Eq. (1.44).

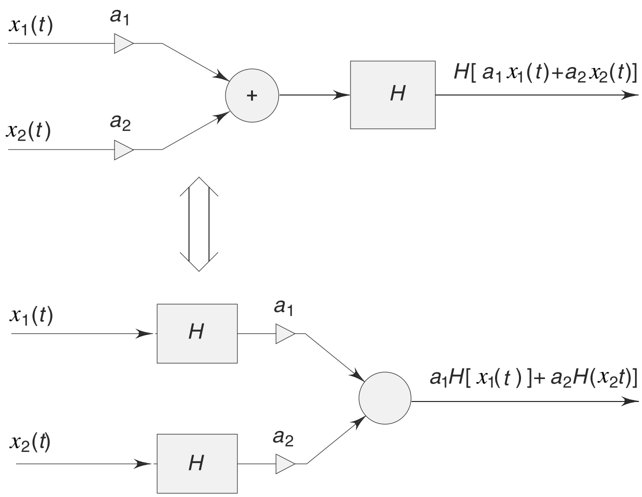


Fig. 1.12 *Illustration of the Superposition Principle*

$$H[a_1x_1(n) + a_2x_2(n)] = a_1H[x_1(n)] + a_2H[x_2(n)] \quad (1.44)$$

where $H[x(n)] = y(n)$ is the response of the discrete-time system to the input $x(n)$.

Example 1.23 *Determine whether the system described by the differential equation* $\frac{dy(t)}{dt} + 2y(t) = x(t)$ *is linear.*

Solution Let the response of the system to $x_1(t)$ be $y_1(t)$ and the response of the system to $x_2(t)$ be $y_2(t)$. Thus, for the input $x_1(t)$, the describing equation is

$$\frac{dy_1(t)}{dt} + 2y_1(t) = x_1(t)$$

and for the input $x_2(t)$,

$$\frac{dy_2(t)}{dt} + 2y_2(t) = x_2(t)$$

Multiplying these equations by a_1 and a_2, respectively, and adding yields,

$$a_1\frac{dy_1(t)}{dt} + a_2\frac{dy_2(t)}{dt} + 2a_1y_1(t) + 2a_2y_2(t) = a_1x_1(t) + a_2x_2(t)$$

i.e.

$$\frac{d}{dt}(a_1y_1(t) + a_2y_2(t)) + 2(a_1y_1(t) + a_2y_2(t)) = a_1x_1(t) + a_2x_2(t)$$

The response of the system to the input $a_1x_1(t) + a_2x_2(t)$ is $a_1y_1(t) + a_2y_2(t)$. Thus, the superposition condition is satisfied and hence the system is linear.

Example 1.24 *Determine whether the system described by the differential equation* $\frac{dy(t)}{dt} + y(t) + 4 = x(t)$ *is linear.*

Solution Let the response of the system to $x_1(t)$ be $y_1(t)$ and the response of the system to $x_2(t)$ be $y_2(t)$. Thus, for input $x_1(t)$, the describing equation is

$$\frac{dy_1(t)}{dt} + y_1(t) + 4 = x_1(t)$$

and for input $x_2(t)$,

$$\text{i.e.} \quad \frac{dy_2(t)}{dt} + y_2(t) + 4 = x_2(t)$$

Multiplying these equations by a_1 and a_2, respectively, and adding yields,

$$\frac{d}{dt}(a_1 y_1(t) + a_2 y_2(t)) + (a_1 y_1(t) + a_2 y_2(t)) + 4(a_1 + a_2) = a_1 x_1(t) + a_2 x_2(t)$$

This equation cannot be put into the same form as the original differential equation describing the system. Hence, the system is non-linear.

1.6.3 Time-variant and Time-invariant Systems

A time-invariant system is one whose input-output relationship does not vary with time. A time-invariant system is also called a *fixed system*. The condition for a system to be fixed is

$$H[x(t - \tau)] = y(t - \tau) \tag{1.45}$$

A time-invariant system satisfies Eq. (1.45) for any $x(t)$ and any value of τ. Equation (1.45) states that if $y(t)$ is the response of the system to any input $x(t)$, then the response of the system to the time-shifted input is the response of the system to $x(t)$ time shifted by the same amount. In discrete time, this property is also referred to as shift-invariance. For a discrete-time system, the condition for shift-invariance is given by

$$H[x(n - k)] = y(n - k) \tag{1.46}$$

where k is an integer. A system not satisfying either Eq. (1.45) or Eq. (1.46) is said to be time-variant. The systems satisfying both linearity and time-invariant conditions are called ***linear, time-invariant*** systems, or simply ***LTI systems.***

1.6.4 Causal and Non-causal Systems

A causal system is non-anticipatory. The response of the causal system to an input does not depend on future values of that input, but depends only on the present and/or past values of the input. If the response of the system to an input depends on the future values of that input, then the system is non-causal or anticipatory. Non-causal systems are unrealisable. The following difference equations describe causal systems.

(i) $y(n) = 0.5\, x(n) - x(n - 2)$
(ii) $y(n) = x(n)$
(iii) $y(n - 2) + y(n) = x(n) + 0.98\, x(n - 1)$

The following equations describe non-causal systems.

(i) $y(n - 1) = x(n)$
(ii) $y(n) = 0.11x(n - 1) + x(n) - 0.8\, x(n + 1)$

1.6.5 Stable and Unstable Systems

A system is said to be bounded-input, bounded-output (BIBO) stable, if every bounded input produces a bounded output. A bounded signal has an amplitude that remains finite. Thus, a BIBO stable system will have a bounded output for any bounded input so that its output does not grow unreasonably large. The conditions for a system to be BIBO stable are given as follows.

(i) If the system transfer function is a rational function, the degree of the numerator must be no larger than the degree of the denominator.

(ii) The poles of the system must lie in the left half of the s-plane or within the unit circle in the z-plane.

(iii) If a pole lies on the imaginary axis, it must be a single-order one, i.e. no repeated poles must lie on the imaginary axis.

The systems not satisfying the above conditions are unstable.

It is to be noted that there are several misconceptions in finding whether the system is stable or unstable. It is the amplitude of the input and output signals that must be finite for stable systems. In this context, BIBO stability can be applied for any type of system.

1.6.6 Invertible System

A system is said to be invertible, if a distinct input produces a distinct output. For two inputs $x_1(n)$ and $x_2(n)$ with $x_1(n) \neq x_2(n)$, an invertible system produces outputs $y_1(n)$ and $y_2(n)$ such that $y_1(n) \neq y_2(n)$. For every invertible system, there exists an inverse system.

A discrete linear time-invariant system with impulse response $h(n)$ will be invertible, if and only if $h(n)*h^{-1}(n) = \delta(n)$ is satisfied. Here, $h^{-1}(n)$ is the impulse response of the inverse system.

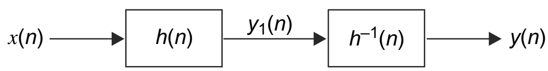


Fig. 1.13 *Cascade of a system and its inverse*

Let us consider two LTI systems, interconnected as shown in Fig. 1.13. Then, the overall response of the system is given by

$$y(n) = y_1(n)* h^{-1}(n) = x(n)* h(n)* h^{-1}(n)$$

Using associative property, we can write

$$y(n) = x(n)* [h(n)* h^{-1}(n)]$$

If $h(n)* h^{-1}(n) = \delta(n)$, then

$$y(n) = x(n)* \delta(n) = x(n)$$

That is, the input is recovered from the output.

Similarly, for a continuous-time LTI system to be invertible, it must satisfy the following condition, namely $h(t)* h^{-1}(t) = \delta(t)$.

Example 1.25 *Determine whether the following systems are invertible.*

(a) $y(n) = Ax(n), A \neq 0$ (b) $y(n) = x(n) - x(n-1)$

Solution

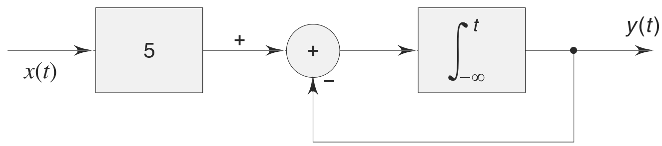
(a) The system is invertible because given the output $y(n)$, it is possible to recover the input using the inverse system $x(n) = \frac{1}{A} y(n)$

(b) The output $y(n)$ can be obtained for two different inputs namely $x(n)$ and $x(n) + c$ where c is a constant. Hence, the system is not invertible.

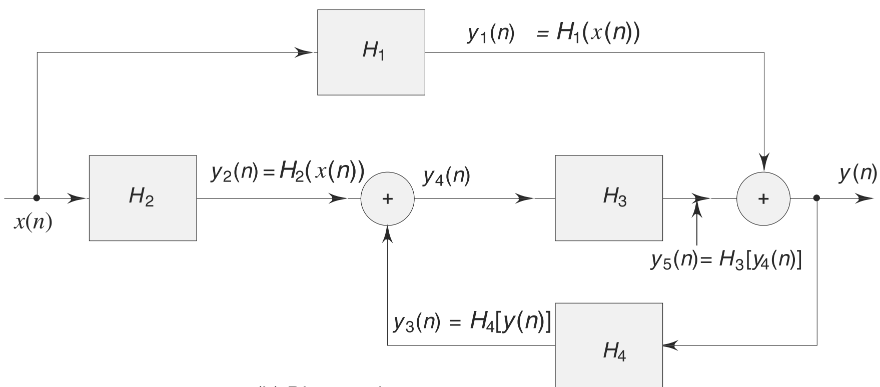
REPRESENTATIONS OF SYSTEMS 1.7

The representation of a system helps in visualising the system with its components and their interconnections. A system can be represented using a diagram featuring various components of the system. These components are represented by symbols. An electrical system is thus represented by a diagram consisting of different symbols representing resistors, capacitors or any other device. A mechanical system can be represented using symbols for different elements like damper, acceleration, zero friction, etc. Thus, a system can be visualised when represented in the form of a diagram.

A more convenient form of representing a system is the block-diagram representation. In this form of representation, each box is an operator on the input signal and the operation is shown on the box itself. Some typical operations include integration, differentiation, scalar multiplication, delay, etc. The lines connecting the individual boxes are directional and these lines show the direction of the signal flow. A number of lines may terminate at a node. The node may be either an accumulator or a multiplier. These nodes are represented by circles with the symbols '+' or '×' marked on them. Figure 1.14 shows the block diagram representation of continuous-time and discrete-time systems.



(a) Continuous-time



(b) Discrete-time

Fig. 1.14 *Block Diagram Representation of (a) Continuous-Time System, (b) Discrete-Time System*

A system can also be represented using mathematical models. The analysis of system characteristics and performance can be carried out using the mathematical model of a system. The mathematical model of a system consists of equations relating to the signals of interest. Both discrete-time and continuous-time signals can be modelled using the following three methods.

(i) A linear difference/differential equation
(ii) The impulse-response sequence
(iii) A state-variable or matrix description.

All the above three methods help in determining the output of the system from the knowledge of the input to the system. A system can be interpreted in different ways depending on the model since each model emphasises certain aspects of the system. Together, these models provide a very good understanding of the system and how the system works. In the following sections, these models are discussed further.

1.7.1 Linear Difference/Differential Equations

A discrete-time system is modelled by a difference equation, whereas a continuous-time system is modelled by a differential equation. In a linear discrete-time system, the input sequence $\{x_n\}$ is transformed into an output sequence $\{y_n\}$ according to some difference equation. For example,

$$y(n) = x(n) + 3x(n-1) + 2x(n-2) \tag{1.47}$$

is a linear difference equation which tells that the nth member of the output sequence $y(n)$ is obtained by accumulating (adding) the input at the present moment, $x(n)$, with thrice the previous input, $x(n-1)$ and twice the input delayed twice, $x(n-2)$. Let the input sequence be $x(n) = \{0, 1, 1, 2, 0, 0, 0, ...\}$. The output sequence for the system as described by Eq. (1.47) is $y(n) = \{0, 1, 4, 7, 8, 4, 0, 0, ...\}$. The block diagram representation of the system described by Eq. (1.47) is shown in Fig. 1.15.

In digital signal processing applications, our prime concern is of linear, time-invariant discrete-time systems. Such systems are modelled using linear difference equations with constant coefficients. The block diagram representation of these systems contain only unit delays, constant multipliers and adders.

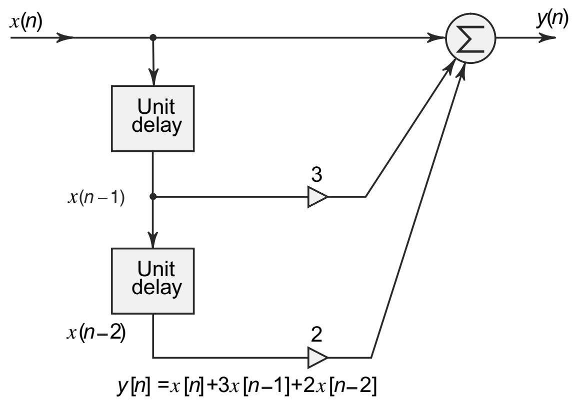


Fig. 1.15 *Discrete-Time System Corresponding to Eq. (1.47)*

A continuous-time system is modelled by a linear differential equation. An ordinary linear differential equation with constant coefficients characterises linear, constant parameter systems. For example, an nth order system is represented by

$$a_n \frac{d^n y(t)}{dt^n} + a_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \cdots + a_1 \frac{dy(t)}{dt} + y(t) = x(t) \tag{1.48}$$

The general solution of the above equation consists of two components, namely, the homogeneous solution and the particular solution. The homogeneous solution is the sourcefree, natural solution of the system, whereas the particular solution is the component due to the source $x(t)$.

1.7.2 Impulse Response of a System

The impulse response of a system is another method for modelling a system. The impulse response of a linear, time-invariant system is the response of the system when the input signal is a unit-impulse function. The system is assumed to be initially relaxed, i.e. the system has zero initial conditions. The impulse response of a system is represented by the notation $h(t)$ (continuous-time) or $h(n)$ (discrete-time). If $y(t)$ is the system response for an input $x(t)$, then the response of the system when $x(t) = \delta(t)$ is $y(t) = h(t)$.

The impulse response of a system can be directly obtained from the solution of the differential or difference equation characterising the system. The impulse response is also determined by finding the output of the system to the rectangular pulse input $x(t) = \frac{1}{\varepsilon}\Pi\left(\frac{t}{\varepsilon}\right)$ and then taking the limit of the resulting system response, $y(t)$ as $\varepsilon \to 0$. The unit-impulse function is nothing but the derivative of the unit-step signal. Therefore, the impulse response of the system can also be obtained by computing the derivative of the step response of the system.

1.7.3 State-Variable Technique

The state-variable technique provides a convenient formulation procedure for modelling a multi-input, multi-output system. This technique also facilitates the determination of the internal behaviour of the system very easily. The state of a system at time t_0 is the minimum information necessary to completely specify the condition of the system at time t_0 (or n_0) and it allows determination of the system outputs at any time $t > t_0$ (or $n > n_0$), when inputs up to time t are specified. The state of a system at time t_0 (or n_0) is a set of values, at time t_0 (or n_0) of a set variables. The information-bearing variables of a system are called state variables. The set of all state variables is called the system's *state.* State variables contain sufficient information so that all future states and outputs can be computed if the past history, input/output relationships and future inputs of the system are known.

The number of state variables is equal to the order of the system. The state variables are chosen such that they correspond to physically measurable quantities. It can be extended to nonlinear and time varying systems. It is also convenient to consider an N-dimensional space in which each coordinate is defined by one of the state variables $x_1, x_2 \ldots x_n$, where n is the order of the system. This N-dimensional space is called the state space. The state vector is defined as an N-dimensional vector $x(n)$, whose elements are the state variables. The state vector defines a point in the state space at any time t. As the time changes, the system state changes and a set of points, which is nothing but the locus of the tip of the state vector as time progresses, is called a *trajectory* of the system.

An alternative time-domain representation of a causal LTI discrete time system is by means of the state space equation. They can be obtained by reducing the N-th order difference equation to a system of N-first order equations.

In continuous time circuits, the state variables are defined as the inductor currents or capacitor voltages that do not change instantaneously. In discrete-time systems, the state variables are defined as the outputs of delay elements.

Since the state space analysis is essentially in time-domain approach, digital computers can be more effectively used. Conventional control method using Bode and Nyquist plots that are frequency domain approach requires Laplace transform for continuous-time systems and z-transform for discrete time systems. But for both continuous and discrete-time systems, vector matrix form of state space representation greatly simplifies system representation and gives accuracy of system performance.

The signal of an integrator can be labelled as $\frac{dy}{dt} = x(t)$, i.e., the input signal is the derivative of the output signal. A state space representation is derived from an integrator block diagram shown in Fig. 1.16 establishing the continuity between transfer functions and differential equations.

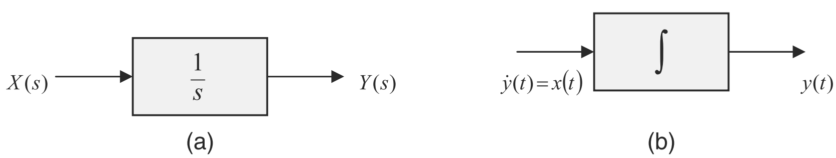


Fig. 1.16 *An Integrator*

State Space Representation of Continuous-Time LTI Systems A linear system of order n with m inputs and k outputs can be represented by n first-order differential equations and k output equations as shown below.

$$\begin{aligned}
\frac{dx_1(t)}{dt} &= a_{11}x_1(t)+a_{12}x_2(t)+\cdots+a_{1n}x_n(t)+b_{11}u_1(t)+b_{12}u_2(t)+\cdots+b_{1m}u_m(t)\\
\frac{dx_2(t)}{dt} &= a_{21}x_1(t)+a_{22}x_2(t)+\cdots+a_{2n}x_n(t)+b_{21}u_1(t)+b_{22}u_2(t)+\cdots+b_{2m}u_m(t)\\
&\vdots\\
\frac{dx_n(t)}{dt} &= a_{n1}x_1(t)+a_{n2}x_2(t)+\cdots+a_{nn}x_n(t)+b_{n1}u_1(t)+b_{n2}u_2(t)+\cdots+b_{nm}u_m(t)
\end{aligned} \tag{1.49}$$

and

$$\begin{aligned}
y_1(t) &= c_{11}x_1(t)+c_{12}x_2(t)+\cdots+c_{1n}x_n(t)+d_{11}u_1(t)+d_{12}u_2(t)+\cdots+d_{1m}u_m(t)\\
y_2(t) &= c_{21}x_1(t)+c_{22}x_2(t)+\cdots+c_{2n}x_n(t)+d_{21}u_1(t)+d_{22}u_2(t)+\cdots+d_{2m}u_m(t)\\
&\vdots\\
y_k(t) &= c_{k1}x_1(t)+c_{k2}x_2(t)+\cdots+c_{kn}x_n(t)+d_{k1}u_1(t)+d_{k2}u_2(t)+\cdots+d_{km}u_m(t)
\end{aligned} \tag{1.50}$$

where u_i, $i = 1, 2, \ldots, m$ are the system inputs, x_i, $i = 1, 2, 3, \ldots, n$ are called the state variables and y_i, $i = 1, 2, 3, \ldots, k$ are the system outputs. Equations (1.49) are called the *state equations* and Eqs. (1.50) are the *output equations*. Equations (1.49) and (1.50) together constitute the state-equation model of the system. Generally, the a, b, c and d's may be functions of time. The solution of such a set of time-varying state equations is very difficult. If the system is assumed to be time-invariant, then the solution of the state equations can be obtained without much difficulty.

The state variable representation of a system offers a number of advantages. The most obvious advantage of this representation is that multiple-input, multiple-output systems can be easily represented and analysed. The model is in the time-domain, and one can obtain the simulation diagram for the equation directly. This is of much use when computer simulation methods are used to analyse the system. Also, a compact matrix notation can be used for the state model and using the laws of linear algebra the state equations can be very easily manipulated. For example, Eq. (1.49) and Eq. (1.50) expressed in a compact matrix form is shown below. Let us define vectors

$$x(t)=\begin{bmatrix} x_1(t)\\ x_2(t)\\ \vdots\\ x_n(t)\end{bmatrix}, u(t)=\begin{bmatrix} u_1(t)\\ u_2(t)\\ \vdots\\ u_m(t)\end{bmatrix}, y(t)=\begin{bmatrix} y_1(t)\\ y_2(t)\\ \vdots\\ y_n(t)\end{bmatrix} \tag{1.51}$$

and matrices

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}, B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{21} & b_{22} & \cdots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nm} \end{bmatrix} \tag{1.52}$$

$$C = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1} & a_{k2} & \cdots & c_{kn} \end{bmatrix}, D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{k1} & d_{k2} & \cdots & d_{km} \end{bmatrix}$$

Now, Eq. (1.49) and Eq. (1.50) can be compactly written as

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1.53}$$

$$y(t) = Cx(t) + Du(t) \tag{1.54}$$

where $\dot{x}(t) = dx/dt$. Equations (1.53) and (1.54) may be illustrated schematically as shown in Fig. 1.17. The double lines indicate a multiple-variable signal flow path. The blocks represent matrix multiplication of the vectors and matrices. The integrator block consists of n integrators with appropriate connections specified by the A and B matrices.

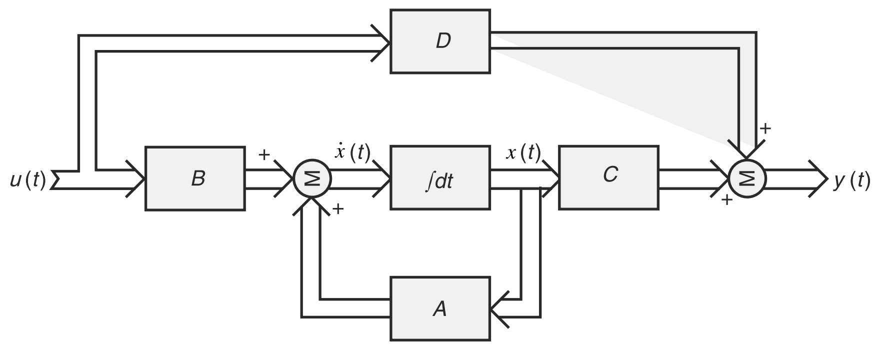


Fig. 1.17 *Block diagram of the State-Variable Model*

Solution of State Equation for Continuous-Time Systems

$$\dot{x}(t) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t)$$

Since all physical systems are low pass filters, D is taken as zero.
Assuming zero initial conditions and taking the Laplace transform, we get

$$sX(s) = AX(s) + Bu(s) \tag{1.55}$$

$$Y(s) = CX(s) \tag{1.56}$$

From Eq. (1.55), we have

$[sI - A]\, X(s) = BU(s)$, since A is a matrix, the variable s is multiplied with an Identity matrix, I.

Rearranging the above equation, we get

$$X(s) = \frac{1}{[sI - A]} BU(s) \tag{1.57}$$

Substituting Eq. (1.57) in Eq. (1.56), we get

$$Y(s) = C[sI - A]^{-1} BU(s)$$

$$\text{Transfer function} = \frac{Y(s)}{U(s)} = C[sI - A]^{-1} B \tag{1.58}$$

Therefore, $C[sI - A]^{-1}B$ is the transfer function of a system whose output is $Y(s)$ and input is $U(s)$.
Note: The solution in time-domain is given by

$$y(t) = Ce^{At} x(0) + \int_0^t Ce^{A(t-\tau)} Bu(\tau) d\tau + Du(t)$$

State Space Representation of Discrete-Time LTI Systems Consider a single-input single-output discrete-time LTI system which is described by an Nth-order difference equation

$$y[n] + a_1 y[n-1] + \cdots + a_N y[n-N] = u[n] \tag{1.59}$$

Here, if $u[n]$ is given for $n \geq 0$, Eq. (1.59) requires N initial conditions $y[-1]$, $y[-2], \ldots y[-N]$ to uniquely determine the complete solution for $n > 0$. Thus, N values are required to specify the state of the system at any time.

Then, N state variables $x_1[n], x_2[n], \ldots, x_N[n]$ are defined as

$$\begin{aligned} x_1[n] &= y[n-N] \\ x_2[n] &= y[n-(N-1)] = y[n-N+1] \\ &\vdots \\ x_N[n] &= y[n-1] \end{aligned} \tag{1.60}$$

Then from Eq. (1.59) and Eq. (1.60), we have

$$\begin{aligned} x_1[n+1] &= x_2[n] \\ x_2[n+1] &= x_3[n] \\ &\vdots \\ x_N[n+1] &= -a_N x_1[n] - a_{N-1} x_2[n] - \cdots - a_1 x_N[n] + u[n] \end{aligned}$$

and $$y[n] = -a_N x_1[n] - a_{N-1} x_2[n] - \cdots - a_1 x_N[n] + u[n]$$

In matrix form, the above equations can be expressed as

$$\begin{bmatrix} x_1[n+1] \\ x_2[n+1] \\ \vdots \\ x_N[n+1] \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_N & -a_{N-1} & -a_{N-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1[n] \\ x_2[n] \\ \vdots \\ x_N[n] \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} u[n] \tag{1.61}$$

$$y[n] = [-a_N \quad -a_{N-1} \quad \cdots \quad -a_1] \begin{bmatrix} x_1[n] \\ x_2[n] \\ \vdots \\ x_N[n] \end{bmatrix} + [1]u[n] \tag{1.62}$$

Now the state vector $x[n]$ is defined by an $N \times 1$ matrix (or N-dimensional vector) as

$$x[n] = \begin{bmatrix} x_1[n] \\ x_2[n] \\ \vdots \\ x_N[n] \end{bmatrix}$$

Then Eq. (1.61) and Eq. (1.62) can be rewritten compactly as

$$x[n+1] = Ax[n] + Bu[n] \tag{1.63}$$

$$y(n) = Cx[n] + Du[n] \tag{1.64}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_N & -a_{N-1} & -a_{N-2} & \cdots & -a_1 \end{bmatrix}; \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

$$C = [-a_N \quad -a_{N-1} \quad \cdots \quad -a_1] \text{ and } \quad D = 1$$

Equations (1.63) and (1.64) are called an N-dimensional state space representation (or state equations) of the system, and the $N \times N$ matrix A is called the system matrix.

Multiple-Input Multiple-Output Systems If a discrete-time LTI system has m inputs and p outputs and N state variables, then a state space representation of the system can be represented as

$$x[n+1] = Ax[n] + Bu[n]$$

$$y(n) = Cx[n] + Du[n]$$

where

$$x[n] = \begin{bmatrix} x_1[n] \\ x_2[n] \\ \vdots \\ x_N[n] \end{bmatrix}; \; u[n] = \begin{bmatrix} u_1[n] \\ u_2[n] \\ \vdots \\ u_m[n] \end{bmatrix}; \; y[n] = \begin{bmatrix} y_1[n] \\ y_2[n] \\ \vdots \\ y_p[n] \end{bmatrix}$$

and

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix}_{N \times N} \qquad B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{21} & b_{22} & \cdots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{N1} & b_{N2} & \cdots & b_{Nm} \end{bmatrix}_{N \times m}$$

$$C = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1N} \\ c_{21} & c_{22} & \cdots & c_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ c_{p1} & c_{p2} & \cdots & c_{pN} \end{bmatrix}_{p\times N} \qquad D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{p1} & d_{p2} & \cdots & d_{pm} \end{bmatrix}_{p\times m}$$

Solutions of State Equations for Discrete-time LTI Systems Consider an N-dimensional state representation

$$x[n+1] = Ax[n] + Bu[n] \tag{1.65}$$

$$y(n) = Cx[n] + Du[n] \tag{1.66}$$

where A, B, C and D are $N \times N$, $N \times 1$, $1 \times N$, and 1×1 matrices, respectively. Taking the z-transform of Eq. (1.65) and Eq. (1.66) and using time shifting property of z-transform, we get

$$zX(z) - zx(0) = AX(z) + BU(z) \tag{1.67}$$

$$Y(z) = CX(z) + BU(z) \tag{1.68}$$

where $X(z) = Z\{x[n]\}$, $Y(z) = Z\{y[n]\}$, and

$$X(z) = Z\{x[n]\} = \begin{bmatrix} X_1(z) \\ X_2(z) \\ \vdots \\ X_N(z) \end{bmatrix} \text{ where } X_k(z) = Z\{x_k[n]\}$$

Rearranging Eq. (1.67), we have

$$(zI - A)X(z) = zx(0) + BU(z) \tag{1.69}$$

Premultiplying both sides of Eq. (1.69) by $(zI - A)^{-1}$, we get

$$X(z) = (zI - A)^{-1} zx(0) + (zI - A)^{-1} BU(z) \tag{1.70}$$

Taking inverse z-transform of Eq. (1.70), we get

$$x[n] = Z^{-1}\left\{(zI - A)^{-1} z\right\} x(0) + Z^{-1}\left\{(zI - A)^{-1} BU(z)\right\} \tag{1.71}$$

Substituting Eq. (1.71) into Eq. (1.66), we get

$$y[n] = CZ^{-1}\left\{ (zI - A)^{-1} z\right\} x(0) + CZ^{-1}\left\{(zI - A)^{-1} BU(z)\right\} + Du[n] \tag{1.72}$$

To Determine System Function H(z) The system function $H(z)$ of a discrete-time LTI system is defined by $H(z) = Y(z)/U(z)$ with zero initial conditions. Thus, setting $x[0] = 0$ in Eq. (1.70), we have

$$X(z) = (zI - A)^{-1} BU(z) \tag{1.73}$$

Substituting Eq. (1.73) into Eq. (1.68), we get

$$Y(z) = \left[C\ (zI - A)^{-1} B + D\right] U(z)$$

Thus,

$$H(z) = \frac{Y(z)}{U(z)} = \left[C(zI - A)^{-1} B + D\right]$$

Example 1.26 *Obtain the transfer function of the system defined by the following state space equations.*

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} -1 & 1 & -1 \\ 0 & -2 & 1 \\ 0 & 0 & -3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$$

Solution Transfer function $= \dfrac{Y(s)}{U(s)} = C\left[sI - A\right]^{-1} B$

$$\left[sI - A\right] = \begin{bmatrix} s+1 & -1 & 1 \\ 0 & s+2 & -1 \\ 0 & 0 & s+3 \end{bmatrix}$$

Therefore,

$$\left[sI - A\right]^{-1} = \begin{bmatrix} \dfrac{1}{s+1} & \dfrac{1}{(s+1)\,(s+2)} & \dfrac{-1}{(s+2)\,(s+3)} \\ 0 & \dfrac{1}{s+2} & \dfrac{1}{(s+2)\,(s+3)} \\ 0 & 0 & \dfrac{1}{s+3} \end{bmatrix}$$

Transfer function = $C[sI-A]^{-1}B$

$$= \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \dfrac{1}{s+1} & \dfrac{1}{(s+1)\,(s+2)} & \dfrac{-1}{(s+2)\,(s+3)} \\ 0 & \dfrac{1}{s+2} & \dfrac{1}{(s+2)\,(s+3)} \\ 0 & 0 & \dfrac{1}{s+3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Transfer function $= \begin{bmatrix} \dfrac{2(s+2)}{(s+1)(s+3)} & \dfrac{1}{s+1} \\ \dfrac{1}{s+2} & \dfrac{1}{s+2} \end{bmatrix}$

Example 1.27 *Find the state equation and output equation for the system given by*

$$G(s) = \frac{1}{s^3 + 4s^2 + 3s + 3}$$

Solution Given $G(s) = \left.\dfrac{Y(s)}{U(s)}\right|_{Initial \;\; conditions = 0}$

$$\frac{Y(s)}{U(s)} = \frac{1}{s^3 + 4s^2 + 3s + 3}$$

$$\left[s^3 + 4s^2 + 3s + 3\right]Y(s) = U(s)$$

Taking inverse Laplace transform with all initial condition $s = 0$.

$$\frac{d^3 y(t)}{dt^3} + 4\frac{d^2 y(t)}{dt^2} + 3\frac{dy(t)}{dt} + 3y(t) = u(t)$$

or, $\dddot{y}(t) + 4\ddot{y}(t) + 3\dot{y}(t) + 3y(t) = u(t)$

The state variables $y(t)$ and its derivatives are chosen as

$$x_1(t) = y(t);\; x_2(t) = \dot{y}(t);\; x_3(t) = \ddot{y}(t)$$

To find state equation Differentiating the state variables, we get the state equations as

$$\dot{x}_1(t) = \dot{y}(t) = x_2(t)$$

$$\dot{x}_2(t) = \ddot{y}(t) = x_3(t)$$

$$\dot{x}_3(t) = \dddot{y}(t) = -3y(t) - 3\dot{y}(t) - 4\ddot{y}(t) + u(t)$$

$$\dot{x}_3(t) = -3x_1(t) - 3x_2(t) - 4x_3(t) + u(t)$$

The state equation in matrix form is given by

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\\ \dot{x}_3(t)\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -3 & -3 & -4\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\\ x_3(t)\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 1\end{bmatrix}u(t)$$

Output equation The output equation is $y(t) = x_1(t)$

$$y(t) = [1\;\;0\;\;0]\begin{bmatrix}x_1(t)\\ x_2(t)\\ x_3(t)\end{bmatrix}$$

Example 1.28 *Given a continuous-time LTI system with system function*

$H(s) = \dfrac{3s+7}{(s+1)(s+2)(s+5)}$, *determine a state representation of the system.*

Solution Given $H(s) = \dfrac{3s+7}{(s+1)(s+2)(s+5)} = \dfrac{3s+7}{s^3 + 8s^2 + 17s + 10} = \dfrac{3s^{-2} + 7s^{-3}}{1 + 8s^{-1} + 17s^{-2} + 10s^{-3}}$

Comparing the above equation with the standard system function $H(s) = \dfrac{b_0 + b_1 s^{-1} + b_2 s^{-2} + \ldots}{1 + a_1 s^{-1} + a_2 s^{-2} + \ldots}$, we get

$$a_1 = 8;\; a_2 = 17;\; a_3 = 10;\; b_0 = b_1 = 0;\; b_2 = 3;\; b_3 = 7$$

Representing these values in matrix form as

$$\dot{x}(t)=\begin{bmatrix}-8 & 1 & 0\\ -17 & 0 & 1\\ -10 & 0 & 0\end{bmatrix}x(t)+\begin{bmatrix}0\\ 3\\ 7\end{bmatrix}u(t)$$

$$y(t)=\begin{bmatrix}1 & 0 & 0\end{bmatrix}x(t)$$

Example 1.29 *Obtain state space representation for system* $\dddot{y}(t)+8\ddot{y}(t)+11\dot{y}(t)+6y(t)=u(t)$

Solution State variables are chosen as

$$x_1(t)=y(t);\ x_2(t)=\dot{y}(t);\ x_3(t)=\ddot{y}(t)$$

We know that number of state variables is equal to the order of the system. Therefore, the order of the system is 3 and hence the number of state variables is also 3.

$$\dot{x}_1(t)=\dot{y}(t)=x_2(t)$$

$$\dot{x}_2(t)=\ddot{y}(t)=x_3(t)$$

$$\dot{x}_3(t)=\dddot{y}(t)=-8\ddot{y}(t)-11\dot{y}(t)-6y(t)+u(t)$$

$$\dot{x}_3(t)=-8x_3(t)-11x_2(t)-6x_1(t)+u(t)=-6x_1(t)-11x_2(t)-8x_3(t)+u(t)$$

State space representation is given by

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\\ \dot{x}_3(t)\end{bmatrix}=\begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -6 & -11 & -8\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\\ x_3(t)\end{bmatrix}+\begin{bmatrix}0\\ 0\\ 1\end{bmatrix}u(t)$$

The output equation is $y(t)=x_1(t)$

$$y(t)=[1\quad 0\quad 0]\begin{bmatrix}x_1(t)\\ x_2(t)\\ x_3(t)\end{bmatrix}$$

Example 1.30 *Find state space representation for system* $\ddot{y}(t)+6\dot{y}(t)+2y(t)=0.$

Solution The state variables $y(t)$ and its derivatives are

$$x_1(t)=y(t)$$

$$x_2(t)=\dot{y}(t)$$

To Find State Equation Differentiating the state variables, we get the state equations as

$$\dot{x}_1(t)=\dot{y}(t)=x_2(t)$$

$$\dot{x}_2(t)=\ddot{y}(t)=-2y(t)-6\dot{y}(t)$$

$$\dot{x}_2(t)=-2x_1(t)-6x_2(t)$$

The state equation in matrix form is

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix}=\begin{bmatrix}0 & 1\\ -2 & -6\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}$$

The output equation is $y(t) = x_1(t)$

$$y(t) = [1 \quad 0]\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

Example 1.31 *Determine state equations of a continuous-time LTI system described by* $\ddot{y}(t) + 3\dot{y}(t) + 2y(t) = u(t)$

Solution The state variables are chosen as

$$x_1(t) = y(t);\ x_2(t) = \dot{y}(t)$$

Therefore,

$$x_1(t) = x_2(t)$$

$$x_2(t) = -2x_1(t) - 3x_2(t) + u(t)$$

$$y(t) = x_1(t)$$

Representing the above equation in matrix form, we have

$$x(t) = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t)$$

Example 1.32 *Find a state space representation for the RLC circuit shown in Fig. E1.32. Let the output y(t) be the loop current.*

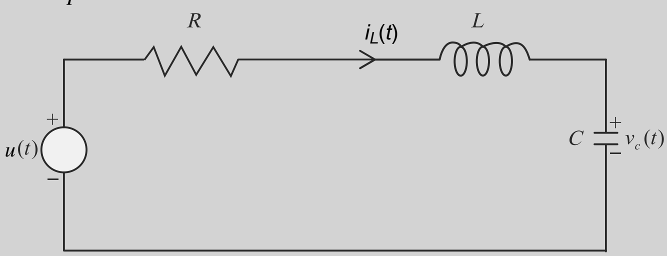


Fig. E.1.32 *RLC Circuit*

Solution The state variables are chosen as

$$x_1(t) = i_L(t) \quad \text{and} \quad x_2(t) = v_c(t)$$

Using Kirchhoff's law, we get

$$L\dot{x}_1(t) + Rx_1(t) + x_2(t) = u(t)$$

$$C\dot{x}_2(t) = x_1(t)$$

$$y(t) = x_1(t)$$

Rearranging and writing in matrix form, we get

$$\dot{x}(t) = \begin{bmatrix} \frac{-R}{L} & \frac{-1}{L} \\ \frac{1}{C} & 0 \end{bmatrix} x(t) + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} u(t)$$

$$y(t) = [1 \quad 0]\ x(t)$$

Example 1.33 *Determine the state equations of a discrete-time LTI system with system function* $H(z)=\dfrac{b_0+b_1z^{-1}+b_2z^{-2}}{1+a_1z^{-1}+a_2z^{-2}}$.

Solution $H(z)=\dfrac{Y(z)}{U(z)}=\dfrac{b_0+b_1z^{-1}+b_2z^{-2}}{1+a_1z^{-1}+a_2z^{-2}}$

The above equation can be written as

$$(1+a_1z^{-1}+a_2z^{-2})Y(z)=(b_0+b_1z^{-1}+b_2z^{-2})U(z)$$

Upon rearranging, we get

$$Y(z)=-a_1z^{-1}Y(z)-a_2z^{-2}Y(z)+b_0U(z)+b_1z^{-1}U(z)+b_2z^{-2}U(z)$$

For this equation, the Canonical state representation of the filter structure is shown in Fig. E1.33. Selecting the outputs of unit-delay elements as state variables as shown in Fig. E1.33, we get

$$\begin{aligned} y(n) &= x_1(n)+b_0u(n)\\ x_1(n+1) &= -a_1y(n)+x_2(n)+b_1u(n)\\ &= -a_1x_1(n)+x_2(n)+(b_1-a_1b_0)u(n)\\ x_2(n+1) &= -a_2y(n)+b_2u(n)\\ &= -a_2x_1(n)+(b_2-a_2b_0)u(n) \end{aligned}$$

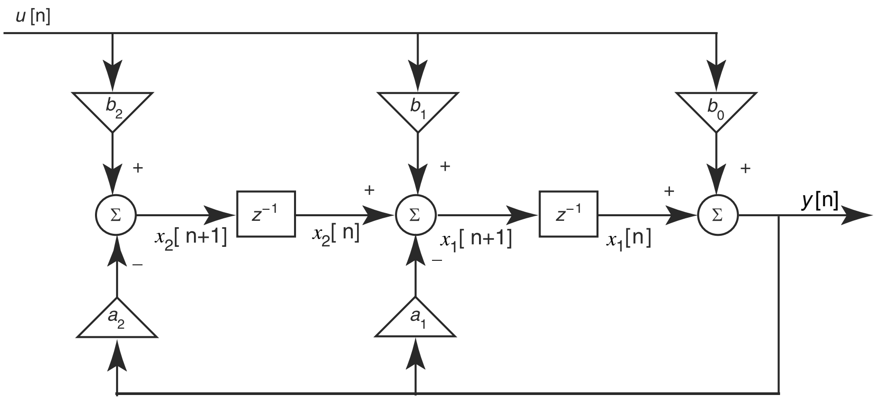


Fig. E1.33 *Canonical State Representation of the Filter Structure*

We can write the above equation in matrix form as

$$x[n+1]=\begin{bmatrix}-a_1 & 1\\ -a_2 & 0\end{bmatrix}x[n]+\begin{bmatrix}b_1 & -a_1b_0\\ b_2 & -a_2b_0\end{bmatrix}u[n]$$

$$y[n]=[1 \quad 0]x[n]+b_0u[n]$$

Example 1.34 *Given a discrete-time LTI system with system function* $H(z)=\dfrac{z}{2z^2-3z+1}$, *find a state representation of the system.*

Solution Given $H(z)=\dfrac{z}{2z^2-3z+1}$

Therefore, $H(z)=\dfrac{z}{2z^2\left(1-\frac{3}{2}z^{-1}+\frac{1}{2}z^{-2}\right)}=\dfrac{\frac{1}{2}z^{-1}}{\left(1-\frac{3}{2}z^{-1}+\frac{1}{2}z^{-2}\right)}$

$$H(z)=\frac{Y(z)}{U(z)}=\frac{\frac{1}{2}z^{-1}}{\left(1-\frac{3}{2}z^{-1}+\frac{1}{2}z^{-2}\right)}$$

Hence, $Y(z)=\frac{3}{2}z^{-1}Y(z)-\frac{1}{2}z^{-2}Y(z)+\frac{1}{2}z^{-1}U(z)$

Comparing the expression for $H(z)$ with the standard equation, $H(z)=\dfrac{b_0+b_1z^{-1}+b_2z^{-2}+\cdots}{1+a_1z^{-1}+a_2z^{-2}+\cdots}$, we get

$$a_1=-\frac{3}{2};\ a_2=\frac{1}{2};\ b_0=0;\ b_1=\frac{1}{2};\ b_2=0$$

Using the results obtained in Example 1.33, we get

$$y[n]=x_1[n]$$

$$x_1[n+1]=-\frac{3}{2}y[n]+x_2[n]+\frac{1}{2}u[n]=-\frac{3}{2}x_1[n]+x_2[n]+\left(\frac{1}{2}\right)u[n]$$

$$x_2[n+1]=-\frac{1}{2}y(n)$$

Therefore, $x[n+1]=\begin{bmatrix}\frac{3}{2} & 1\\ -\frac{1}{2} & 0\end{bmatrix}x[n]+\begin{bmatrix}\frac{1}{2}\\ 0\end{bmatrix}u[n]$

$$y[n]=[1 \quad 0]x(n)$$

Example 1.35 *Sketch a block diagram of a discrete-time system with the state space representation* $x[n+1]=\begin{bmatrix}0 & 1\\ 2 & 5\end{bmatrix}x[n]+\begin{bmatrix}0\\ 1\end{bmatrix}u[n]$

$$y[n]=[8 \ \ -7]x[n]$$

Solution The given equation can be expressed as

$$x_1[n+1]=x_2[n]$$

$$x_2[n+1]=2x_1[n]+5x_2[n]+u[n]$$

$$y[n]=8x_1[n]-7x_2[n]$$

Therefore, block diagram representation for the above equations is shown below.

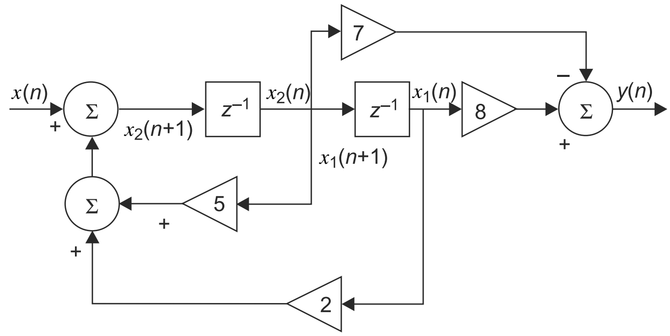


Fig. E1.35

ANALOG-TO-DIGITAL CONVERSION OF SIGNALS 1.8

A discrete-time signal is defined by specifying its value only at discrete times, called *sampling instants*. When the sampled values are quantised and encoded, a digital signal is obtained. A digital signal can be obtained from the analog signal by using an analog-to-digital converter. In the following sections the process of analog-to-digital conversion is discussed in some detail and this enables one to understand the relationship between the digital signals and discrete-time signals.

Figure 1.18 shows the block diagram of an analog-to-digital converter. The *sampler* extracts the sample values of the input signal at the sampling instants. The output of the sampler is the discrete-time signal with continuous amplitude. This signal is applied to a *quantiser* which converts this continuous amplitude into a finite number of sample values. Each sample value can be represented by a digital word of finite word length. The final stage of analog-to-digital conversion is encoding. The *encoder* assigns a digital word to each quantised sample. Sampling, quantising and encoding are discussed in the following sections.

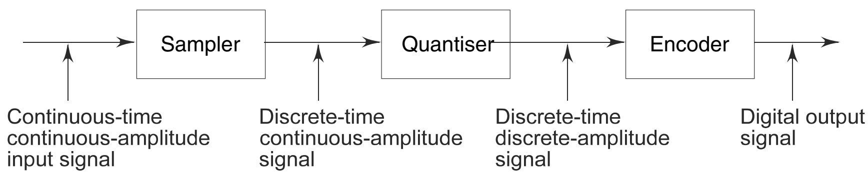


Fig. 1.18 *Analog-to-Digital Converter*

1.8.1 Sampling of Continuous-time Signals

In the application of signal processing, the measurement and analysis of signals are very important to understand their characteristics. If the signal is unknown, the process of analysis begins with the acquisition of the signal. The most common technique of acquiring signals is by sampling. Sampling a signal is the process of acquiring its values only at discrete points in time. The signals acquired in this way, in most of signal processing and analysis is done using digital computers. The discrete time signals are expressed as solutions of discrete time difference equations and transforms.

The analysis of sampled signal in the frequency domain is extremely difficult using s-plane representation because the equations of the signal contain infinite long polynomials due to infinite number of poles and zeros. Fortunately, this problem may be overcome by using z-transform, which reduces the number of poles and zeros to be finite in the z-plane.

Sampling is a process by which a continuous-time signal is converted into a discrete-time signal. This can be accomplished by representing the continuous-time signal $x(t)$, at a discrete number of points. These discrete number of points are determined by the sampling period, T, i.e. the samples of $x(t)$ can be obtained at discrete points $t = nT$, where n is an integer. The process of sampling is illustrated in Fig. 1.19. The sampling unit can be thought of as a switch, where, to one of its inputs the continuous-time signal is applied. The signal is available at the output only during the instants the switch is closed. Thus, the signal at the output end is not a continuous function of time but only discrete samples. In order to extract samples of $x(t)$, the switch closes briefly every T seconds. Thus, the output signal has the same amplitude as $x(t)$ when the switch is closed and a value of zero when the switch is open. The switch can be any high speed switching device.

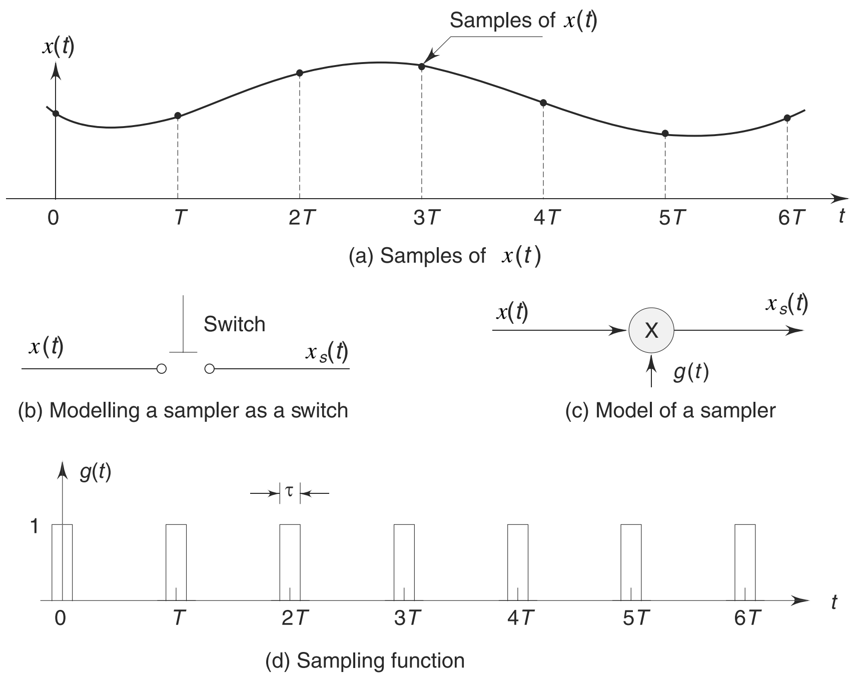


Fig. 1.19 *The Sampling Process*

The continuous-time signal $x(t)$ must be sampled in such a way that the original signal can be reconstructed from these samples. Otherwise, the sampling process is useless. Let us obtain the condition necessary to faithfully reconstruct the original signal from the samples of that signal. The condition can be easily obtained if the signals are analysed in the frequency domain. Let the sampled signal be represented by $x_s(t)$. Then,

$$x_s(t) = x(t)\, g(t) \tag{1.74}$$

where $g(t)$ is the *sampling function*. The sampling function is a continuous train of pulses with a period of T seconds between the pulses, and it models the action of the sampling switch. The sampling function is shown in Fig. 1.19(c) and (d). The frequency spectrum of the sampled signal $x_s(t)$ helps in determining the appropriate values of T for reconstructing the original signal. The sampling function $g(t)$ is periodic and can be represented by a Fourier series (Fourier Series and transforms are discussed in Chapter 6), i.e.,

$$g(t) = \sum_{n=-\infty}^{\infty} C_n e^{jn2\pi f_s t} \tag{1.75}$$

where

$$C_n = \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} g(t)e^{-jn2\pi f_s t}\, \mathrm{d}t \tag{1.76}$$

is the nth Fourier coefficient of $g(t)$, and f_s = the fundamental frequency of $g(t)$. The fundamental frequency, f_s is also called the *sampling frequency*. From Eq. (1.74), and Eq. (1.75), we have

$$x_s(t) = x(t)\sum_{n=-\infty}^{\infty} C_n e^{jn2\pi f_s t} = \sum_{n=-\infty}^{\infty} C_n x(t) e^{jn2\pi f_s t} \tag{1.77}$$

The spectrum of $x_s(t)$, denoted by $X_s(f)$, can be determined by taking the Fourier transform of Eq. (1.77), i.e.

$$X_s(f) = \int_{-\infty}^{\infty} x_s(t)e^{-j2\pi ft}\, \mathrm{d}t \tag{1.78}$$

Using Eq. (1.77) in the above equation,

$$X_s(f) = \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} C_n x(t) e^{jn2\pi f_s t} \cdot e^{-j2\pi ft}\, \mathrm{d}t \tag{1.79}$$

Interchanging the order of integration and summation,

$$X_s(f) = \sum_{n=-\infty}^{\infty} C_n \int_{-\infty}^{\infty} x(t)e^{-j2\pi(f-nf_s)t}\, \mathrm{d}t \tag{1.80}$$

But from the definition of the Fourier transform

$$\int_{-\infty}^{\infty} x(t)e^{-j2\pi(f-nf_s)t}\, \mathrm{d}t = X(f-nf_s)$$

Thus,

$$X_s(f) = \sum_{n=-\infty}^{\infty} C_n X(f-nf_s) \tag{1.81}$$

From Eq. (1.81), it is understood that the spectrum of the sampled continuous-time signal is composed of the spectrum of $x(t)$ plus the spectrum of $x(t)$ translated to each harmonic of the sampling frequency. The spectrum of the sampled signal is shown in Fig. 1.20. Each frequency translated spectrum is multiplied by a constant. To reconstruct the original signal, it is enough to just pass the spectrum of $x(t)$ and suppress the spectra of other translated frequencies. The amplitude response of such a filter is also shown in Fig. 1.20. As this filter is used to reconstruct the original signal, it is often referred to as a *reconstruction filter*. The output of the reconstructed filter will be $C_0X(f)$ in the frequency domain and $x(t)$ in the time-domain.

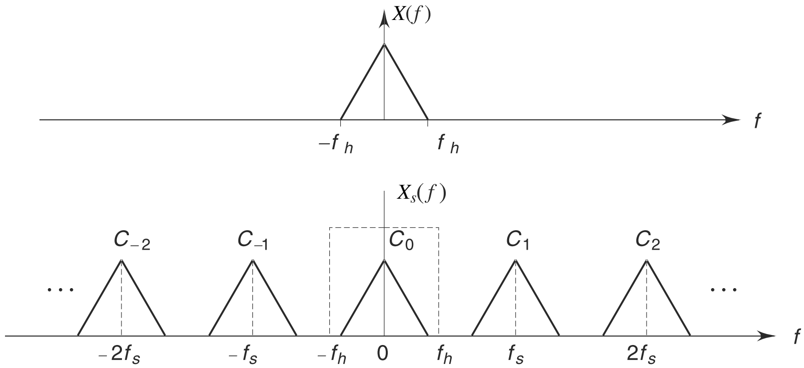


Fig. 1.20 *Spectrum of Sampled Signal*

The signal $x(t)$, in this case, is assumed to have no frequency components above f_h, i.e. in the frequency domain, $X(f)$ is zero for $|f| \geq f_h$. Such a signal is said to be ***bandlimited***. From Fig. 1.20, it is clear that in order to recover $X(f)$ from $X_s(f)$, we must have

$$f_s - f_h \geq f_h$$

or equivalently,

$$f_s \geq 2f_h, \text{ Hz} \tag{1.82}$$

That is, in order to recover the original signal from the samples, the sampling frequency must be greater than or equal to twice the maximum frequency in $x(t)$. *The sampling theorem is thus derived, which states that a bandlimited signal $x(t)$ having no frequency components above f_h hertz, is completely specified by samples that are taken at a uniform rate greater than $2f_h$ hertz.* The frequency equal to twice the highest frequency in $x(t)$, i.e. $2f_h$, is called the *Nyquist rate*.

Sampling by Impulse Function

The sampling function $g(t)$, discussed above, was periodic. The pulse width of the sampling function must be very small compared to the period, T. The samples in digital systems are in the form of a number, and the magnitude of these numbers represent the value of the signal $x(t)$ at the sampling instants. In this case, the pulse width of the sampling function is infinitely small and an infinite train of impulse functions of period T can be considered for the sampling function. That is,

$$g(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \tag{1.83}$$

The sampling function as given in Eq. (1.83) is shown in Fig. 1.22. When this sampling function is used, the weight of the impulse carries the sample value.

The sampling function $g(t)$ is periodic and can be represented by a Fourier series as in Eq. (1.75), which is repeated here.

$$g(t) = \sum_{n=-\infty}^{\infty} C_n e^{jn2\pi f_s t}$$

where

$$C_n = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \delta(t) e^{-jn2\pi f_s t} \, dt \tag{1.84}$$

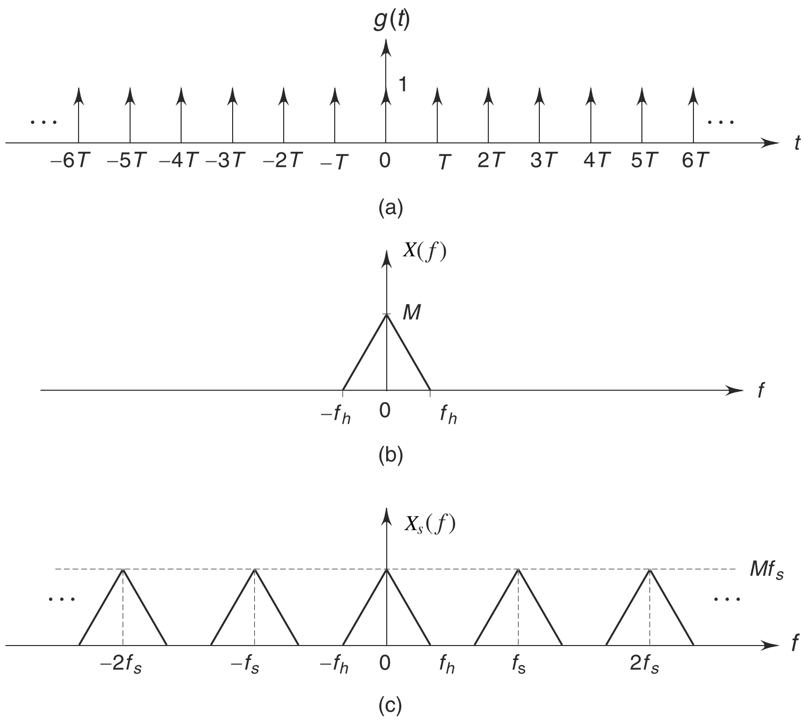


Fig. 1.21 *(a) Impulse Sampling Function (b) Spectrum of the Signal x(t) (c) Spectrum of Impulse Sampled Signal*

Since $\delta(t)$ has its maximum energy concentrated at $t = 0$, a more formal mathematical definition of the unit-impulse function may be defined as a functional

$$\int_{-\infty}^{\infty} x(t)\delta(t)\,dt = x(0) \tag{1.85}$$

where $x(t)$ is continuous at $t = 0$. Using Eq. (1.85) in Eq. (1.84), we have

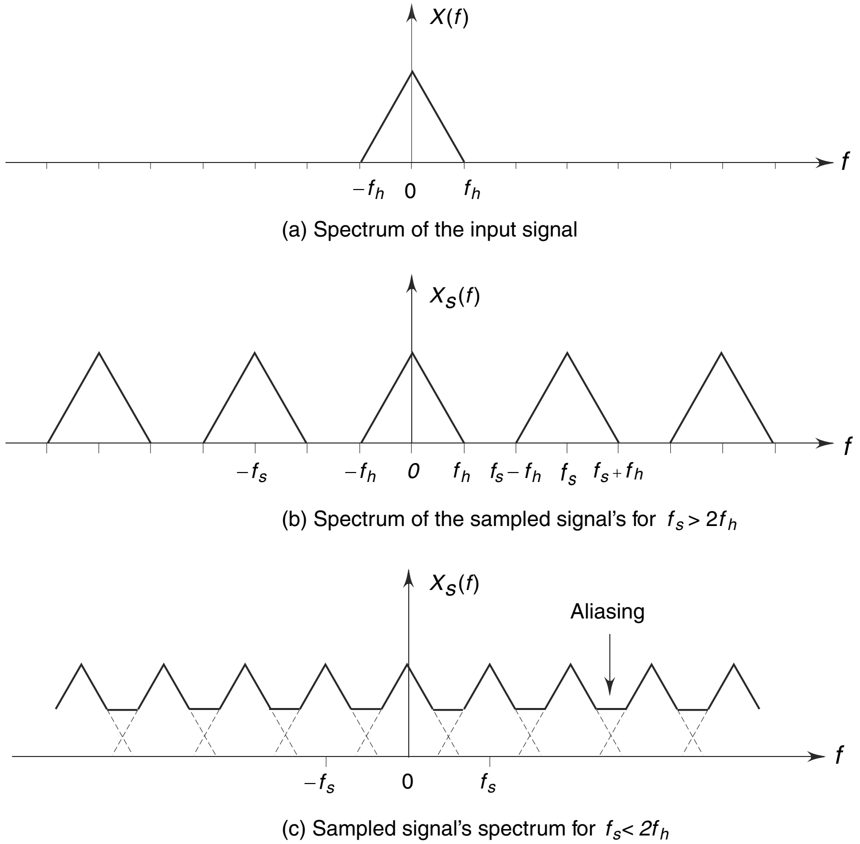
$$C_n = \frac{1}{T}e^0 = \frac{1}{T} = f_s \tag{1.86}$$

Thus C_n is same as the sampling frequency f_s, for all n. The spectrum of the impulse sampled signal, $x_s(t)$ is given by

$$X_s(f) = f_s \sum_{n=-\infty}^{\infty} X(f - nf_s) \tag{1.87}$$

The spectra of the signal $x(t)$ and the impulse sampled signal $X_s(t)$ are shown in Fig. 1.21(b) and (c). The effect of impulse sampling is same as sampling with a train of pulses. However, all the frequency translated spectra have the same amplitude. The original signal $X(f)$ can be reconstructed from $X_s(f)$ using a low-pass filter. Figure 1.22 shows the effect of sampling at a rate lower than the Nyquist rate.

Consider a bandlimited signal $x(t)$, with f_h as its highest frequency content, being sampled at a rate lower than the Nyquist rate, i.e., sampling frequency $f_s < 2f_h$. This results in overlapping of adjacent spectra, i.e. higher frequency components of $X_s(f)$ get superimposed on lower frequency components as shown in Fig. 1.22. Here, faithful reconstruction or recovery of the original continuous time signal



(a) Spectrum of the input signal

(b) Spectrum of the sampled signal's for $f_s > 2f_h$

(c) Sampled signal's spectrum for $f_s < 2f_h$

Fig. 1.22 *Illustration of Aliasing*

from its sampled discrete-time equivalent by filtering is very difficult because portions of $X(f - f_s)$ and $X(f + f_s)$ overlap $X(f)$, and thus add to $X(f)$ in producing $X_s(f)$. The original shape of the signal is lost due to undersampling, i.e. down-sampling. This overlap is known as *aliasing* or *overlapping* or *fold over*. Aliasing, as the name implies, means that a signal can be impersonated by another signal. In practice, no signal is strictly bandlimited but there will be some frequency beyond which the energy is very small and negligible. This frequency is generally taken as the highest frequency content of the signal.

To prevent aliasing, the sampling frequency f_s should be greater than two times the frequency f_h of the sinusoidal signal being sampled. The condition to be satisfied by the sampling frequency to prevent aliasing is called the *sampling theorem*. In some applications, an analog anti-aliasing filter is placed before sample/hold circuit in order to prevent the aliasing effect.

A useful application of aliasing due to undersampling arises in the *sampling oscilloscope*, which is meant for observing very high frequency waveforms.

1.8.2 Signal Reconstruction

Any signal $x(t)$ can be faithfully reconstructed from its samples if these samples are taken at a rate greater than or equal to the Nyquist rate. It can be seen from the spectrum of the sampled signal, $X_s(t)$ that it consists of the spectra of the signal and its frequency translated harmonics. Thus, if the spectrum

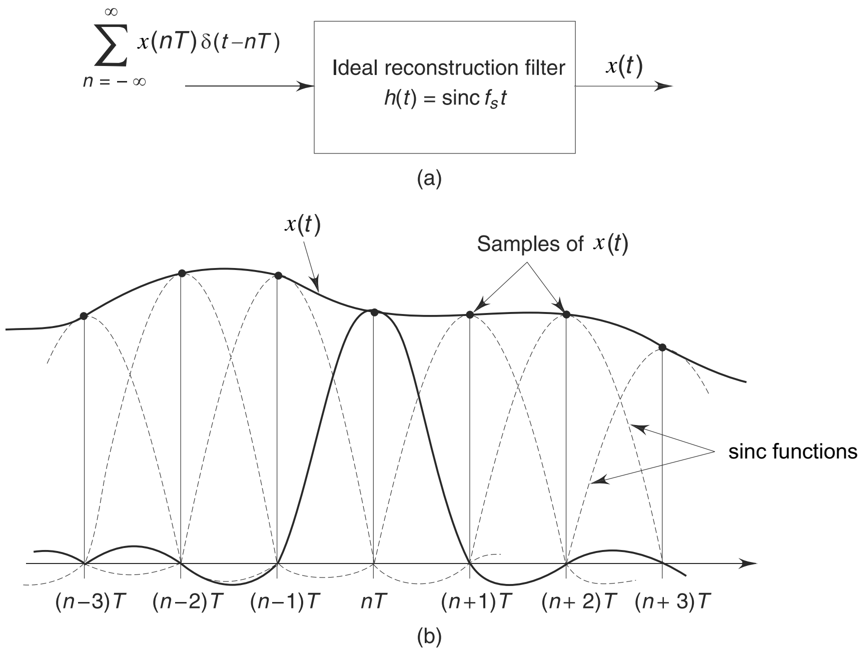


Fig. 1.23 *Signal Reconstruction (a) Reconstruction Filter (b) Time Domain Representation*

of the signal alone can be separated from that of the harmonics then the original signal can be obtained. This can be achieved by filtering the sampled signal using a low-pass filter with a bandwidth greater than f_h and less than $f_s - f_h$ hertz.

If the sampling function is an impulse sequence, we note from Eq. (1.87) that the spectrum of the sampled signal has an amplitude equal to $f_s = 1/T$. Therefore, in order to remove this scaling constant, the low-pass filter must have an amplitude response of $1/f_s = T$. Assuming that sampling has been done at the Nyquist rate, i.e. $f_s = 2f_h$, the bandwidth of the low-pass filter will be $f_h = \frac{f_s}{2}$. Therefore, the unit impulse response of an ideal filter for this bandwidth is

$$h(t) = T\int_{-f_s/2}^{f_s/2} e^{j2\pi ft}\,\mathrm{d}t \tag{1.88}$$

That is

$$h(t) = \frac{T}{j2\pi t}\left(e^{j\pi f_s t} - e^{-j\pi f_s t}\right)$$

The above expression can be alternatively written as

$$h(t) = Tf_s\frac{\sin \pi f_s t}{\pi f_s t} = \text{sinc } f_s t \tag{1.89}$$

The ideal reconstruction filter is shown in Fig. 1.23(a). The input to this filter is the sampled signal $x(nT)$ and the output of the filter is the reconstructed signal $x(t)$. The output signal $x(t)$ is given by

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT)h(t-nT)$$

Using Eq. (1.89), we get

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT)\sin c\, f_s(t-nT) \tag{1.90}$$

The above expression is a ***convolution*** expression and the signal $x(t)$ is reconstructed by convoluting its samples with the unit-impulse response of the filter. Eq. (1.90) can also be interpreted as follows. The original signal can be reconstructed by weighting each sample by a sinc function and adding them all. This process is shown in Fig. 1.23(b).

1.8.3 Sampling of Band-Pass Signals

Shannon's sampling theorem states that if a signal is sampled for all time at a rate more than twice the highest frequency at which its continuous-time Fourier transform (CTFT) is nonzero, it can be exactly reconstructed from the samples. If we sample at a rate higher than twice the highest frequency, the aliases do not overlap and the original signal can be recovered. If we sample at a slower rate, then aliases would overlap. That is true for all signals, but for some signals the minimum sampling rate can be reduced if a continuous-time signal has a band-pass spectrum that is nonzero only for $f_1 < |f| < f_h$ with a bandwidth $(f_h - f_1)$. The general formula for the minimum possible sampling rate for a bandpass signal can be recovered from the samples is given by

$$f_s > \frac{2f_h}{\text{Largest integer not exceeding } \dfrac{f_h}{(f_h - f_l)}}$$

If $f_l = 0$, this reduces to Shannon's sampling theorem. When $f_h = M\,(f_h — f_1)$, where M is an integer, the formula becomes

$$f_s > \frac{2f_h}{M} = 2(f_h - f_1)$$

The absolute minimum sampling rate in the most favourable situation is twice the bandwidth of the signal, not the highest frequency.

In the sampling process discussed so far, the low-pass signals are intrinsically band-pass nature. Now let us consider a scheme called quadrature sampling for the uniform sampling of band-pass signals. This scheme represents a natural extension of the sampling of low-pass signals.

Instead of sampling the band-pass signal directly, we start the sampling operation by preparatory processing of the signal. The band-pass signal is represented by in-phase and quadrature components which may be sampled separately.

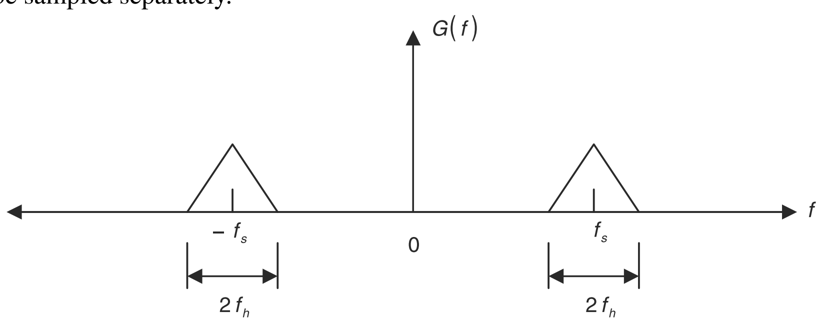


Fig. 1.24(a) *Spectrum of Band-pass g(t)*

Consider a band-pass signal $g(t)$ whose spectrum is limited to be a band-width $2f_h$, centered around the frequency f_s, as shown in Fig. 1.24(a). It is assumed that $f_s > f_h$.

Let $g_1(t)$ denote the in-phase component and $g_2(t)$ denote its quadrature component of the band-pass signal $g(t)$. Then

$$g(t)=g_1(t)\cos(2\pi f_s t)-g_2(t)\sin(2\pi f_s t)$$

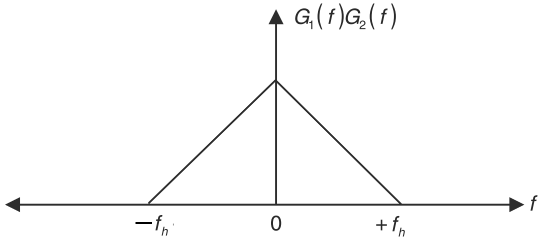


Fig. 1.24(b) *Spectrum of Low Pass in-phase Component $g_1(t)$ and Quadrature Component $g_2(t)$*

The cosine part of a general sinusoid is called the in-phase component, and the sine part is called the quadrature component. The in-phase component $g_1(t)$ and the quadrature component $g_2(t)$ are obtained by multiplying the band pass signal $g(t)$ by cos $(2\pi f_s t)$ and $\sin(2\pi f_s t)$, respectively and then suppressing the sum frequency components by low-pass filters. When $f_s > f_h$, we find that $g_1(t)$ and $g_2(t)$ are both low-pass signals limited to $-f_h < f < f_h$, as shown in Fig. 1.24(b). Accordingly each component may be sampled at the rate of $2f_h$ samples per second. This form of sampling is called quadrature sampling.

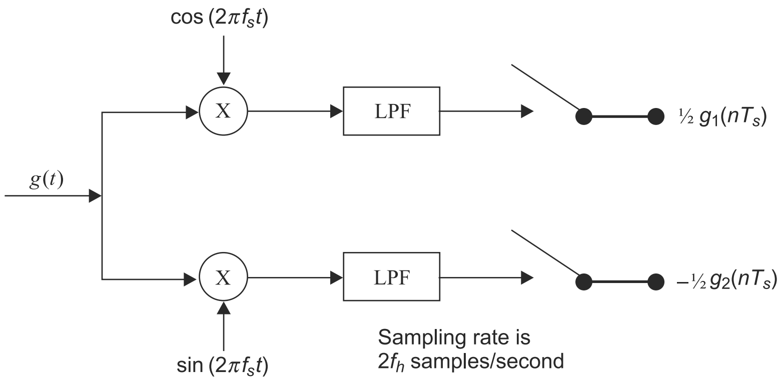


Fig. 1.24(c) *Generation of In-phase and Quadrature Samples from Band-pass Signal $g(t)$*

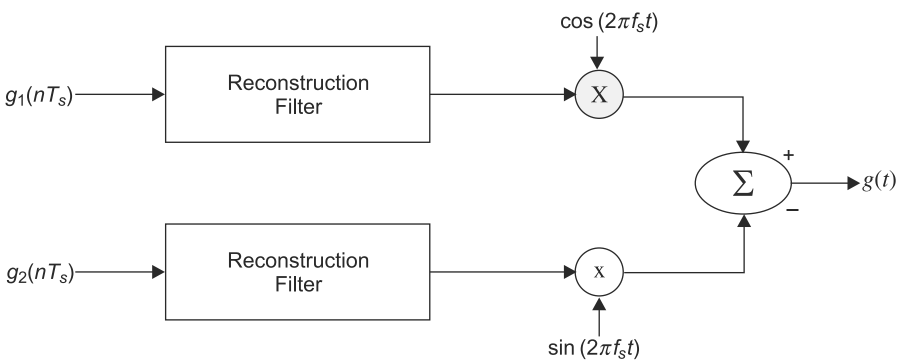


Fig. 1.24(d) *Reconstruction of a Band-pass Signal $g(t)$*

To reconstruct the original band-pass signal from its quadrature sampled version, the in-phase component $g_1(t)$ and the quadrature component $g_2(t)$ are reconstructed from their respective samples. Then they are multiplied by cos $(2\pi f_s t)$ and sin $(2\pi f_s t)$ respectively and the results are added.

The operations that constitute quadrature sampling are shown in Fig. 1.24(c), and those pertaining to reconstruction process are shown in Fig. 1.24(d). Here, the samples of the quadrature component in the generator of Fig. 1.24(c) are the negative of those used as input in the reconstruction circuit of Fig. 1.24(d).

Let us now consider the reconstruction of a stationary message signal from its samples. When a wide sense stationary message signal, whose power spectrum is strictly band-limited, is reconstructed from the sequence of its samples taken at its rate equal to twice the highest frequency component, the reconstructed signal equals the original signal in mean-square sense for all time.

For a wide sense stationary message signal $x(t)$ with auto-correlation function $r_{xx}(\tau)$ and power spectral density $S_x(f)$, we assume that $S_x(f) = 0$ for $|f| \geq f_h$.

An infinite sequence of samples is taken at a uniform rate equal to $2f_h$, that is twice the highest frequency component of the process. Using $x(t)$ to denote the reconstructed process, we have

$$x'(t) = \sum_{n=\infty}^{\infty} x\left(\frac{n}{2f_h}\right) \sin c\,(2f_h t - n) \tag{1.91}$$

where $x\left(\frac{n}{2f_h}\right)$ is the random variable obtained by sampling the message signal $x(t)$ at time $t = \frac{n}{2f_h}$.

The mean square error between the original message process $x(t)$ and the reconstructed message process $x'(t)$ is expressed by

$$\begin{aligned}\xi &= E\left[\left(x(t) - x'(t)\right)^2\right] \\ &= E\left[x^2(t)\right] - 2E\left[x(t)x'(t)\right] + E\left[\left(x'(t)\right)^2\right]\end{aligned} \tag{1.92}$$

The first expectation term on the right side of Eq. (1.92) is the mean square value of $x(t)$, which equals $r_{xx}(0)$. Therefore,

$$E\left[x^2(t)\right] = r_{xx}(0) \tag{1.93}$$

Substituting Eq. (1.91) in the second expectation term of Eq. (1.92), we get

$$E\left[x(t)x'(t)\right] = E\left[x(t) \sum_{n=-\infty}^{\infty} x\left(\frac{n}{2f_h}\right) \sin c\,(2f_h t - n)\right]$$

Interchanging the order of summation and expectation, we have

$$\begin{aligned}E\left[x(t)\,x'(t)\right] &= \sum_{n=-\infty}^{\infty} E\left[x(t)x\left(\frac{n}{2f_h}\right)\right] \sin c(2f_h t - n) \\ &= \sum_{n=-\infty}^{\infty} r_{xx}\left[t - \frac{n}{2f_h}\right] \sin c\,(2f_h t - n)\end{aligned} \tag{1.94}$$

For a wide sense stationary process, the expectation $E[x(t)x'(t)]$ is independent of time t. Hence, for $t = 0$ in the right side of Eq. (1.94), we have

$$r_{xx}\left(-\frac{n}{2f_h}\right)=r_{xx}\left(\frac{n}{2f_h}\right)$$

Therefore, $$E[x(t)x'(t)]=\sum_{n=-\infty}^{\infty} r_{xx}\left(\frac{n}{2f_h}\right)\text{sinc}(-n) \tag{1.95}$$

The term $r_{xx}\left(\frac{n}{2f_h}\right)$ represents a sample of autocorrelation function $r_{xx}(\tau)$ taken at $\tau = n/2f_h$. Since the power spectral density $S_x(f)$ or equivalently the Fourier transform of $r_{xx}(\tau)$ is zero for $|f|> f_h, r_{xx}(\tau)$, $r_{xx}(\tau)$ is represented in terms of its samples taken at $\tau = n/2f_h$ as

$$r_{xx}(\tau)=\sum_{n=-\infty}^{\infty} r_{xx}\left(\frac{n}{2f_h}\right)\text{sinc}(2f_h t-n) \tag{1.96}$$

Similarly, from Eq. (1.95) and Eq. (1.96), we have

$$E[x(t)x'(t)]=r_{xx}(0) \tag{1.97}$$

From Eq. (1.91), the third expectation term on right side of Eq. (1.92) becomes

$$E\left[(x'(t))^2\right]=E\left[\sum_{n=-\infty}^{\infty} x\left(\frac{n}{2f_h}\right)\text{sinc}(2f_h t-n)\sum_{k=-\infty}^{\infty} x\left(\frac{k}{2f_h}\right)\text{sinc}(2f_h t-k)\right]$$

Interchanging the order of expectation and inner summation, we have

$$\begin{aligned}E\left[(x'(t))^2\right]&=\sum_{n=-\infty}^{\infty}\text{sinc}(2f_h t-n)\sum_{k=-\infty}^{\infty} E\left[x\left(\frac{n}{2f_h}\right)x\left(\frac{k}{2f_h}\right)\text{sinc}(2f_h t-k)\right]\\&=\sum_{n=-\infty}^{\infty}\text{sinc}(2f_h t-n)\sum_{k=-\infty}^{\infty} r_{xx}\left(\frac{n-k}{2f_h}\right)\text{sinc}(2f_h t-k)\end{aligned} \tag{1.98}$$

From Eq. (1.96), the inner summation on the right side of Eq. (1.98) equals $r_{xx}\left(t-\frac{n}{2f_h}\right)$, which is simplified as

$$\begin{aligned}E\left[(x'(t))^2\right]&=\sum_{n=-\infty}^{\infty} r_{xx}\left(t-\frac{n}{2f_h}\right)\text{sinc}(2f_h t-n)\\&=r_{xx}(0)\end{aligned} \tag{1.99}$$

Finally, substituting Eq. (1.93), Eq. (1.97) and Eq. (1.99) in Eq. (1.92), we get $\xi = 0$.

Thus the Sampling Theorem states that if a wide sense stationary message signal contains no frequencies higher than f_h hertz, it may be reconstructed from its samples at a sequence of points spaced $\frac{1}{2f_h}$ seconds apart with zero mean squared error, i.e. zero error power.

Example 1.36 *If the continuous-time signal* $x(t) = 2\cos(400\,\pi t) + 5\sin(1200\,\pi t) + 6\cos(4400\,\pi t) + 2\sin(5200\,\pi t)$ *is sampled at a 8 kHz rate generating the sequence* $x[n]$, *find* $x[n]$.

Solution We know that $t=nT=\frac{n}{8000}$. Therefore,

$$x[n]=2\cos\left(\frac{400\pi n}{8000}\right)+5\sin\left(\frac{1200\pi n}{8000}\right)+6\cos\left(\frac{4400\pi n}{8000}\right)+2\sin\left(\frac{5200\pi n}{8000}\right)$$

$$=2\cos\left(\frac{\pi n}{20}\right)+5\sin\left(\frac{3\pi n}{20}\right)+6\cos\left(\frac{11\pi n}{20}\right)+2\sin\left(\frac{13\pi n}{20}\right)$$

$$=2\cos\left(\frac{\pi n}{20}\right)+5\sin\left(\frac{3\pi n}{20}\right)+6\cos\left(\frac{(20-9)\pi n}{20}\right)+\sin\left(\frac{(20-7)\pi n}{20}\right)$$

$$=2\cos\left(\frac{\pi n}{20}\right)+5\sin\left(\frac{3\pi n}{20}\right)+6\cos\left(\frac{9\pi n}{20}\right)-\sin\left(\frac{7\pi n}{20}\right)$$

Example 1.37 *Determine the Nyquist frequency and Nyquist rate for the following signals.*

(a) $x(t) = 50\cos(1000\,\pi t)$ (b) $x(t)=20\,\text{rect}\left(\frac{t}{2}\right)$ (c) $x(t) = 5\,\text{sinc}\,(5t)$

Solution

(a) $X(f)=\frac{50}{2}[\delta(f-500)+\delta(f+500)]$

The highest frequency present in the above signal is 500 Hz. The Nyquist frequency is 500 Hz and the Nyquist rate is 1000 Hz.

(b) $X(f) = 40\,\text{sinc}\,(2f)$

The sinc function does not go to zero and stays there at a finite frequency. Hence, the highest frequency in the signal is infinite and the Nyquist frequency and rate are also infinite. Therefore, the rectangle function is not band limited.

(c) $X(f)=\text{rect}\left(\frac{f}{5}\right)$

The highest frequency existing in $x(t)$ is the value of f at which the rect function has its discontinuous transition from one to zero, $f = 2.5$ Hz. Hence, the Nyquist frequency is 2.5 Hz and the Nyquist rate is 5 Hz.

Example 1.38 *Determine the Nyquist rate for the analog signal given by*

$$x(t)=2\cos 50\pi t+5\sin 300\pi t-4\cos 100\pi t$$

Solution The frequencies existing in the given signal are $f_1 = 25$ Hz, $f_2 = 150$ Hz and $f_3 = 50$ Hz.

Here, the maximum frequency is $f_h = 150$ Hz. We know that according to sampling theorem, $f_s > 2f_h = 300$ Hz.

Therefore the Nyquist rate is $f_N = 300$ Hz.

Example 1.39 *The analog signal is given by* $x(t) = 5\cos 2000\pi t + 3\sin 6000\pi t + 2\cos 12{,}000\pi t$

(*a*) *Determine the Nyquist rate for this signal.*

(*b*) *If the sampling rate* $f_s = 5000$ *samples/s, find the discrete-time signal* $x(n)$ *after sampling.*

Solution

(a) The frequencies existing in the given analog signal are

f_1 = 1 kHz, f_2 = 3 kHz and f_1 = 6 kHz,

Thus f_h = 6 kHz. Hence, according to the sampling theorem, $f_s > 2f_h$ = 12 kHz.

The Nyquist rate is

f_N = 12 kHz,

(b) Since we have chosen f_s = 5 kHz, the folding frequency is

$$\frac{f_s}{2} = 2.5 \text{ kHz}$$

This is the maximum frequency that can be represented uniquely by the sampled signal.

$$x(n) = x(nT) = x\left(\frac{n}{f_s}\right)$$

$$= 5\cos 2\pi\left(\frac{1}{5}\right)n + 3\sin 2\pi\left(\frac{3}{5}\right)n + 2\cos 2\pi\left(\frac{6}{5}\right)n$$

$$= 5\cos 2\pi\left(\frac{1}{5}\right)n + 3\sin 2\pi\left(\frac{3}{5}\right)n + 2\cos 2\pi\left(1+\frac{1}{5}\right)n$$

$$= 5\cos 2\pi\left(\frac{1}{5}\right)n + 3\sin 2\pi\left(-\frac{2}{5}\right)n + 2\cos 2\pi\left(\frac{6}{5}\right)n$$

Therefore $$x(n) = 7\cos 2\pi\left(\frac{1}{5}\right)n - 3\sin 2\pi\left(\frac{3}{5}\right)n$$

Example 1.40 *The transfer of an ideal bandpass filter is given by*

$H(f)$ = 1 *for 45* kHz $\le f \le$ *60* kHz

= 0, otherwise

Determine the minimum samping frequency to avoid aliasing.

Solution We know that the sampling frequency is $f_s \ge 2\,(f_n - f_1)$

Therefore, the minimum sampling f_s = 2 (60 – 5) kHz = 30 kHz

Example 1.41 *An analog signal x(t) has a band-limited spectrum $X_a(j\Omega)$ with $\Omega_1 = 150\pi$ and $\Omega_2 = 200\pi$. Determine the smallest sampling rate that can be employed to sample x(t) so that it can be fully recovered from its sampled version x[n]. Sketch the Fourier transform $X_p(j\Omega)$ of x[n] and the frequency response of the ideal reconstruction filter needed to fully recover x(t).*

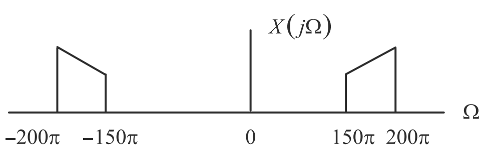


Fig. E1.41(a)

Solution $\Omega_2 = 200\pi$, $\Omega_1 = 150\pi$. Thus, $\Delta\Omega = \Omega_2 - \Omega_1 = 50\pi$. Note $\Delta\Omega$ is an integer multiple of Ω_2. Hence we choose the sampling angular frequency as

$$\Omega_T = 2\Delta\Omega = 2(\Omega_2 - \Omega_1) = 100\pi = \frac{2\times 200\pi}{M}$$, which

is satisfied for $M = 4$. The sampling frequency is therefore $F_T = 50$ Hz.

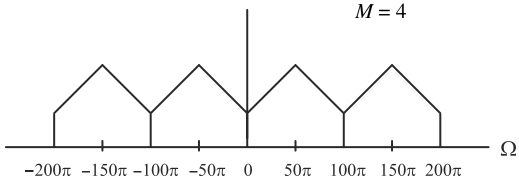


Fig. E1.41(b)

1.8.4 Signal Quantisation and Encoding

A discrete-time signal with continuous-valued amplitudes is called a *sampled data signal*, whereas a continuous-time signal with discrete-valued amplitudes is referred to as a quantised boxcar signal. Quantisation is a process by which the amplitude of each sample of a signal is rounded off to the nearest permissible level. That is, quantisation is conversion of a discrete-time continuous-amplitude signal into a discrete-time, discrete-valued signal. Then encoding is done by representing each of these permissible levels by a digital word of fixed wordlength.

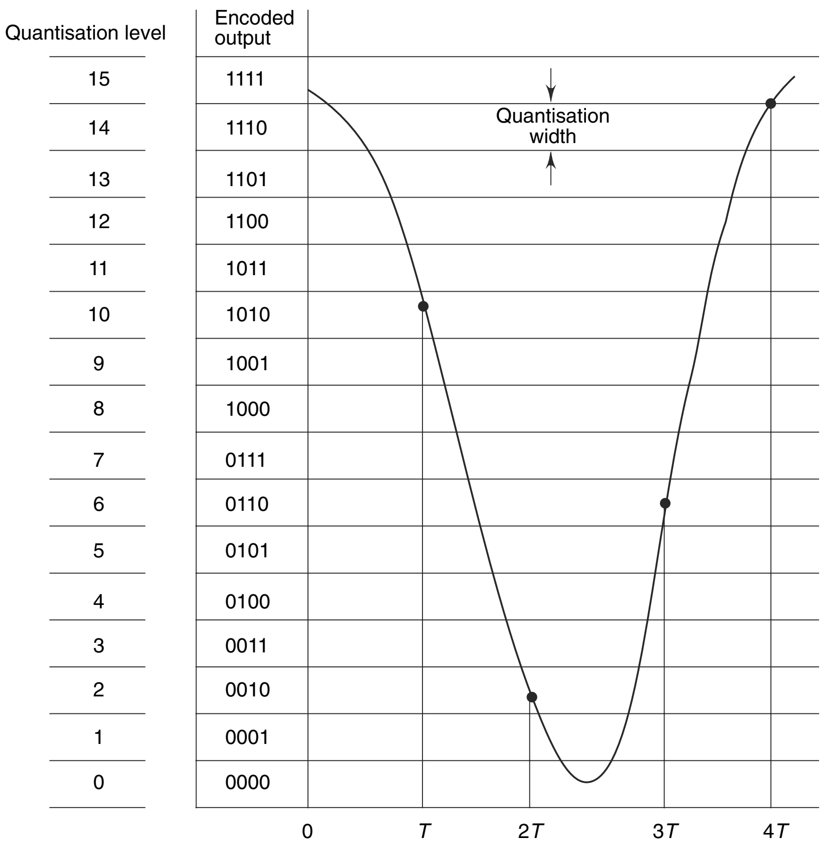


Fig. 1.25 *Quantising and Encoding*

The process of quantisation introduces an error called *quantisation error* and it is simply the difference between the value of the analog input and the analog equivalent of the digital representation. This error will be small if there are more permissible levels and the width of these quantisation levels is very small. In the analog-to-digital conversion process, the only source of error is the quantiser. Even if there are more quantisation levels, error can occur if the signal is at its maximum or minimum value for significant time intervals. Figure 1.25 shows how a continuous-time signal is quantised in a quantiser that has 16 quantising levels.

REVIEW QUESTIONS

1.1 Define unit-impulse function.

1.2 What is a unit-step function? How can it be obtained from a unit-impulse function?

1.3 What is unit-ramp function? How can it be obtained from a unit-impulse function?

1.4 What is pulse function?

1.5 Define exponential sequence.

1.6 Define sinusoidal sequence.

1.7 Write the mathematical and graphical representation of a unit sample sequence.

1.8 Give the mathematical and graphical representation of a unit-step sequence.

1.9 Evaluate

(a) $\int_{-\infty}^{\infty} e^{-at^2}\delta(t-10)\,dt$ (b) $\int_{-\infty}^{\infty} e^{-2t^2}\delta(t+5)\,dt$

(c) $\int_{-\infty}^{\infty} 40e^{-2t^2}\delta(t-10)\,dt$ and (d) $\int_{-\infty}^{\infty} e^{-2t^2}\delta(t-10)\,dt$

Ans:

(a) e^{-100a} (b) 0 (c) $40\,e^{-200}$ (d) e^{-200}

1.10 What are the major classifications of signals?

1.11 With suitable examples distinguish a deterministic signal from a random signal.

1.12 What are periodic signals? Give examples.

1.13 Describe the procedure used to determine whether the sum of two periodic signals is periodic or not.

1.14 Determine which of the following signals are periodic and determine the fundamental period.

(a) $x_1(t) = 10\sin 25\pi t$ (b) $x_2(t) = 10\sin\sqrt{5}\pi t$

(c) $x_3(t) = \cos 10\pi t$ (d) $x_4(t) = x_1(t) + x_2(t)$

(e) $x_5(t) = x_1(t) + x_3(t)$ (f) $x_4(t) = x_2(t) + x_3(t)$

1.15 Check whether each of the following signals is periodic or not. If a signal is periodic, determine its fundamental period.

(a) $x(t) = \cos\left(2t + \frac{\pi}{4}\right)$ (b) $x(t) = \cos^2 t$

(c) $x(t) = (\cos 2\pi t)\,u(t)$ (d) $x(t) = e^{j\pi t}$

(e) $x[n] = e^{j[(n/4)-\pi]}$ (f) $x[n] = \cos\left(\frac{\pi n^2}{8}\right)$

(g) $x[n] = \cos\left(\frac{n}{2}\right)\cos\left(\frac{\pi n}{4}\right)$ (h) $x[n] = \cos\left(\frac{\pi n}{4}\right) + \sin\left(\frac{\pi n}{8}\right) - 2\cos\left(\frac{\pi n}{2}\right)$

Ans:

(a) Periodic with period = π (b) Periodic with period = π
(c) Non-periodic (d) Periodic with period = 2
(e) Non-periodic (f) Periodic with period = 8
(g) Non-periodic (h) Periodic with period = 16

1.16 What are even signals? Give examples.

1.17 What are odd signals? Give examples.

1.18 Find the even and odd components of the following signals.

(a) $x(t) = u(t)$ (b) $x(t) = \sin\left(\omega_0 t + \frac{\pi}{4}\right)$
(c) $x[n] = \delta[n]$

Ans:

(a) $x_e(t) = \frac{1}{2},\ x_0(t) = \frac{1}{2}\sin t$ (b) $x_e(t) = \frac{1}{\sqrt{2}}\cos\omega_0 t, x_0(t) = \frac{1}{\sqrt{2}}\sin\omega_0 t$

(c) $x_e[n] = \delta[n],\ x_0[n] = 0$

1.19 What is an energy signal?

1.20 What is a power signal?

1.21 Check whether the given signal is power signal or energy signal and find its value.

$$x[n] = \begin{cases} 3(-1)^n, & n \geq 0 \\ 0, n < 0. \end{cases}$$

Ans: Power signal with 4.5 W.

1.22 What are singularity functions?

1.23 With illustrations, explain shifting, folding and time scaling operations on discrete-time signals.

1.24 Write down the corresponding equation of the signals shown in Fig. Q1.24 in terms of unit step function.

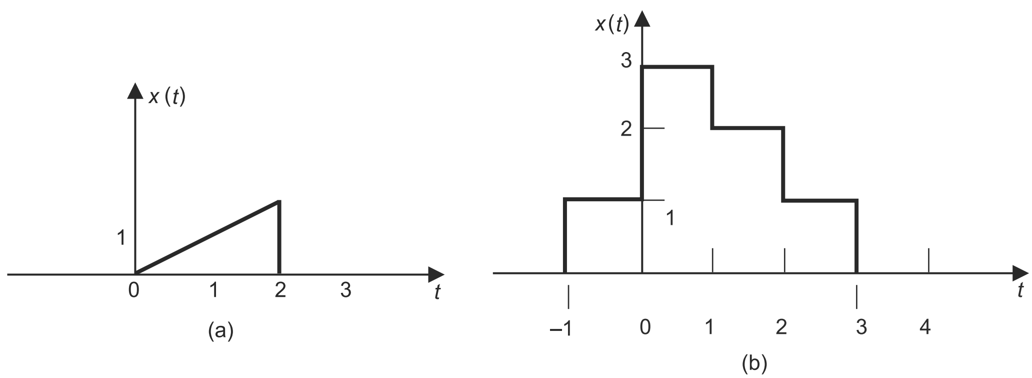


Fig. Q1.24

Ans: (a) $x(t) = \frac{t}{2}[u(t) - u(t-2)]$

(b) $x(t) = u(t+1) + 2u(t) - u(t-1) - u(t-2) - u(t-3)$

1.25 Write down the corresponding equation of the sequences shown in Fig. Q1.25 (a) in terms of unit step functions.

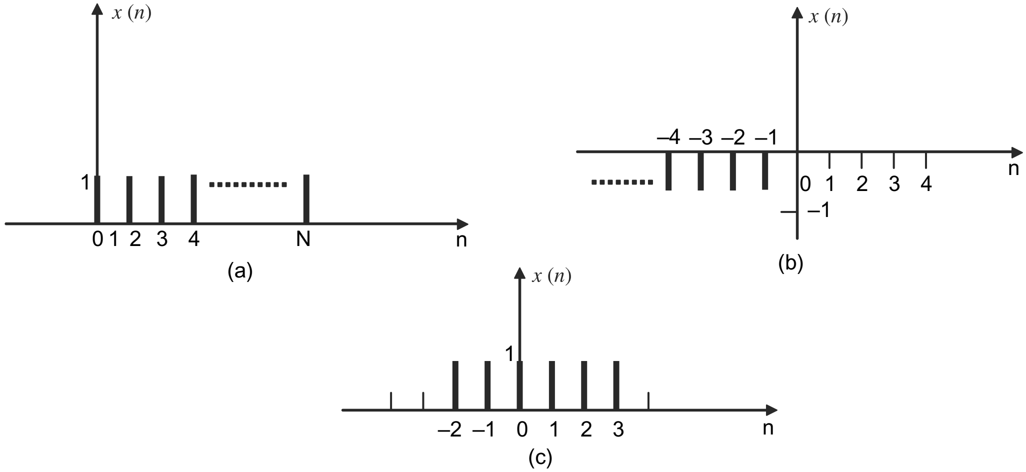


Fig. Q1.25

Ans:

(a) $x[n] = u[n] - u[n-(N+1)]$

(b) $x[n] = -u[-n-1]$

(c) $x[n] = u[n+2] - u[n-3]$

1.26 Write down the corresponding equation for the signal shown in Fig. Q1.26.

Ans: $x(t) = 4[u(t) - u(2-t)] + 2[u(t-4) - u(6-t)]$

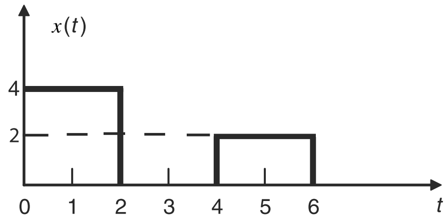


Fig. Q1.26

1.27 What are $\delta[2n]$ and $\delta[3t]$?

Ans: $\delta[n]$, $\frac{1}{3}\delta(t)$

1.28 Using the discrete-time signals $x_1(n)$ and $x_2(n)$ shown in Fig. Q1.28, represent the following signals by a graph and by a sequence of numbers.

(a) $y_1(n) = x_1(n) + x_2(n)$; (b) $y_2(n) = 2x_1(n)$; (c) $y_3(n) = x_1(n)x_2(n)$

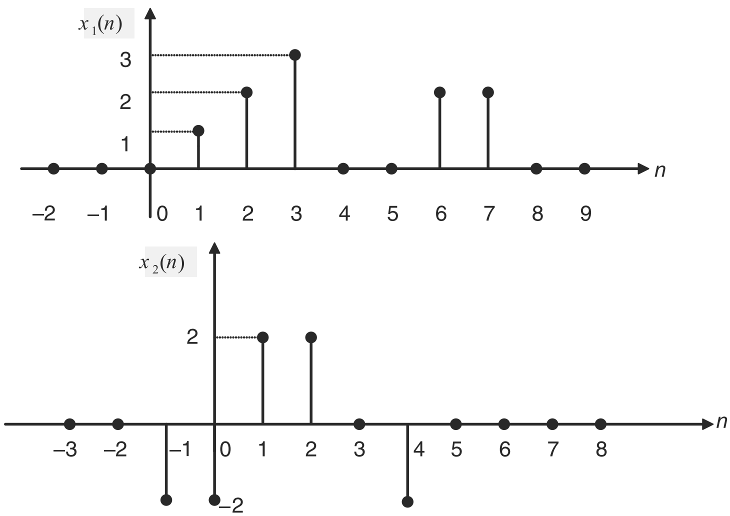


Fig. Q1.28

Ans:

(a) $y_1(n) = ..., 0, -2, -2, 3, 4, 3 - 2, 0, 2, 2, 0, ...$
$\uparrow$

(b) $y_2(n) = \{..., 0, 2, 4, 6, 0, 0, 4, 4, 0, ...\}$
$\uparrow$

(c) $y_2(n) = \{..., 0, 2, 4, 0, ...\}$
$\uparrow$

1.29 Explain the terms single-sided spectrum and double-sided spectrum with respect to a signal.

1.30 Sketch the single-sided and double-sided frequency spectra of the signals

(a) $x_1(t) = 10\sin\left(10\pi t - \frac{2\pi}{3}\right), -\infty < t < \infty$

(b) $x_2(t) = 25\cos\left(5\pi t - \frac{\pi}{2}\right), -\infty < t < \infty$

(c) $x_3(t) = 100\sin\left(10\pi t - \frac{2\pi}{3}\right) + 50\cos\left(25\pi t - \frac{\pi}{3}\right), -\infty < t < \infty$

1.31 How are systems classified?

1.32 Distinguish static systems from dynamic systems.

1.33 What is a linear system?

1.34 Determine whether the following systems are linear

1.35 Show that the following systems are linear

(i) $y(n) - ny(n-1) = x(n)$; (ii) $y(n+2) + 2y(n) = x(n+1)$

1.36 What is LTI system?

1.37 Give an example of a linear time-varying system such that with a periodic input the corresponding output is not periodic.

Ans:

$y[n] = F\{x[n]\} = nx[n]s$

1.38 Show that the following systems are nonlinear and time invariant.

(a) $y(n) - x(n)\, y(n-1) = x(n)$

(b) $y(n+2) + 2y(n) = x(n+1) + 2$

1.39 Prove that the following system is time invariant

$$y(n-1) - x(n-1)\, y(n-2) = x(n)$$

1.40 Prove that the following system is time-variant.

$$y(n+2) + ny(n) = x(n+1)$$

1.41 What is a causal system? Why are non-causal systems unrealisable?

1.42 What is BIBO stability?

1.43 What are the conditions for BIBO stability?

1.44 State the condition for a system to be invertible and explain the same.

1.45 Express the given sequence in terms of unit step response for

$$x[n] = \{1, 3, -2, 4\}, 0 \le n \le 3$$

Ans: $x(n) = u(n) + 3u(n-1) - 2u(n-2) + 4u(n-3)$

1.46 Express the given sequence in terms of unit sample response,

Ans: $x[n] = \{1, -2, 4, \underset{\uparrow}{-6}, 3, -2, 4\}$ for $-3 \le n \le 3$

$$x(n) = \delta(n+3) - 2\delta(n+2) + 4\delta(n+1) - 6\delta(n) + 3\delta(n-1) - 2\delta(n-2) + 4\delta(n-3)$$

1.47 What are the different ways of representing a system?

1.48 Explain how difference/differential equations are used to model a system.

1.49 Explain how impulse response can model a system.

1.50 Discuss the state-variable modelling of a system.

1.51 Explain the following terms (i) state variable (ii) state space (iii) state vector (iv) trajectory (v) state equations and (vi) output equations.

1.52 State and derive the state space representation for n-first order continuous-time LTI system with m-inputs and k-outputs. Also, find the transfer function.

1.53 State and derive the state space representation of the system with the state variables $x_1(n)$ and $x_2(n)$. Also find the system transfer function $H(z)$.

1.54 State and derive an expression for multiple-input and multiple-output system.

1.55 Determine the state variable model for a system described by the following differential equation $\dddot{y}(t) + \ddot{y}(t) + \dot{y}(t) + 4y(t) = 4$

1.56 Determine the state variable model for the transfer function given by $H(s) = \dfrac{Y(s)}{U(s)} = \dfrac{5(s+4)}{s(s+1)(s+3)}$

1.57 Find the state transition matrix for $A = \begin{bmatrix} 0 & 1 \\ -6 & -5 \end{bmatrix}$

1.58 A system is described by the following state-variable model

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -3 & -2 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = [4\ 6\ 2]x(t)$$

Determine the transfer function $H(s) = Y(s)/U(s)$

1.59 Obtain a state model for the following

(a) $\dddot{y}(t) + 2\ddot{y}(t) + 6\dot{y}(t) + 7y(t) = 4u(t)$, (b) $\dfrac{Y(s)}{X(s)} = \dfrac{s^2+3s+2}{s(s+1)(s+3)}$

1.60 A discrete-time LTI system is specified by the difference equation $y[n] + y[n-1] - 6y[n-2] = 2x[n-1] + x[n-2]$. Determine its state space representation.

1.61 With a block diagram explain the process of analog-to-digital conversion.

1.62 What is meant by sampling? State the sampling theorem.

1.63 Explain how sampling can be done with an impulse function.

1.64 Draw the spectrum of a sampled signal and explain aliasing.

1.65 Explain the process of reconstruction of the signal from its samples. Obtain the impulse response of an ideal reconstruction filter.

1.66 What is meant by quantisation and encoding?

1.67 What is a quantised boxcar signal?

Chapter 2 Fourier Analysis of Periodic and Aperiodic Continuous-Time Signals and Systems

INTRODUCTION 2.1

A signal which is repetitive is a periodic function of time. Any periodic function of time $f(t)$ can be represented by an infinite series called the ***Fourier Series***. A function of time $f(t)$ is said to be periodic of period T if $f(t) = f(t + T)$ for all t. For example, the periodic waveforms of sinusoidal and exponential forms are shown in Fig. 2.1.

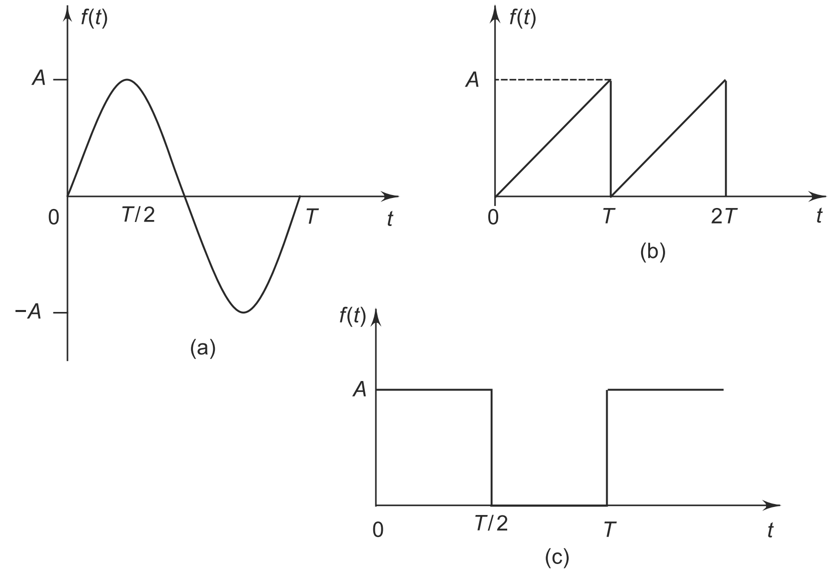


Fig. 2.1 *Waveforms Representing Periodic Functions*

Examples of periodic processes are the vibration of a tuning fork, oscillations of a pendulum, conduction of heat, alternating current passing through a circuit, propagation of sound in a medium, etc. Fourier series may be used to represent either functions of time or functions of space co-ordinates. In a similar manner, functions of two and three variables may be represented as double and triple Fourier series respectively. Periodic waveforms may be expressed in the form of Fourier series. Non-periodic waveforms may be expressed by Fourier transforms.

TRIGONOMETRIC FOURIER SERIES 2.2

A periodic function $f(t)$ can be expressed in the form of trigonometric series as

$$f(t) = \frac{1}{2}a_0 + a_1\cos\omega_0 t + a_2\cos 2\omega_0 t + a_3\cos 3\omega_0 t + \cdots$$
$$+ b_1\sin\omega_0 t + b_2\sin 2\omega_0 t + b_3\sin 3\omega_0 t + \cdots \tag{2.1}$$

where $\omega_0 = 2\pi f = \frac{2\pi}{T}$, f is the frequency and a's and b's are the coefficients. The Fourier series exists only when the function $f(t)$ satisfies the following three conditions called ***Dirichlet's conditions.***

(i) $f(t)$ is well defined and single-valued, except possibly at a finite number of points, i.e. $f(t)$ has a finite average value over the period T.
(ii) $f(t)$ must possess only a finite number of discontinuities in the period T.
(iii) $f(t)$ must have a finite number of positive and negative maxima in the period T.

Equation 2.1 may be expressed by the Fourier series

$$f(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n \cos n\ \omega_0 t + \sum_{n=1}^{\infty} b_n \sin n\omega_0 t \tag{2.2}$$

where a_n and b_n are the coefficients to be evaluated. Integrating Eq. (2.2) for a full period, we get

$$\int_{-T/2}^{T/2} f(t)\,\mathrm{d}t = \frac{1}{2}a_0 \int_{-T/2}^{T/2} \mathrm{d}t + \int_{-T/2}^{T/2} \sum_{n=1}^{\infty} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t)\,\mathrm{d}t$$

Integration of cosine or sine function for a complete period is zero.

Therefore, $$\int_{-T/2}^{T/2} f(t)\,\mathrm{d}t = \frac{1}{2}a_0 T$$

Hence, $$a_0 = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\,\mathrm{d}t \tag{2.3}$$

or equivalently $a_0 = \frac{2}{T}\int_0^T f(t)\,\mathrm{d}t$

Multiplying both sides of Eq. (2.2) by $\cos m\omega_0 t$ and integrating, we have

$$\int_{-T/2}^{T/2} f(t)\cos m\omega_0 t\,\mathrm{d}t = \frac{1}{2}\int_{-T/2}^{T/2} a_0 \cos m\omega_0 t\,\mathrm{d}t +$$
$$\int_{-T/2}^{T/2} \sum_{n=1}^{\infty} a_n \cos n\omega_0 t \cos m\omega_0 t\,\mathrm{d}t + \int_{-T/2}^{T/2} \sum_{n=1}^{\infty} b_n \sin n\omega_0 t \cos m\omega_0 t\,\mathrm{d}t$$

Here, $$\frac{1}{2}\int_{-T/2}^{T/2} a_0 \cos m\omega_0 t\,\mathrm{d}t = 0$$

$$\int_{-T/2}^{T/2} a_n \cos n\omega_0 t \cos m\omega_0 \mathrm{d}t = \frac{a_n}{2}\int_{-T/2}^{T/2} [\cos(m+n)\omega_0 t + \cos(m-n)\omega_0 t]\mathrm{d}t$$

$$= \begin{cases} 0, & \text{for } m \neq n \\ \dfrac{T}{2}a_n, & \text{for } m = n \end{cases}$$

$$\int_{-T/2}^{T/2} b_n \sin n\omega_0 t \cos m\omega_0 t\mathrm{d}t = \frac{b_n}{2}\int_{-T/2}^{T/2} [\sin(m+n)\omega_0 t - \sin(m-n)\omega_0 t]\mathrm{d}t$$

$$= 0$$

Therefore, $\displaystyle\int_{-T/2}^{T/2} f(t)\cos n\omega_0 t\mathrm{d}t = \frac{Ta_n}{2}, \quad \text{for } m = n$

Hence, $$a_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\cos n\omega_0 t\,\mathrm{d}t \tag{2.4}$$

or equivalently $$a_n = \frac{2}{T}\int_{0}^{T} f(t)\cos n\omega_0 t\,\mathrm{d}t$$

Similary, multiplying both sides of Eq. (2.2) by sin $m\omega_0 t$ and integrating, we get

$$\int_{-T/2}^{T/2} f(t)\sin m\omega_0 t\,\mathrm{d}t = \frac{1}{2}\int_{-T/2}^{T/2} a_0 \sin m\omega_0 t\,\mathrm{d}t +$$

$$\int_{-T/2}^{T/2}\sum_{n=1}^{\infty} a_n \cos n\omega_0 t \sin m\omega_0 t\,\mathrm{d}t + \int_{-T/2}^{T/2}\sum_{n=1}^{\infty} b_n \sin n\omega_0 t \sin m\omega_0 t\,\mathrm{d}t$$

Here, $\displaystyle\frac{1}{2}\int_{-T/2}^{T/2} a_0 \sin m\omega_0 t\,\mathrm{d}t = 0$

$$\int_{-T/2}^{T/2} a_n \cos n\omega_0 t \sin m\omega_0 t\,\mathrm{d}t = 0$$

$$\int_{-T/2}^{T/2} b_n \sin n\,\omega_0 t \sin m\omega_0 t\,\mathrm{d}t = \begin{cases} 0, & \text{for } m \neq n \\ \dfrac{T}{2}b_n, & \text{for } m = n \end{cases}$$

Therefore, $\displaystyle\int_{-T/2}^{T/2} f(t)\sin m\omega_0 t\,\mathrm{d}t = \frac{T}{2}b_n, \quad \text{for } m = n$

Hence, $$b_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\sin n\omega_0 t\,\mathrm{d}t \tag{2.5}$$

or equivalently $\displaystyle b_n = \frac{2}{T}\int_{0}^{T} f(t)\sin n\omega_0 t\,\mathrm{d}t$

The number n = 1, 2, 3, ... gives the values of the harmonic frequencies.

Symmetry Conditions

(i) If the function $f(t)$ is even, then $f(-t) = f(t)$. For example, cos t, t^2, t sin t, are all even. The cosine is an even function, since it may be expressed as the power series

$$\cos t = 1 - \frac{t^2}{2!} + \frac{t^4}{4!} - \frac{t^6}{6!} + \cdots$$

The waveforms representing the even functions of t are shown in Fig. 2.2. Geometrically, the graph of an even function will be symmetrical with respect to the y-axis and only cosine terms are present (d.c. term optional). When $f(t)$ is even,

$$\int_{-a}^{a} f(t)\mathrm{d}t = 2\int_{0}^{a} f(t)\mathrm{d}t$$

The sum or product of two or more even function is an even function.

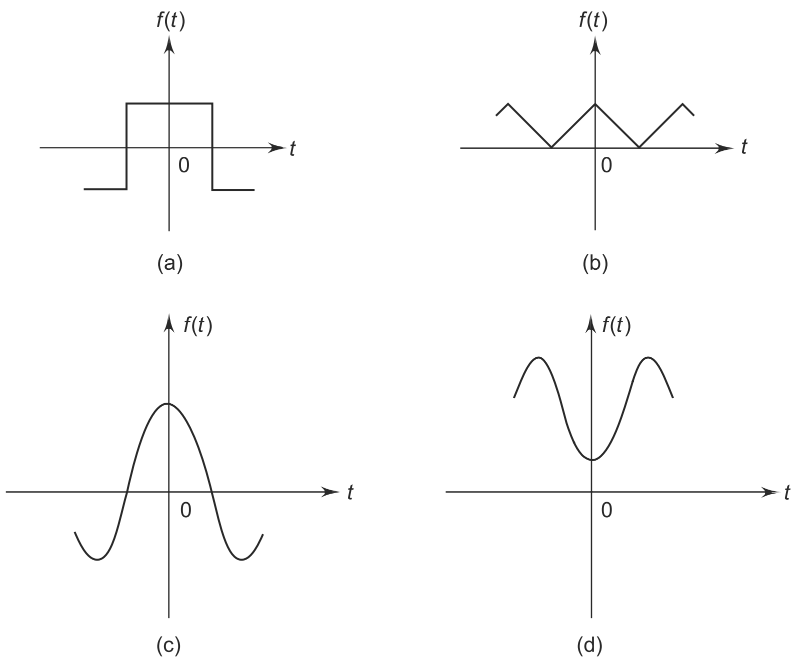


Fig. 2.2 *Waveforms Representing Even Functions*

(ii) If the function $f(t)$ is odd, then $f(-t) = -f(t)$ and only sine terms are present (d.c. term optional). For example, sin t, t^3, t cos t are all odd. The waveforms shown in Fig. 2.3 represent odd functions of t. The graph of an odd function is symmetrical about the origin. If $f(t)$ is odd, $\int_{-a}^{a} f(t)\mathrm{d}t = 0.$ The sum of two or more odd functions is an odd function and the product of two odd functions is an even function.

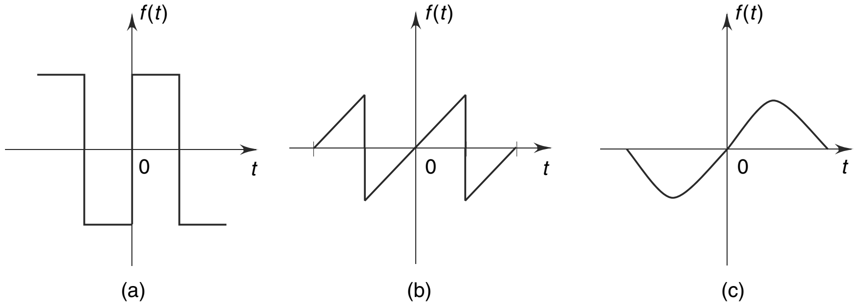


Fig. 2.3 *Waveforms Representing Odd Functions*

(iii) If $f(t+T/2)=f(t)$, only even harmonics are present.

(iv) If $f(t+T/2)=-f(t)$, only odd harmonics are present and hence the waveform has half-wave symmetry.

Example 2.1 *Obtain the Fourier components of the periodic square wave signal which is symmetrical with respect to the vertical axis at time $t = 0$, as shown in Fig. E2.1.*

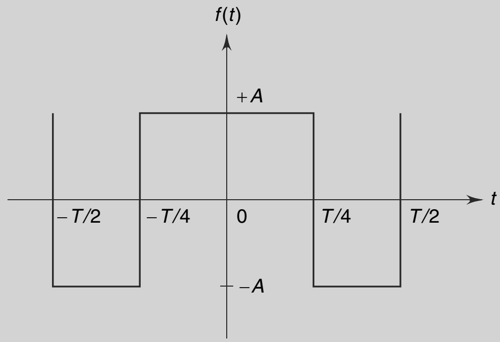


Fig. E2.1

Solution Since the given waveform is symmetrical about the horizontal axis, the average area is zero and hence the d.c. term $a_0 = 0$. In addition, $f(t) = f(-t)$ and so only cosine terms are present, i.e., $b_n = 0$.

Now,
$$a_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\cos n\omega_0 t\,dt$$

where
$$f(t) = \begin{cases} -A, & \text{from} \quad -T/2 < t < -T/4 \\ +A, & \text{from} \quad -T/4 < t < +T/4 \\ -A, & \text{from} \quad +T/4 < t < +T/2 \end{cases}$$

Therefore,
$$a_n = \frac{2A}{T}\left[\int_{-T/2}^{-T/4} (-\cos n\omega_0 t)dt + \int_{-T/4}^{T/4} (\cos n\omega_0 t)dt + \int_{T/4}^{T/2} (-\cos n\omega_0 t)dt\right]$$

$$= \frac{2A}{T}\left[\left(\frac{-\sin n\omega_0 t}{n\omega_0}\right)_{-T/2}^{-T/4} + \left(\frac{\sin n\omega_0 t}{n\omega_0}\right)_{-T/4}^{T/4} + \left(\frac{-\sin n\omega_0 t}{n\omega_0}\right)_{T/4}^{T/2}\right]$$

$$= \frac{2A}{n\omega_0 T}\left[-\sin\left(\frac{-n\omega_0 T}{4}\right)+\sin\left(\frac{-n\omega_0 T}{2}\right)+\sin\left(\frac{n\omega_0 T}{4}\right)\right.$$

$$\left.-\sin\left(\frac{-n\omega_0 T}{4}\right)-\sin\left(\frac{n\omega_0 T}{2}\right)+\sin\left(\frac{n\omega_0 T}{4}\right)\right]$$

$$= \frac{8A}{n\omega_0 T}\sin\left(\frac{n\omega_0 T}{4}\right)-\frac{4A}{n\omega_0 T}\sin\left(\frac{n\omega_0 T}{2}\right)$$

When $\omega_0 T = 2\pi$, the second term is zero for all integer of n.

Hence, $$a_n = \frac{8A}{2n\pi}\sin\left(\frac{n\pi}{2}\right) = \frac{4A}{n\pi}\sin\left(\frac{n\pi}{2}\right)$$

$$a_n = 0\,(\text{d.c. term})$$

$$a_1 = \frac{4A}{\pi}\sin\left(\frac{\pi}{2}\right) = \frac{4A}{\pi}$$

$$a_2 = \frac{4A}{\pi}\sin(\pi) = 0$$

$$a_3 = \frac{4A}{3\pi}\sin\left(\frac{3\pi}{2}\right) = -\frac{4A}{3\pi}$$

$$\vdots$$

Substituting the values of the coefficients in Eq. (2.2), we get

$$f(t) = \frac{4A}{\pi}\left[\cos(\varphi_0 t) - \frac{1}{3}\cos(3\varphi_0 t) + \frac{1}{5}\cos(5\varphi_0 t) - \cdots\right]$$

Example 2.2 *Obtain the Fourier Components of the periodic rectangular waveform shown in Fig. E2.2.*

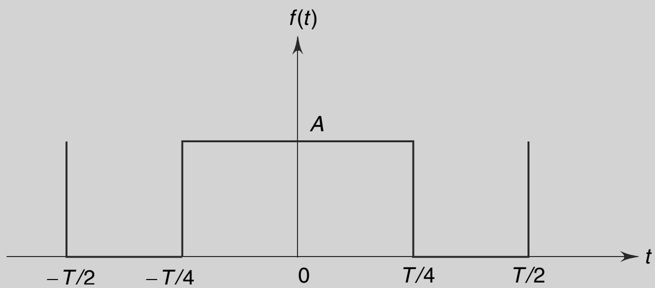


Fig. E2.2

Solution The given waveform for one period can be written as

$$f(t) = \begin{cases} 0, & \text{for } -T/2 < t < -T/4 \\ A, & \text{for } -T/4 < t < T/4 \\ 0, & \text{for } T/4 < t < T/2 \end{cases}$$

For the given waveform, $f(-t) = f(t)$ and hence it is an even function and has $b_n = 0$.

The value of the d.c. term is

$$a_0 = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\,\mathrm{d}t = \frac{2}{T}\int_{-T/4}^{T/4} A\,\mathrm{d}t = \frac{2A}{T}\times\frac{T}{2} = A$$

$$a_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\cos n\omega_0 t\,\mathrm{d}t$$

$$= \frac{2}{T}\int_{-T/4}^{T/4} A\cos n\omega_0 t\,\mathrm{d}t = \frac{2A}{T}\left[\frac{\sin n\omega_0 t}{n\omega_0}\right]_{-T/4}^{T/4} = \frac{4A}{n\omega_0 T}\sin(n\omega_0 T/4)$$

When $\omega_0 T = 2\pi$, we have

$$a_n = \frac{2A}{n\pi}\sin\left(\frac{n\pi}{2}\right)$$

$$= 0, \quad \text{for } n = 2, 4, 6, \ldots$$

$$= \frac{2A}{n\pi}, \quad \text{for } n = 1, 5, 9, 13, \ldots$$

$$= -\frac{2A}{n\pi}, \quad \text{for } n = 3, 7, 11, 15, \ldots$$

Substituting the values of the coefficients in Eq. (2.2), we obtain

$$f(t) = \frac{A}{2} + \frac{2A}{\pi}\left(\cos\omega_0 t - \frac{1}{3}\cos 3\omega_0 t + \frac{1}{5}\cos 5\omega_0 t - \cdots\right)$$

Example 2.3 *Obtain the trigonometric Fourier series for the half-wave rectified sine wave shown in Fig. E2.3.*

Solution As the waveform shows no symmetry, the series may contain both sine and cosine terms. Here $f(t) = A\sin\omega_0 t$.

To evaluate a_0

$$a_0 = \frac{2}{T}\int_0^T A\sin\omega_0 t\,\mathrm{d}t = \frac{2}{T}\int_0^{T/2} A\sin\omega_0 t\,\mathrm{d}t$$

$$= \frac{2A}{\omega_0 T}[-\cos\omega_0 t]_0^{T/2} = \frac{2A}{\omega_0 T}[-\cos(\omega_0 T/2) + 1]$$

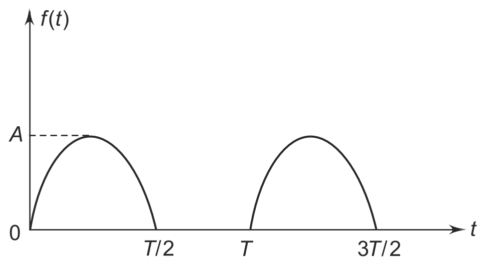


Fig. E2.3

Substituting $\omega_0 T = 2\pi$, we have $a_0 = \frac{2A}{\pi}$.

To evaluate a_n

$$a_n = \frac{2}{T}\int_0^T f(t)\cos n\omega_0 t\,\mathrm{d}t = \frac{2}{T}\int_0^{T/2} A\sin\omega_0 t\cos n\omega_0 t\,\mathrm{d}t$$

$$= \frac{2A}{\omega_0 T}\left[\frac{-n\sin\omega_0 t\sin n\omega_0 t - \cos n\omega_0 t\cos\omega_0 t}{-n^2 + 1}\right]_0^{T/2}$$

Substituting $\omega_0 T = 2\pi$, we have

$$a_n = \frac{A}{\pi(1-n^2)}[\cos n\pi + 1]$$

Hence, $$a_n = \frac{2A}{\pi(1-n^2)}, \quad \text{for } n \text{ even}$$

$$= 0, \quad \text{for } n \text{ odd}$$

For $n = 1$, this expression is infinite and hence we have to integrate separately to evaluate a_1.

Therefore, $$a_1 = \frac{2}{T}\int_0^{T/2} A\sin\omega_0 t\cos\omega_0 t\,dt = \frac{A}{T}\int_0^{T/2}\sin 2\omega_0 t\,dt = \frac{A}{2\omega_0 T}[-\cos 2\omega_0 t]_0^{T/2}$$

When $\omega_0 T = 2\pi$, we have $a_1 = 0$.

To find b_n $$b_n = \frac{2}{T}\int_0^T f(t)\sin n\omega_0 t\,dt = \frac{2}{T}\int_0^{T/2} A\sin\omega_0 t\sin n\omega_0 t\,dt$$

$$= \frac{2A}{\omega_0 T}\left[\frac{n\sin\omega_0 t\cos n\omega_0 t - \sin n\omega_0 t\cos\omega_0 t}{-n^2+1}\right]_0^{T/2}$$

When $\omega_0 T = 2\pi$, we have $b_n = 0$.
For $n = 1$, the expression is infinite and hence b_1 has to be calculated as

$$b_1 = \frac{2A}{T}\int_0^{T/2}\sin^2\omega_0 t\,dt = \frac{2A}{\omega_0 T}\left[\frac{\omega_0 t}{2} - \frac{\sin 2\omega_0 t}{4}\right]_0^{T/2}$$

When $\omega_0 T = 2\pi$, we have $b_1 = \frac{A}{2}$.

Substituting the values of the coefficients in Eq. (2.2), we get

$$f(t) = \frac{A}{\pi}\left\{1 + \frac{\pi}{2}\sin\omega_0 t - \frac{2}{3}\cos 2\omega_0 t - \frac{2}{15}\cos 4\omega_0 t - \cdots\right\}$$

Example 2.4 *Obtain the trigonometric Fourier series of the triangular waveform shown in Fig. E2.4.*

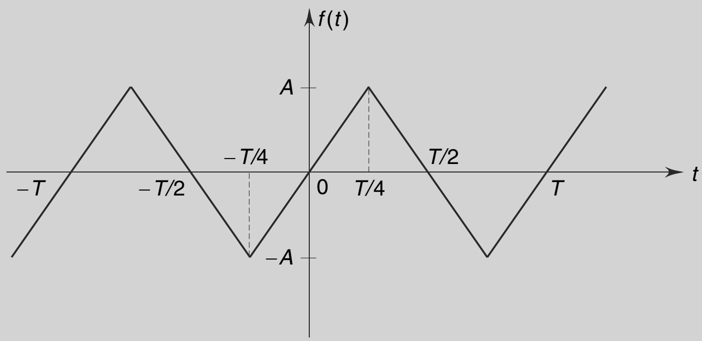


Fig. E2.4

Solution

(i) As the waveform has equal positive and negative area in one cycle, the average value of $a_0 = 0$.

(ii) As $f(t) = -f(t)$, it is an odd function and hence $a_n = 0$ and $b_n = \frac{4}{T}\int_0^{T/2} f(t)\sin n\omega_0 t\,dt$

(iii) Here, $f(t \pm T/2) = -f(t)$. Hence it has half-wave odd symmetry and $a_n = b_n = 0$ for n even.

(iv) To find $f(t)$ for the given waveform

The equation of a straight line is $\frac{y - y_1}{x - x_1} = \frac{y_1 - y_2}{x_1 - x_2}$

For the region $0 < t < \frac{T}{4}$

$$\frac{f(t) - 0}{t - 0} = \frac{0 - A}{0 - T/4}$$

Therefore, $f(t) = \frac{4A}{T}t$

For the region, $\frac{T}{4} < t < \frac{T}{2}$,

$$\frac{f(t) - A}{t - \frac{T}{4}} = \frac{A - 0}{\frac{T}{4} - \frac{T}{2}}$$

$$f(t) - A = -\frac{4A}{T}\left(t - \frac{T}{4}\right) = -\frac{4A}{T}t + A$$

Therefore, $f(t) = \frac{4A}{T}t + 2A.$

Now,
$$b_n = \frac{4}{A}\int_0^{T/2} f(t)\sin n\omega_0 t\,dt$$

$$= \frac{4}{T}\int_0^{T/2}\left(\frac{4A}{T}\right)t\sin n\omega_0 t\,dt + \frac{4}{T}\int_{T/4}^{T/2}\left(-\frac{4A}{T}t + 2A\right)\sin n\omega_0 t\,dt$$

$$= \frac{16A}{T^2}\int_0^{T/4} t\sin n\omega_0 t\,dt - \frac{16A}{T^2}\int_{T/4}^{T/2} t\sin n\omega_0 t\,dt + \frac{8A}{T}\int_{T/4}^{T/2}\sin n\omega_0 t\,dt$$

$$= \frac{16A}{T^2}\left[\left\{t\frac{\cos n\omega_0 t}{-n\omega_0}\right\}_0^{T/4} - \int_0^{T/4}\frac{\cos n\omega_0 t}{-n\omega_0}dt\right]$$

$$- \frac{16A}{T^2}\left[\left\{t\frac{\cos \omega_0 t}{-n\omega_0}\right\}_{T/4}^{T/2} - \int_{T/4}^{T/2}\frac{\cos n\omega_0 t}{-n\omega_0}dt\right]$$

$$+ \frac{8A}{T}\left\{\frac{\cos n\omega_0 t}{-n\omega_0}\right\}_{T/4}^{T/2}$$

$$= \frac{16A}{T^2}\left[\frac{T}{4}\frac{\cos n\omega_0 T/4}{-n\omega_0} + \left\{\frac{\sin n\omega_0 t}{n^2\omega_0^2}\right\}_0^{T/4}\right]$$

$$-\frac{16A}{T^2}\left[\frac{T}{2}\frac{\cos n\omega_0 T/2}{(-n\omega_0)} + \frac{T}{4}\frac{\cos n\omega_0 T/4}{n\omega_0} + \left\{\frac{\sin n\omega_0 t}{n^2\omega_0^2}\right\}_{T/4}^{T/2}\right]$$

$$+\frac{8A}{T}\left[\frac{\cos n\omega_0 T/2}{(-n\omega_0)} + \frac{\cos n\omega_0 T/4}{n\omega_0}\right]$$

Substituting $\omega_0 = \frac{2\pi}{T}$, we have

$$b_n = \frac{16A}{T^2}\frac{2\sin n\pi/2}{n^2 4\pi^2/T^2} + \frac{16A}{T^2}\cdot\frac{T}{2}\frac{\cos n}{n.2\pi/T} - \frac{8A}{T}\frac{\cos n\pi}{n2\pi/T}$$

Simplifying, we get

$$b_n = \frac{8A}{n^2\pi^2}\sin(n\pi/2)$$

$$b_1 = \frac{8A}{\pi^2}\sin(\pi/2) = \frac{8A}{\pi^2}$$

$$b_3 = \frac{8A}{3^2\pi^2}\sin\frac{3\pi}{2} = -\frac{8A}{3^2\pi^2}$$

$$b_5 = \frac{8A}{5^2\pi^2}\sin\frac{5\pi}{2} = \frac{8A}{5^2\pi^2}$$

$$\vdots$$

Substituting the values of the coefficients in Eq. (2.2), we get

$$f(t) = \frac{8A}{\pi^2}\left[\sin\omega_0 t - \frac{1}{3^2}\sin 3\omega_0 t + \frac{1}{5^2}\sin 5\omega_0 t + \cdots\right]$$

Example 2.5 *Deduce the Fourier series for the waveform of a positive going rectangular pulse train shown in Fig. E2.5.*

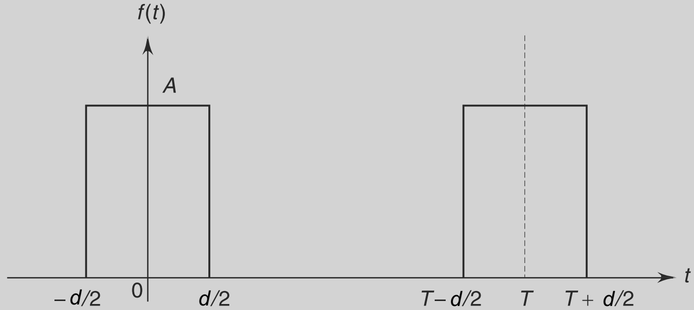


Fig. E2.5

Solution

The periodic function of the Fourier series for the given pulse train is expressed by

$$f(t)=\frac{1}{2}a_0+\sum_{n=1}^{\infty}a_n\cos n\omega_0 t+\sum_{n=1}^{\infty}b_n\sin n\omega_0 t$$

where

$$a_0=\frac{2}{T}\int_{-T/2}^{T/2}f(t)\,\mathrm{d}t$$

$$=\frac{2}{T}\int_{-d/2}^{d/2}A\,\mathrm{d}t=\frac{2A}{T}[t]_{-d/2}^{d/2}=\frac{2Ad}{T}$$

Here, since the choice of $t = 0$ is at the centre of a pulse, b_n coefficient are zero.

Therefore,

$$a_n=\frac{2}{T}\int_{-T/2}^{T/2}f(t)\cos n\omega_0 t\,dt=\frac{2}{T}\int_{-d/2}^{d/2}\cos n\omega_0\,t\,\mathrm{d}t$$

$$=\frac{2A}{T}\left[\frac{\sin n\omega_0 t}{n\omega_0}\right]_{-d/2}^{d/2}=\frac{2A}{n\omega_0 T}\left[\sin\left(\frac{n\omega_0 d}{2}\right)-\sin\left(\frac{-n\omega_0 d}{2}\right)\right]$$

$$=\frac{4A}{n\omega_0 T}\sin\frac{n\omega_0 d}{2}$$

Hence,

$$f(t)=\frac{Ad}{T}+\frac{2Ad}{T}\sum_{n=1}^{\infty}\frac{\sin(n\omega_0 d/2)}{n\omega_0\,d/2}\cos n\omega_0 t$$

COMPLEX OR EXPONENTIAL FORM OF FOURIER SERIES 2.3

From Eq. (2.2), the trigonometric form of Fourier series is

$$f(t)=\frac{1}{2}a_0+\sum_{n=1}^{\infty}(a_n\cos n\omega_0 t+b_n\sin n\omega_0 t)$$

An alternative but convenient way of writing the periodic function $f(t)$ is in exponential form with complex quantities. Since

$$\cos n\omega_0 t=\frac{e^{jn\omega_0 t}+e^{-jn\omega_0 t}}{2}$$

$$\sin n\omega_0 t=\frac{e^{jn\omega_0 t}-e^{-jn\omega_0 t}}{2j}$$

Substituting these quantities in the expression for the Fourier series gives

$$f(t)=\frac{1}{2}a_0+\sum_{n=1}^{\infty}a_n\left(\frac{e^{jn\omega_0 t}+e^{-jn\omega_0 t}}{2}\right)+\sum_{n=1}^{\infty}b_n\left(\frac{e^{jn\omega_0 t}-e^{-jn\omega_0 t}}{2j}\right)$$

$$=\frac{1}{2}a_0+\sum_{n=1}^{\infty}\left(\frac{(a_n-jb_n)e^{jn\omega_0 t}}{2}\right)+\left(\frac{(a_n+jb_n)e^{-jn\omega_0 t}}{-jb}\right)$$

Here, taking $$c_n = \frac{1}{2}(a_n - jb_n) \tag{2.6}$$

$$c_{-n} = \frac{1}{2}(a_n + jb_n)$$

$$c_0 = a_0$$

where, c_{2n} is the complex conjugate of c_n. Substituting expressions for the coefficients a_n and b_n from Eqs. (2.4) and (2.5) gives

$$c_n = \frac{1}{T}\int_{-T/2}^{T/2} f(t)[\cos n\omega_0 t - j\sin n\omega_0 t]\mathrm{d}t$$

$$= \frac{1}{T}\int_{-T/2}^{T/2} f(t)e^{-jn\omega_0 t}\,\mathrm{d}t \tag{2.7}$$

and $$c_{-n} = \frac{1}{T}\int_{-T/2}^{T/2} f(t)[\cos n\omega_0 t + j\sin n\omega_0 t]\mathrm{d}t \tag{2.8}$$

$$= \frac{1}{T}\int_{-T/2}^{T/2} f(t)e^{jn\omega_0 t}\,\mathrm{d}t$$

with $$f(t) = c_0 + \sum_{n=1}^{\infty} c_n e^{jn\omega_0 t} + \sum_{n=-\infty}^{-1} c_n e^{jn\omega_0 t} \tag{2.9}$$

where the values of n are negative in the last term and are included under the Σ sign. Also, c_0 may be included under the Σ sign by using the value of $n = 0$.

Therefore,

$$f(t) = \sum_{n=-8}^{\infty} c_n e^{jn\omega_0} \tag{2.10}$$

It is clear from the result given in Eq. (2.10) that the periodic function $f(t)$ may be expressed mathematically by an infinite set of positive and negative frequency components. The negative frequencies have not only mathematical significance, but also physical significance, since a positive frequency may be associated with an anti-clockwise rotation and a negative frequency with a clockwise rotation.

The complex Fourier series furnishes a method of decomposing a signal in terms of a sum of elementary signals of the form $\{e^{jn\omega_0 t}\}$. This representation may be used for signals $f(t)$ that are

(i) Periodic, $f(t) = f(t+T)$, in which case the representation is valid on $(-\infty, \infty)$
(ii) Aperiodic, in which case the representation is valid on a finite interval (t_1, t_2). The periodic extension of $f(t)$ is obtained outside of (t_1, t_2).

Note that similar to the evaluation of integrals a_n and b_n, the limits of integration in Eq. (2.7) may be the end points of any convenient full period and not essentially 0 to T or 0 to 2π. For $f(t)$ to be real, $C_{-n} = C_n^*$, so that only positive value on n are considered in Eq. (2.7). Also, we have

$$a_n = 2\,\mathrm{Re}\,[c_n] \quad \text{and} \quad b_n = -2\,\mathrm{Im}\,[c_n] \tag{2.11}$$

For an even waveform, the trigonometric Fourier series has only cosine terms and hence, by Eq. (2.6), the exponential Fourier series coefficients will be pure real numbers. Similarly, for an odd waveform, the trigonometric Fourier series contains only sine terms and hence the exponential Fourier series coefficients will be pure imaginary.

Example 2.6 *Find the exponential Fourier series coefficients for $x(t) = \sin \omega_0 t$.*

Solution

$$x(t) = \sin \omega_0 t$$
$$= \frac{e^{j\omega_0 t} - e^{-j\omega_0 t}}{2j} = \frac{1}{2j} e^{j\omega_0 t} - \frac{1}{2j} e^{-j\omega_0 t}$$

The Fourier series coefficients are

$$c_1 = \frac{1}{2j}, \; c_{-1} = -\frac{1}{2j} \text{ and } c_n = 0; \text{ for } n \neq 1 \text{ and } -1.$$

Example 2.7

(a) *Find the trigonometric Fourier series of the waveform shown in Fig. E2.7, and*

(b) *Determine the exponential Fourier series and hence find a_n and b_n of the trigonometric series and compare the results.*

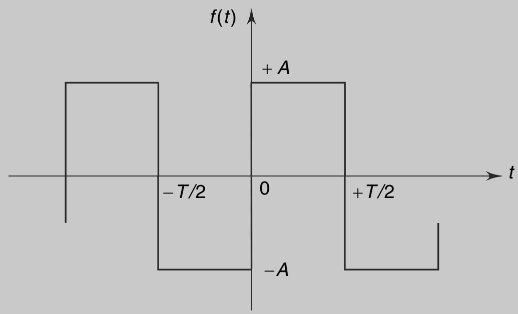


Fig.E2.7

Solution (a) The function of the given waveform for one period can be written as

$$f(t)\begin{cases} -A, & \text{for} -T/2 < t < 0 \\ +A, & 0 < t < T/2 \end{cases}$$

As the waveform is symmetrical about the origin, the function of the waveform is odd and hence $a_0 = a_n = 0$, and

$$b_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t) \sin n\omega_0 t \, dt$$
$$= \frac{2}{T}\left[\int_{-T/2}^{0} (-A \sin n\omega_0 t)\, dt + \int_{0}^{T/2} A \sin n\omega_0 t \, dt\right]$$
$$= \frac{2A}{T}\left\{\left[\frac{\cos n\omega_0 t}{n\omega_0}\right]_{-T/2}^{0} + \left[\frac{-\cos n\omega_0 t}{n\omega_0}\right]_{0}^{T/2}\right\}$$
$$= \frac{2A}{n\omega_0 T}\{[1-\cos(n\omega_0 T/2)] + [1-\cos(n\omega_0 T/2)]\} = \frac{4A}{n\omega_0 T}[1-\cos(n\omega_0 T/2)]$$

When $$\omega_0 = \frac{2\pi}{T}, \quad \text{we have}$$

$$b_n = \frac{4A}{n.2\pi}\left[1 - \cos n\left(\frac{2\pi}{T}\cdot\frac{T}{2}\right)\right] = \frac{2A}{n\pi}[1 - \cos n\pi]$$

$$b_n = \begin{cases} 0, & \text{if } n \text{ is even} \\ \dfrac{4A}{n\pi}, & \text{if } n \text{ is odd} \end{cases}$$

Substituting the values of the coefficients in Eq. (2.2), we obtain

$$f(t) = \frac{4A}{\pi}\left[\sin\omega_0 t + \frac{1}{3}\sin 3\omega_0 t + \frac{1}{5}\sin 5\omega_0 t + \cdots\right], \text{ where } \omega_0 = \frac{2\pi}{T}$$

(b) To determine exponential Fourier series

Here, $$c_0 = \left|\frac{1}{2}a_0\right| = 0$$

To evaluate c_n

Since the wave is odd, c_n consists of pure imaginary coefficients. From Eq. (2.7), we have

$$c_n = \frac{1}{T}\int_0^T f(t)e^{-jn\omega_0 t}\,dt$$

$$= \frac{1}{T}\left[\int_{-T/2}^{0}(-A)e^{-jn\omega_0 t}\,dt + \int_0^{T/2} Ae^{-jn\omega_0 t}\,dt\right]$$

$$= \frac{A}{T}\left\{\left[(-1)\frac{1}{(-jn\omega_0)}e^{-jn\omega_0 t}\right]_{-T/2}^{0} + \left[\frac{1}{(-jn\omega_0)}e^{-jn\omega_0 t}\right]_0^{T/2}\right\}$$

$$= \frac{A}{T}\cdot\frac{1}{(-jn\omega_0)}\{-e^0 + e^{jn\omega_0(T/2)} + e^{-jn\omega_0(T/2)} - e^0\}$$

When $$\omega_0 = \frac{2\pi}{T}, \quad \text{we get}$$

$$c_n = \frac{A}{T}\cdot\frac{T}{jn2\pi}\{-e^0 + e^{jn(2\pi/T)(T/2)} + e^{-jn(2\pi/2)(t/2)} - e^0\}$$

$$= \frac{A}{(-j2\pi n)}\{-e^0 + e^{jn\pi} + e^{-jn\pi} - e^0\} = j\frac{A}{n\pi}(e^{jn\pi} - 1)$$

Here, $e^{jn\pi} = +1$ for even n and $e^{jn\pi} = -1$ for odd n.

Therefore, $c_n = -j\left(\dfrac{2A}{n\pi}\right)$ for odd n only.

Hence, the exponential Fourier series is

$$f(t) = \ldots + j\frac{2A}{3\pi}e^{-j3\omega_0 t} + j\frac{2A}{\pi}e^{-j\omega_0 t} - j\frac{2A}{\pi}e^{j\omega_0 t} - j\frac{2A}{3\pi}e^{j3\omega_0 t}$$

By using Eq. (2.11), the trigonometric Fourier series coefficients, a_n and b_n can be evaluated as

$$a_n = 2\,\mathrm{Re}[c_n] = 2\,|\,c_n\,| = 0 \text{ and } b_n = -2\,\mathrm{Im}[c_n] = \frac{4A}{n\pi} \text{ for odd } n \text{ only.}$$

These coefficients are the same as the coefficients obtained in the trigonometric Fourier series.

Example 2.8

(a) *Find the trigonometric Fourier series of the waveform shown in Fig. E2.8 and*

(b) *Determine the exponential Fourier series and hence find a_n and b_n of the trigonometric series and compare the results.*

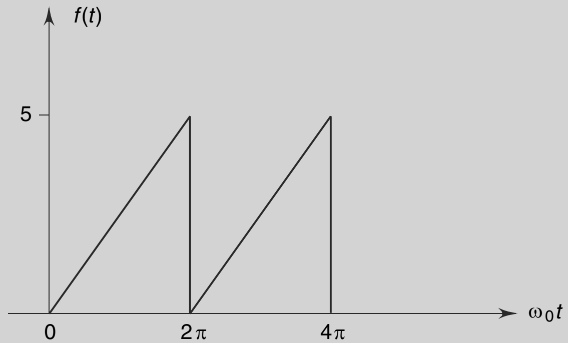


Fig. E2.8

Solution (a) As the waveform is periodic with period 2π in $\omega_0 t$ and continuous for $0 < \omega_0 t < 2\pi$, with discontinuities at $\omega_0 t = n(2\pi)$, where $n = 0, 1, 2, \ldots$, the Dirichlet conditions are satisfied.

To find f(t) for the given waveform of region $0 < \omega_0 t < 2\pi$

The equation of the straight line is $\dfrac{y - y_1}{x - x_1} = \dfrac{y_1 - y_2}{x_1 - x_2}$

Substituting $(x_1, y_1) = (0, 0)$ and $(x_2, y_2) = (2\pi, 5)$, we get

$$\frac{f(t) - 0}{\omega_0 t - 0} = \frac{0 - 5}{0 - 2\pi}$$

Therefore, $$f(t) = \left(\frac{5}{2\pi}\right)\omega_0 t$$

To find Fourier coefficients

Using Eq. (2.3), we obtain the average term,

$$a_0 = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\mathrm{d}t$$

$$= \frac{2}{2\pi}\int_0^{2\pi} \frac{5}{2\pi}\omega_0 t\,\mathrm{d}(\omega_0 t) = \frac{10}{(2\pi)^2}\left[\frac{(\omega_0 t)^2}{2}\right]_0^{2\pi} = \frac{10}{(2\pi)^2}\frac{(2\pi)^2}{2} = 5$$

Using Eq. (2.4), we obtain $a_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\cos n\omega_0 t\,\mathrm{d}t$

$$= \frac{2}{2\pi}\int_0^{2\pi}\left(\frac{5}{2\pi}\right)\omega_0 t\cos n\omega_0 t\,\mathrm{d}(\omega_0 t)$$

$$= \frac{5}{2\pi^2}\left[\frac{\omega_0 t}{n}\sin n\omega_0 t + \frac{1}{n^2}\cos n\omega_0 t\right]_0^{2\pi} = \frac{5}{2\pi^2 n^2}(\cos n2\pi - \cos 0) = 0$$

Hence, the series contains no cosine terms.

Using Eq. (2.4), we obtain $b_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\sin n\omega_0 t\,\mathrm{d}t$

$$= \frac{1}{\pi}\int_0^{2\pi}\left(\frac{5}{2\pi}\right)\omega_0 t\sin n\omega_0 t\,\mathrm{d}(\omega_0 t)$$

$$= \frac{5}{2\pi^2}\left[-\frac{\omega_0 t}{n}\cos n\omega_0 t + \frac{1}{n^2}\sin n\omega_0 t\right]_0^{2\pi} = -\frac{5}{n\pi}$$

Combining the average term and the sine-term coefficients, the series becomes

$$f(t) = \frac{1}{2}a_0 + b_1\sin\omega_0 t + b_2\sin 2\omega_0 t + \cdots$$

$$= \frac{5}{2} - \frac{5}{\pi}\sin\omega_0 t - \frac{5}{2\pi}\sin 2\omega_0 t - \frac{5}{3\pi}\sin 3\omega_0 t - \cdots = \frac{5}{2} - \frac{5}{\pi}\sum_{n=1}^{\infty}\frac{\sin n\omega_0 t}{n}$$

To determine exponential Fourier series

Here, $c_0 = \left|\frac{1}{2}a_0\right| = \frac{5}{2}$

To evaluate c_n

From Eq. (2.7), we have

$$c_n = \frac{1}{T}\int_0^T f(t)e^{-jn\omega_0 t}\mathrm{d}t$$

$$c_n = \frac{1}{2\pi}\int_0^{2\pi}\left(\frac{5}{2\pi}\right)\omega_0 t\, e^{-jn\omega_0 t}\mathrm{d}(\omega_0 t)$$

$$= \frac{5}{(2\pi)^2}\left[\frac{e^{-jn\omega_0 t}}{(-jn)^2}(-jn\omega_0 t - 1)\right]_0^{2\pi} = j\frac{5}{2\pi n}$$

Substituting the coefficients c_n in Eq. (2.9), the exponential Fourier series is

$$f(t) = \cdots - j\frac{5}{4\pi}e^{-j2\omega_0 t} - j\frac{5}{2\pi}e^{-j\omega_0 t} + \frac{5}{2} + j\frac{5}{2\pi}e^{j\omega_0 t} + j\frac{5}{4\pi}e^{-j2\omega_0 t} + \cdots$$

By using Eq. (2.11), the trigonometric Fourier series coefficients a_n and b_n can be evaluated as

$$a_n = 2\,\text{Re}[c_n] = 2\,|\,c_n\,| = 0 \text{ and } b_n = -2\,\text{Im}[c_n] = -\frac{5}{n\pi}$$

Hence, $$f(t) = \frac{5}{2} - \frac{5}{\pi}\sin\omega_0 t - \frac{5}{2\pi}\sin 2\omega_0 t - \frac{5}{3}\pi\sin 3\omega_0 t - \cdots$$

This result is the same as that of the trigonometric Fourier series method.

PARSEVAL'S IDENTITY FOR FOURIER SERIES 2.4

A periodic function $f(t)$ with a T is expressed by the Fourier series as

$$f(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty}(a_n \cos n\omega_0 t + b_n \sin n\omega_0 t)$$

Now $$[f(t)]^2 = \frac{1}{2}a_0 f(t) + \sum_{n=1}^{\infty}\left[a_n f(t)\cos n\omega_0 t + b_n f(t)\sin n\omega_0 t\right]$$

Therefore, $$\frac{1}{T}\int_{-T/2}^{T/2}[f(t)]^2\,dt = \frac{(a_0/2)}{T}\int_{-T/2}^{T/2}[f(t)]dt$$

$$+\frac{1}{T}\sum_{n=1}^{\infty}\left[a_n\int_{-T/2}^{T/2} f(t)\cos n\omega_0 t\,dt + b_n\int_{-T/2}^{T/2} f(t)\sin n\omega_0 t\,dt\right]$$

From Eqs. (2.2), (2.3) and (2.4), we have

$$a_0 = \frac{2}{T}\int_{-T/2}^{T/2} f(t)dt$$

$$a_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\cos n\omega_0 t\,dt$$

$$b_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\sin n\omega_0 t\,dt$$

Therefore, substituting all these values, we get

$$\frac{1}{T}\int_{-T/2}^{T/2}[f(t)]^2\,dt = \left(\frac{a_0}{2}\right)^2 + \frac{1}{2}\sum_{n=1}^{\infty}(a_n^2 + b_n^2) \tag{2.12}$$

This is the ***Parseval's identity.***

POWER SPECTRUM OF A PERIODIC FUNCTION 2.5

The power of a periodic signal spectrum $f(t)$ in the time domain is defined as

$$P = \frac{1}{T}\int_{-T/2}^{T/2}[f(t)]^2\,dt$$

The Fourier series of the signal $f(t)$ is

$$f(t)=\sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t}$$

According to Parseval's relation, we have

$$P_{\text{av}}=\frac{1}{T}\int_{-T/2}^{T/2}[f(t)]^2\,\mathrm{d}t$$

$$=\frac{1}{T}\int_{-T/2}^{T/2} f(t)\sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t}\,\mathrm{d}t$$

$$=\sum_{n=-\infty}^{\infty} c_n \frac{1}{T}\int_{-T/2}^{T/2} f(t)e^{jn\omega_0 t}\,\mathrm{d}t=\sum_{n=-\infty}^{\infty} c_n c_{-n}=\sum_{n=-\infty}^{\infty}|c_n|^2,\text{watt}$$

From Eq. (2.12) the above equation becomes

$$\left(\frac{a_0}{2}\right)^2+\frac{1}{2}\sum_{n=1}^{\infty}(a_n^2+b_n^2)=\sum_{n=0}^{\infty}|c_n|^2$$

Here, $$c_0=\left|\frac{a_0}{2}\right| \text{ and } c_n=\sqrt{a_n^2+b_n^2},\ (n\ge 1) \tag{2.13}$$

Thus the power in $f(t)$ is

$$P=\cdots+|c_{-n}|^2+\cdots+|c_{-1}|^2+|c_0|^2+|c_1|^2+\cdots+|c_n|^2+\cdots$$
$$P=|c_0|^2+2|c_1|^2+2|c_2|^2+\cdots+|c_n|^2+\cdots \tag{2.14}$$

Hence, for a periodic function, the power in a time waveform $f(t)$ can be evaluated by adding together the powers contained in each harmonic, i.e. frequency component of the signal $f(t)$.

The power for the n^{th} harmonic component at $n\ \omega_0$ radians per sec is $|c_n|^2$ and that of $-n\ \omega_0$ is $|c_{-n}|$. For the single real harmonic, we have to consider both the frequency components $\pm n\ \omega_0$.

Here, $c_n = c_{-n}^*$ and hence $|c_n|^2 = |c_{-n}|^2$. The power for the n^{th} real harmonic $f(t)$ is

$$P_n=|c_n|^2+|c_{-n}|^2=2|c_n|^2$$

The RMS value of $f(t)$

Using Eqs. (2.12), (2.13) and (2.14), the RMS value of the function $f(t)$ expressed by Eq. (2.1) is

$$F_{\text{rms}}=\sqrt{\left(\frac{a_0}{2}\right)^2+\frac{1}{2}a_1^2+\frac{1}{2}a_2^2+\cdots+\frac{1}{2}b_1^2+\frac{1}{2}b_2^2+\cdots}$$

$$=\sqrt{c_0^2+\frac{1}{2}c_1^2+\frac{1}{2}c_2^2+\cdots} \tag{2.15}$$

Example 2.9 *The complex exponential Fourier representation of a signal $f(t)$ over the interval $(0, T)$ is*

$$f(t)=\sum_{n=-\infty}^{\infty}\frac{3}{4+(n\pi)^2}e^{jn\pi t}$$

(a) *What is the numerical value of T?*
(b) *One of the components f (t) is A cos 3πt. Determine the value of A.*
(c) *Determine the minimum number of terms which must be retained in the representation of f (t) in order to include 99.9% of the energy in the interval.*

Note: $$\sum_{n=-\infty}^{\infty}\left|\frac{3}{4+(n\pi)^2}\right| \approx 0.669$$

Solution The complex exponential Fourier transform representation of a signal $f(t)$ is

$$f(t)=\sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t} \quad \text{where} \quad \omega_0=\frac{2\pi}{T}$$

The given signal $f(t)$ over the interval $(0, T)$ is

$$f(t)=\sum_{n=-\infty}^{\infty}\frac{3}{4+(n\pi)^2}e^{jn\pi t}$$

(a) Comparing the above two equations, we get

$$c_n=\frac{3}{4+(n\pi)^2}$$

and $$e^{jn\frac{2\pi}{T}t}=e^{jn\pi t}$$

Hence, $$\frac{2\pi}{T}=\pi, \quad \text{i.e,} \quad T=2$$

(b) When $n = 3$, the component of $f(t)$ will be

$$c_3=\frac{3}{4+(3\pi)^2}e^{j3\pi t}=\frac{3}{4+(3\pi)^2}[\cos 3\pi t+j\sin 3\pi t]$$

Similarly, when $n = -3$, the component will be

$$c_{-3}=\frac{3}{4+(-3\pi)^2}e^{-j3\pi t}=\frac{3}{4+(3\pi)^2}[\cos 3\pi t-j\sin 3\pi t]$$

Therefore, $$c_3+c_{-3}=\frac{6}{4+(3\pi)^2}\cos 3\pi t$$

Hence, when one of the components of $f(t)$ is $A \cos 3\pi t$, the value of A is

$$A=\frac{6}{4+(3\pi)^2}$$

(c) Total (maximum) power $$P_t=\sum_{n=-\infty}^{\infty}\left|\frac{3}{4+(n\pi)^2}\right| \approx 0.669$$

The power in $f(t)$ is

$$P=|c_0|^2+2\left[|c_1|^2+|c_2|^2+|c_3|^2+|c_4|^2\right]$$

$$=\left|\frac{3}{4}\right|^2+2\left[\left|\frac{3}{4+(\pi)^2}\right|^2+\left|\frac{3}{4+(2\pi)^2}\right|^2+\left|\frac{3}{4+(3\pi)^2}\right|^2+\left|\frac{3}{4+(4\pi)^2}\right|^2\right]$$

$$=0.5625+0.0935+9.52\times10^{-3}+2.088\times10^{-3}+6.866\times10^{-4}=0.66836$$

Therefore, energy contained in the four terms is

$$\frac{P}{P_t}\times 100=\frac{0.66836}{0.669}\times 100=99.9\%$$

Hence, the first four terms include 99.9% of the total energy.

FOURIER TRANSFORM 2.6

The plot of amplitudes at different frequency components for a periodic wave is known as discrete (line) frequency spectrum because amplitude values have significance only at discrete values of $n\,\omega_0$ where $\omega_0 = 2\pi/T$ is the separation between two adjacent (consecutive) harmonic components. If the repetition period T increases, ω_0 decreases.

Hence, when the repetition period T becomes infinity, i.e. $T\rightarrow\infty$, the wave $f(t)$ will become non-periodic, the separation between two adjacent harmonic components will be zero, i.e. $\omega_0 = 0$. Therefore, the discrete spectrum will become a continuous spectrum. When $T\rightarrow\infty$, the adjacent pulses virtually never occur and the pulse train reduces to a single isolated pulse. The exponential form of the Fourier series given in Eq. (2.10) can be extended to aperiodic waveforms such as single pulses or single transients by making a few changes.

Assuming $f(t)$ is initially periodic, from Eq. (2.10), we have

$$f(t)=\sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t}$$

where $$c_n=\frac{1}{T}\int_{-T/2}^{T/2} f(t)\,e^{-jn\omega_0 t}\,\mathrm{d}t$$

In the limit, for a single pulse, we have

$$T\rightarrow\infty,\ \omega_0 = 2\pi/T\rightarrow \mathrm{d}\omega \text{ (a small quantity)}$$

or $$\frac{1}{T}=\omega_0/2\pi\rightarrow \mathrm{d}\omega/2\pi$$

Furthermore, the n^{th} harmonic in the Fourier series is $n\,\omega_0\rightarrow nd\,\omega$. Here n must tend to infinity as ω_0 approaches zero, so that the product is finite, i.e. $n\,\omega_0\rightarrow\omega$.

In the limit, the Σ sign leads to an integral and we have

$$c_n=\frac{d\omega}{2\pi}\int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t$$

and, $$f(t)=\int_{-\infty}^{\infty}\frac{d\omega}{2\pi}\left[\int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t\right]e^{j\omega t}$$

When evaluated, the quantity in bracket is a function of frequency only and is denoted as $F(j\omega)$ where

$$F(j\omega)=\int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t \tag{2.16}$$

It is called the *Fourier transform* of $f(t)$.

Substituting for $f(t)$ above, we obtain

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)e^{j\omega t}\mathrm{d}\omega$$

or equivalently,

$$f(t) = \int_{-\infty}^{\infty} F(j\omega)e^{j\omega t}\mathrm{d}f \tag{2.17}$$

which is called the *inverse Fourier transform*. Now the time function $f(t)$ represents the expression for a single pulse or transient only. Equations (2.16) and (2.17) constitute a *Fourier transform pair*.

From Eqs. (2.16) and (2.17), it is apparent that the Fourier transform and inverse Fourier transform are similar, except for sign change on the exponential component.

2.6.1 Energy Spectrum for a Non-Periodic Function

For a non-periodic energy signal, such as a single pulse, the total energy in $(-\infty, \infty)$ is finite, whereas the average power, i.e. energy per unit time is zero because $\frac{1}{T}$ tends to zero as T tends to infinity. Hence, the total energy associated with $f(t)$ is given by

$$E = \int_{-\infty}^{\infty} f^2(t)\mathrm{d}t$$

Since,

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)e^{j\omega t}d\omega, \text{we obtain}$$

$$E = \int_{-\infty}^{\infty} f(t)\frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)e^{j\omega t}\mathrm{d}\omega\ \mathrm{d}t$$

$$= \frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)\left[\int_{-\infty}^{\infty} f(t)e^{j\omega t}\mathrm{d}t\right]\mathrm{d}\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)F(-j\omega)\mathrm{d}\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)F^*(j\omega)\mathrm{d}\omega$$

$$= \frac{1}{2\pi}\int_{-\infty}^{\infty} |F(j\omega)|^2\,\mathrm{d}\omega = \int_{-\infty}^{\infty} |F(f)|^2\,\mathrm{d}f, \quad \text{joules}$$

$$E = \int_{-\infty}^{\infty} |f(t)|^2\,\mathrm{d}t = \int_{-\infty}^{\infty} |F(f)|^2\,\mathrm{d}f$$

This result is called ***Rayleigh's energy theorem*** or ***Parseval's theorem*** for Fourier transform. The quantity $|F(f)|^2$ is referred to as the *energy spectral density*, $S(f)$, which is equal to the energy per unit frequency.

The integration in Eq. (2.18) is carried out over positive and negative frequencies. If $f(t)$ is real, then $|F(j\omega)| = |F(-j\omega)|$, then the Eq. (2.18) becomes,

$$E = \frac{1}{2\pi}\int_{-\infty}^{\infty} |F(j\omega)|^2\,\mathrm{d}\omega = \frac{1}{\pi}\int_{0}^{\infty}|F(j\omega)|^2\,\mathrm{d}\omega = \int_{0}^{\infty} S(\omega)\,\mathrm{d}\omega$$

Here the integration is carried out over only positive frequencies. The quantity $S(\omega) = |F(j\omega)|^2/\pi$ is called the energy spectral density.

PROPERTIES OF FOURIER TRANSFORM 2.7

Table 2.1 presents important properties of the Fourier transform.

Table 2.1 *Important properties of the Fourier transform*

Operation	$f(t)$	$F(j\omega)$
Transform	$f(t)$	$F(j\omega)\, e^{-j\omega t_0}$
Inverse transform	$\frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)e^{j\omega t}\,\mathrm{d}\omega$	$F(j\omega)$
Linearity	$af_1(t) + bf_2(t)$	$aF_1(j\omega) + bF_2(j\omega)$
Time-reversal	$f(-t)$	$F(-j\omega) = F^*(j\omega), f(t)$ real
Time-shifting (Delay)	$f(t-t_0)$	$F(j\omega)\, e^{-j\omega t_0}$
Time-Scaling	$f(at)$	$\frac{1}{\lvert a\rvert}F\left(\frac{j\omega}{a}\right)$
Time-differentiation	$\frac{\mathrm{d}^n}{\mathrm{d}t^n}f(t)$	$(j\omega)^n F(j\omega)$
Frequency differentiation	$(-jt)\,f(t)$	$\frac{\mathrm{d}F(j\omega)}{\mathrm{d}\omega}$
Time-integration	$\int_{-\infty}^{t} f(\tau)\,\mathrm{d}\tau$	$\frac{1}{j\omega}F(j\omega)+\pi F(0)\delta(\omega)$
Frequency-integration	$\frac{1}{(-jt)}f(t)$	$\int^{\omega} f(j\omega')\,\mathrm{d}\omega'$
Time convolution	$f_1(t) * f_2(t) = \int_{-\infty}^{\infty} f_1(\tau)f_2(t-\tau)\,\mathrm{d}\tau$	$F_1(j\omega)\, F_2(j\omega)$
Frequency convolution (Multiplication)	$f_1(t)\cdot f_2(t)$	$\frac{1}{2\pi}[F_1(j\omega) * F_2(j\omega)]$
Frequency shifting (Modulation)	$f(t)e^{j\omega_0 t}$	$F(j\omega - j\omega_0)$
Symmetry	$F(jt)$	$2\pi f(-\omega)$
Real-time function	$f(t)$	$F(j\omega) = F^*(-j\omega)$ $\mathrm{Re}\,[F(j\omega)] = \mathrm{Re}\,[F(-j\omega)]$ $\mathrm{Im}\,[F(j\omega)] = -\mathrm{Im}\,[F(-j\omega)]$ $\lvert F(j\omega)\rvert = \lvert F(-j\omega)\rvert$ $\Phi f(j\omega) = -\Phi f(j\omega)$
Parseval's theorem	$E = \int_{-\infty}^{\infty} \lvert f(t)\rvert^2\,\mathrm{d}t$	$E = \frac{1}{2\pi}\int_{-\infty}^{\infty} \lvert F(j\omega)\rvert^2\,\mathrm{d}\omega$
Duality	If $f(t) \Rightarrow g(j\omega)$, then $g(t) \Rightarrow 2\pi f(-j\omega)$	

2.7.1 Linearity

The Fourier transform is a linear operation. Therefore, if

$$f_1(t) \Rightarrow F_1(j\omega)$$
$$f_2(t) \Rightarrow F_2(j\omega)$$

then, $af_1(t) + bf_2(t) \Rightarrow aF_1(j\omega) + bF_2(j\omega)$

where a and b are arbitrary constants.

2.7.2 Symmetry

If $f(t) \Rightarrow F(j\omega)$

then, $F(jt) \Rightarrow 2\pi f(-\omega)$

Proof

Since
$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)e^{j\omega t}\,d\omega$$

$$2\pi f(-t) = \int_{-\infty}^{\infty} F(j\omega')e^{-j\omega' t}\,d\omega'$$

where the dummy variable ω is replaced by ω'.
Now if t is replaced by ω, we have

$$2\pi f(-\omega) = \int_{-\infty}^{\infty} F(j\omega')e^{-j\omega'\omega}\,d\omega'$$

Finally, ω' is replaced by t to obtain a more recognisable form and we have

$$2\pi f(-\omega) = \int_{-\infty}^{\infty} F(jt)e^{-j\omega t}\,dt = \mathcal{F}[F(jt)]$$

Therefore, $F(jt) \Rightarrow 2\pi f(-\omega)$

If $f(t)$ is an even function, $f(t) = f(-t)$.

Hence, $\mathcal{F}[F(jt)] = 2\pi f(\omega)$

2.7.3 Scaling

If
$$f(t) \Leftrightarrow F(j\omega)$$

then,
$$f(at) \Leftrightarrow \frac{1}{|a|}F\left(\frac{j\omega}{a}\right)$$

Proof

If $a < 0$, then the Fourier transform of $f(at)$ is

$$\mathcal{F}[f(at)] = \int_{-\infty}^{\infty} f(at)e^{-j\omega t}\,dt$$

Putting $x = at$, we have $dx = a\,dt$. Substituting in the above equation, we get

$$\mathcal{F}[f(at)] = \mathcal{F}[f(x)] = \int_{-\infty}^{\infty} f(x)e^{\frac{-j\omega x}{a}}\frac{dx}{a} = \frac{1}{a}F\left(\frac{j\omega}{a}\right)$$

If $a > 0$, then $\mathcal{F}[f(at)] = -\frac{1}{a}F\left(\frac{j\omega}{a}\right)$

Combining these two results, we get

$$f(at) \Leftrightarrow \frac{1}{|a|}F\left(\frac{j\omega}{a}\right)$$

We conclude that larger the duration of the time function, smaller is the bandwidth of its spectrum by the same scaling factor. Conversely, smaller the duration of the time function, larger is the bandwidth of its spectrum. This scaling property provides an *inverse relationship* between time-duration and bandwidth of a signal, i.e. the time-bandwidth product of an energy signal is a constant.

2.7.4 Convolution

Convolution is a powerful way of characterising the input-output relationship of time-invariant linear systems. There are two convolution theorems, one for the time domain and another for the frequency domain.

Time Convolution

If $x(t) \Leftrightarrow X(j\omega)$ and $h(t) \Leftrightarrow H(j\omega)$,

then $y(t) = x(t) * h(t)$

$$\int_{-\infty}^{\infty} x(\tau)h(t-\tau)\,\mathrm{d}\tau \Leftrightarrow Y(j\omega) = X(j\omega)H(j\omega)$$

Proof

$$\mathcal{F}[y(t)] = Y(j\omega) = \int_{-\infty}^{\infty} e^{-j\omega t}\left[\int_{-\infty}^{\infty} x(\tau)h(t-\tau)\,\mathrm{d}\tau\right]\mathrm{d}t$$

$$= \int_{-\infty}^{\infty} x(\tau)\left[\int_{-\infty}^{\infty} h(t-\tau)e^{-j\omega t}\,\mathrm{d}t\right]\mathrm{d}\tau$$

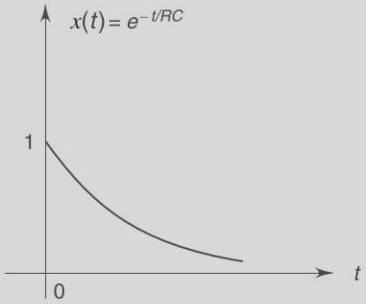
Putting $a = t - \tau$, then $t = a + \tau$ and $\mathrm{d}a = \mathrm{d}t$

Therefore, $$Y(j\omega) = \int_{-\infty}^{\infty} x(\tau)\left[\int_{-\infty}^{\infty} h(a)e^{-j\omega(a+\tau)}\,\mathrm{d}a\right]d\tau$$

$$= \int_{-\infty}^{\infty} x(\tau)e^{-j\omega\tau}\,\mathrm{d}\tau \int_{-\infty}^{\infty} h(a)e^{-j\omega a}\,\mathrm{d}a = X(j\omega)H(j\omega)$$

Hence, the convolutions of the signals in the time domain is equal to the multiplication of their individual Fourier transforms in the frequency domain.

Example 2.10 *In the system shown in Fig. E2.10 determine the output response of the low-pass RC network for an input signal* $x(t) = e^{-\frac{t}{RC}}$.



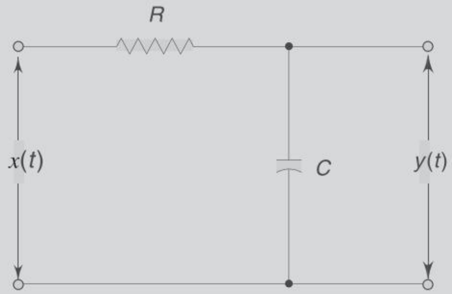


Fig. E2.10

Solution The input signal $x(t) = e^{-\frac{t}{RC}}$

Using the convolution theorem, we can find $y(t)$

$$y(t) = x(t) * h(t) = \mathcal{F}^{-1}[X(j\omega)H(j\omega)]$$

$$X(j\omega) = \mathcal{F}[x(t)] = \int_0^{\infty} e^{-\frac{t}{RC}} e^{-j\omega t}\, \mathrm{d}t$$

$$= \int_0^{\infty} e^{-\left(\frac{t}{RC} + j\omega\right)t}\, \mathrm{d}t = \frac{1}{j\omega + \frac{1}{RC}}$$

Similarly, the transfer function of the network is

$$H(j\omega) = \frac{1/j\omega C}{\left(R + \frac{1}{j\omega C}\right)} = \frac{1}{(j\omega RC + 1)} = \frac{1}{RC} \cdot \frac{1}{\left(j\omega + \frac{1}{RC}\right)}$$

Hence, $$Y(j\omega) = X(j\omega)H(j\omega) = \frac{1}{RC} \cdot \frac{1}{\left(j\omega + \frac{1}{RC}\right)^2}$$

$$Y(t) = \mathcal{F}^{-1}[Y(j\omega)] = \frac{1}{RC} t e^{-\frac{1}{RC}} u(t)$$

Example 2.11 *Determine the output response of the low-pass RC network due to an input* $x(t) = te^{-t/RC}$ *by convolution.*

Solution The transfer function of the network is

$$H(j\omega) = \frac{\frac{1}{j\omega C}}{R + \frac{1}{j\omega C}} = \frac{1}{(1 + j\omega RC)}$$

The given input time function is $x(t) = te^{-t/RC}$

Therefore, $$X(j\omega) = \mathcal{F}[te^{-t/RC}] = \frac{1}{\left(\frac{1}{RC} + j\omega\right)^2} = \frac{(RC)^2}{(1 + j\omega RC)^2}$$

We know that $Y(j\omega) = X(j\omega)H(j\omega)$

Hence, $$Y(j\omega) = \frac{(RC)^2}{(1 + j\omega RC)^2} \cdot \frac{1}{(1 + j\omega RC)} = \frac{(RC)^2}{(1 + j\omega RC)^3}$$

Therefore, $$Y(t) = \mathcal{F}^{-1}\left[\frac{(RC)^2}{(1 + j\omega RC)^3}\right] = \mathcal{F}^{-1}\left[\frac{1}{RC} \cdot \frac{1}{\left(\frac{1}{RC} + j\omega\right)^3}\right]$$

$$= \frac{1}{RC} \frac{t^2 e^{-t/Rc}}{2} u(t)$$

Hence, $$y(t) = \frac{1}{RC} \frac{t^2 e^{-t/Rc}}{2} u(t)$$

Example 2.12 *The input signal $x(t) = e^{-at}u(t)$, $a > 0$ is applied to the system whose transfer function is $h(t) = e^{-bt}u(t)$, $b > 0$. Determine the output signal $y(t)$ when $b \neq a$ and $b = a$.*

Solution The Fourier transform of $x(t)$ and $h(t)$ are

$$X(j\omega) = \frac{1}{(a + j\omega)} \quad \text{and} \quad H(j\omega) = \frac{1}{(b + j\omega)}$$

Therefore, $Y(j\omega) = X(j\omega)\,H(j\omega) = \dfrac{1}{(a + j\omega)(b + j\omega)}$

Expanding the function $Y(j\omega)$ in partial fractions, we get

$$Y(j\omega) = \frac{A}{(a + j\omega)} + \frac{B}{(b + j\omega)}$$

$$A = Y(j\omega)(a + j\omega)\Big|_{j\omega = -a} = \frac{1}{(b - a)}$$

$$B = Y(j\omega)(b + j\omega)\Big|_{j\omega = -b} = \frac{1}{(a - b)}$$

Hence, $$Y(j\omega) = \frac{1}{(b - a)}\left[\frac{1}{(a + j\omega)} - \frac{1}{(b + j\omega)}\right]$$

Taking inverse Fourier transform, we get

$$y(t) = \frac{1}{(b-a)}[e^{-at}u(t) - e^{-bt}u(t)]$$

When $b = a$, the partial fraction expansion is invalid. Hence,

$$Y(j\omega) = \frac{1}{(a+j\omega)^2} = j\frac{d}{d\omega}\left[\frac{1}{a+j\omega}\right]$$

Using dual of the differentiation property,

$$e^{-at}u(t) \Leftrightarrow \frac{1}{a+j\omega}$$

$$te^{-at}u(t) \Leftrightarrow j\frac{d}{d\omega}\left(\frac{1}{a+j\omega}\right) = \frac{1}{(a+j\omega)^2}$$

Therefore, $y(t) = t\,e^{-at}\,u(t)$

2.7.5 Frequency Convolution

If $f(t) \Rightarrow F(j\omega)$ and $g(t) \Rightarrow G(j\omega)$,

then $$f(t)g(t) \Leftrightarrow \frac{1}{2\pi}F(j\omega)*G(j\omega)$$

Proof The inverse transform of $[F(j\omega)*G(j\omega)]/2\pi$ is

$$\mathcal{F}^{-1}\left[\frac{F(j\omega)*G(j\omega)}{2\pi}\right] = \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} e^{j\omega t}\int_{-\infty}^{\infty} F(ju)G(j\omega - ju)\,du\,d\omega$$

$$= \frac{1}{(2\pi)^2}\int_{-\infty}^{\infty} F(ju)\int_{-\infty}^{\infty} G(j\omega - ju)e^{j\omega t}\,d\omega\,du$$

Putting $x = \omega - u$, then $\omega = x + u$ and $dx = d\omega$

Therefore,

$$\mathcal{F}^{-1}\left[\frac{F(j\omega)*G(j\omega)}{2\pi}\right] = \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} F(ju)\int_{-\infty}^{\infty} G(jx)e^{j(x+u)t}\,dx\,du$$

$$= \frac{1}{2\pi}\int_{-\infty}^{\infty} F(ju)e^{jut}\,du\,\frac{1}{(2\pi)}\int_{-\infty}^{\infty} G(jx)e^{jxt}\,dx = f(t)\cdot g(t)$$

Therefore, $$f(t)\cdot g(t) \Leftrightarrow \frac{1}{2\pi}F(j\omega)*G(j\omega)$$

Hence, the multiplication of two signals in the time domain is equal to the convolution of their individual Fourier transforms in the frequency domain. This property is called the ***multiplication theorem.***

2.7.6 Time Shifting (Delay)

If $f(t) \Rightarrow F(j\omega)$,

then $$f\left(t-t_0\right) \Leftrightarrow F(j\omega)\, \mathrm{e}^{-j\omega t_0}$$

Proof The Fourier transform of $f(t - t_0)$ is given by

$$\mathcal{F}[f(t-t_0)] = \int_{-\infty}^{\infty} f(t-t_0)\, e^{-j\omega t}\, \mathrm{d}t$$

Putting $x = t - t_0$, then $t = t_0 + x$ and $\mathrm{d}x = \mathrm{d}t$,

$$\mathcal{F}[f(t-t_0)] = \int_{-\infty}^{\infty} f(x)\, e^{-j\omega(t_0+x)}\, \mathrm{d}x = e^{-j\omega t_0} F(j\omega)$$

Therefore, $$f\left(t-t_0\right) \Leftrightarrow F(j\omega)\, \mathrm{e}^{-j\omega t_0}$$

When the function $f(t)$ is delayed by t_0 the original spectrum is multiplied by $e^{-j\omega t_0}$. Here there is no change in the amplitude spectrum and each frequency component is shifted in phase by an amount $-\omega t_0$.

Example 2.13 *Find the amplitude and phase spectrum of the time shifted impulse signal $f(t) = 10\,\delta\,(t - 2)$.*

Solution $f(t) = 10\,\delta\,(t - 2)$

Using the shifted theorem, $f(t-t_0) \Leftrightarrow F(j\omega)e^{-j\omega t_0}$, we get

$$F(j\omega) = \mathcal{F}\,[10\,\delta\,(t-2)] = 10\, e^{-j2\omega}$$

Table E2.13 *Phase spectrum*

ω	$\underline{\vert F(j\omega)}$
0.25	−0.5
0.50	−1.0
1.0	−2.0
−0.25	+0.5
−0.50	+1.0
−1.0	+2.0

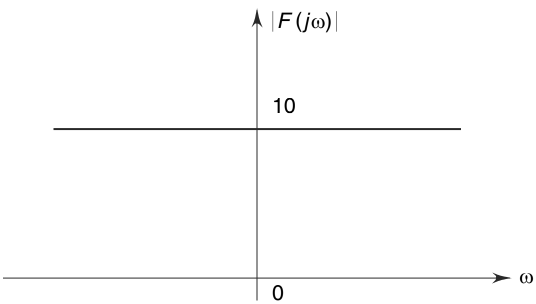


Fig. E2.13(a) *Amplitude Spectrum*

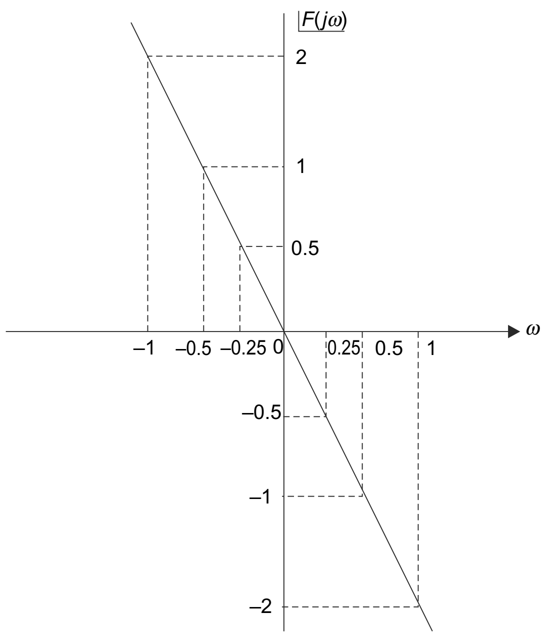


Fig. E2.13(b) *Phase Spectrum*

Hence, the magnitude $|F(j\omega)| = 10$ and the phase $\underline{|F(j\omega)}= -2\omega$

The amplitude spectrum and phase spectrum are shown in Fig. E2.13(a) and (b) respectively.

2.7.7 Frequency Shifting (Modulation)

If $f(t) \Leftrightarrow F(j\omega)$,

then $\quad f(t)e^{j\omega_0 t} \Leftrightarrow F(j\omega - j\omega_0)$

Proof The transform of $f(t)e^{j\omega_0 t}$ is by definition

$$\mathcal{F}[f(t)e^{j\omega_0 t}] = \int_{-\infty}^{\infty} f(t)e^{j\omega_0 t}e^{-j\omega t}\,\mathrm{d}t = \int_{-\infty}^{\infty} f(t)e^{-j(\omega-\omega_0)t}\,\mathrm{d}t = F(j\omega - j\omega_0)$$

Hence, $\quad f(t)e^{j\omega_0 t} \Leftrightarrow F(j\omega - j\omega_0)$

This theorem is very important for understanding the modulation in telecommunication systems.

Example 2.14 *Determine the spectrum of $f(t)\cos\omega_0 t$ by using the frequency shifting theorem.*

Solution The spectrum of $f(t)\cos\omega_0 t$ is

$$\mathcal{F}[f(t)\cos\omega_0 t] = \mathcal{F}\left[f(t)\left\{\frac{e^{j\omega_0 t}+e^{-j\omega_0 t}}{2}\right\}\right]$$

$$= \frac{1}{2}\mathcal{F}\left[f(t)e^{j\omega_0 t}\right] + \frac{1}{2}\mathcal{F}\left[f(t)e^{-j\omega_0 t}\right]$$

$$= \frac{1}{2}\{F(j\omega - j\omega_0) + F(j\omega - j\omega_0)\}, \quad \text{(using the shifting theorem)}$$

Thus, due to the product of $f(t)$ and $\cos\omega_0 t$, the original spectrum is shifted in such a way that half the original spectrum is centred about ω_0 and the other half is centred about $-\omega_0$.

2.7.8 Time Differentiation

If $\quad f(t) \Leftrightarrow f(j\omega)$, then $\dfrac{\mathrm{d}f(t)}{\mathrm{d}t} \Leftrightarrow j\omega F(j\omega)$

and $\quad \dfrac{\mathrm{d}^n}{\mathrm{d}t^n} f(t) \Leftrightarrow (j\omega)^n F(j\omega)$

2.7.9 Frequency Differentiation

If $\quad f(t) \Leftrightarrow F(j\omega)$, then

$$(-jt)f(t) \Leftrightarrow \frac{\mathrm{d}F(j\omega)}{\mathrm{d}\omega}$$

2.7.10 Time Integration

If $f(t) \Leftrightarrow F(j\omega)$, then

$$\int_{-\infty}^{t} f(\tau)\,\mathrm{d}\,\tau \Leftrightarrow \frac{F(j\omega)}{j\omega} + \pi F(0)\delta(\omega)$$

2.7.11 Frequency Integration

If $f(t) \Leftrightarrow F(j\omega)$, then

$$\frac{f(t)}{-jt} \Leftrightarrow \int^{\omega} F(j\omega')\,\mathrm{d}\,\omega'$$

2.7.12 Time Reversal

If $f(t) \Leftrightarrow F(j\omega)$, then $f(-t) \Leftrightarrow F(-j\omega)$

It is clear that the time reversal theorem is similar to the scaling theorem with $a = -1$. When the signal is real, time reversal affects only the phase spectrum because the amplitude spectrum is an even function of frequency.

2.7.13 Complex Conjugation

If $f(t) \Leftrightarrow F(j\omega)$, then $f^*(t) \Leftrightarrow F^*(-j\omega)$

2.7.14 Duality

If $f(t) \Leftrightarrow g(j\omega)$, then $g(t) \Leftrightarrow 2\pi f(-j\omega)$

2.7.15 Area under *f*(*t*)

If $f(t) \Leftrightarrow F(f)$, then $\int_{-\infty}^{\infty} F(f)\,df = f(0)$

Thus, the area under a function $f(t)$ is equal to the value of its Fourier transform $F(f)$ at $f = 0$.

The result can be obtained by substituting $f = 0$ in the formula defining the Fourier transform of the function $f(t)$.

2.7.16 Area under *F*(*f*)

If $f(t) \Leftrightarrow F(f)$, then $\int_{-\infty}^{\infty} F(f)\,df = f(0)$

Thus the value of a function $f(t)$ at $t = 0$ is equal to the area under its Fourier transform $F(f)$. The result can be obtained by substituting $t = 0$ in the formula defining the inverse Fourier transform of $F(f)$.

2.7.17 Parseval's Theorem

The Fourier transform is an energy-conserving relation and the energy may be found from $f(t)$ or its spectrum $|F(j\omega)|$ as,

$$E = \int_{-\infty}^{\infty} |f(t)|^2 \, dt = \frac{1}{2\pi}\int_{-\infty}^{\infty} |F(j\omega)|^2 \, d\omega$$

This relation is called Parseval's theorem or Rayleigh's Energy theorem.

Proof

$$E = \int_{-\infty}^{\infty} |f(t)|^2 \, dt = \int_{-\infty}^{\infty} f(t) f^*(t) dt$$

$$\int_{-\infty}^{\infty} f(t)\left[\frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)e^{j\omega t} d\omega\right]^* dt$$

$$= \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(t)F^*(j\omega)e^{-j\omega t} d\omega\, dt$$

$$= \frac{1}{2\pi}\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt\right]F^*(j\omega)d\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(j\omega)F^*(j\omega)d\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty} |F(j\omega)|^2 \, d\omega$$

Example 2.15 *A certain function of time f(t) has the following Fourier transform*

$$F(j\omega) = \frac{1}{\omega^2+1} \; e^{-2\omega^2/(\omega^2+1)}$$

Using the properties of the Fourier transform, write the Fourier transforms of

(a) $f(2t)$, (b) $f(t-2)e^{jt}$, (c) $4\frac{d}{dt}f(t)$ and (d) $\int_{-\infty}^{t} f(\tau)d\tau$

In each case, state clearly the properties you will use.

Solution $F(j\omega) = \frac{1}{\omega^2+1} e^{-2\omega^2/(\omega^2+1)}$

(a) To find f (2t) Using the scaling property $f(at) = \frac{1}{|a|}F\left(\frac{j\omega}{a}\right)$, we obtain

$$f(2t) = \frac{1}{|2|}F\left(\frac{j\omega}{2}\right)$$

$$= \frac{1}{2}\cdot\frac{1}{\left[\left(\frac{\omega}{2}\right)^2+1\right]}\cdot e^{-2(\omega/2)^2/[(\omega/2)^2+1]} = \frac{2}{(\omega^2+4)}e^{-2\omega^2/(\omega^2+4)}$$

(b) To find f (t – 2) e^jt Using the time shifting property $f(t-t_0) \Leftrightarrow F(j\omega)e^{-j\omega t_0}$, we get

$$f(t-2) = F(j\omega)e^{-j2\omega}$$

Therefore, $f(t-2) = \dfrac{1}{(\omega^2+1)} e^{-2\omega^2/(\omega^2+1)} e^{-j2\omega}$

Using the frequency shifting property $f(t)e^{j\omega_0 t} \Leftrightarrow F(j\omega - j\omega_0)$, we get

$$\begin{aligned} f(t-2)e^{jt} &= F[j(\omega-1)]e^{-j2(\omega-1)} \\ &= \frac{1}{[(\omega-1)^2+1]} \cdot e^{-2(\omega-1)^2/[(\omega-1)^2+1]} e^{-j2(\omega-1)} \end{aligned}$$

(c) To find $4\dfrac{\mathrm{d}}{\mathrm{d}t}f(t)$ Using the differentiation property, $\dfrac{\mathrm{d}f(t)}{\mathrm{d}t} \Leftrightarrow j\omega F(j\omega)$, we get

$$4\frac{\mathrm{d}}{\mathrm{d}t}f(t) = j4\omega F(j\omega) = j4\omega \frac{1}{\omega^2+1} e^{-2\omega^2/(\omega^2+1)}$$

(d) To find $\int_{-\infty}^{t} f(\tau)\,d\tau$ Using the integration property

$$\int_{-\infty}^{t} f(\tau)\,d\tau \Leftrightarrow \frac{F(j\omega)}{j\omega} + \pi F(0)\delta(\omega), \quad \text{we get}$$

$$\Leftrightarrow \frac{1}{j\omega}\frac{1}{(\omega^2+1)} e^{-2\omega^2/(\omega^2+1)} + \pi\delta(\omega), \quad [\text{since } F(0) = 1]$$

— FOURIER TRANSFORM OF SOME IMPORTANT SIGNALS 2.8

Table 2.2 presents the Fourier transform of some important signals.

2.8.1 Gate Function

Let us consider the single gate function (rectangular pulse) shown in Fig. 2.4. It has the analytic expression given by

$$f(t) = \begin{cases} 1, & \text{for } -T/2 < t < T/2 \\ 0, & \text{otherwise} \end{cases}$$

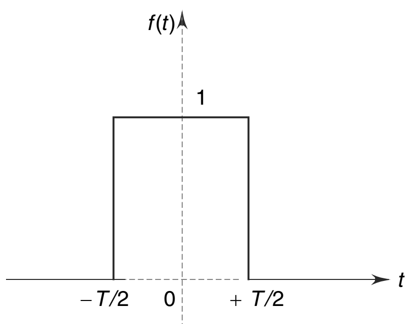
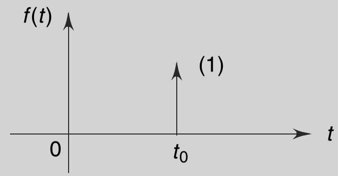
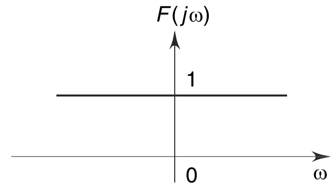
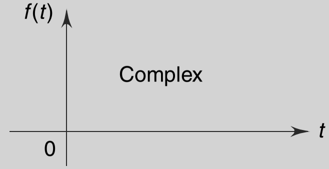
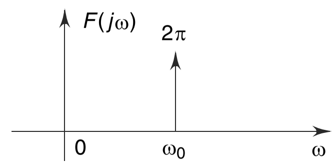
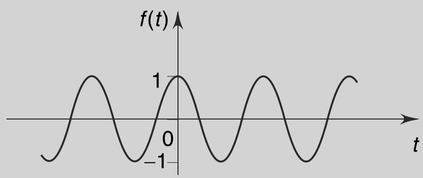
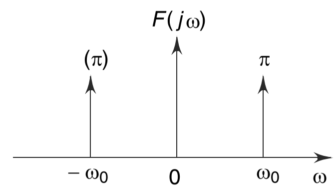
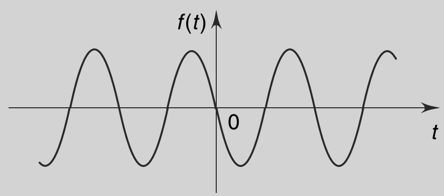
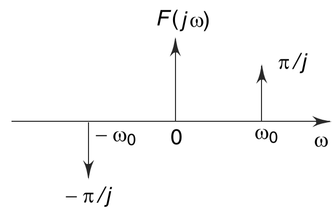
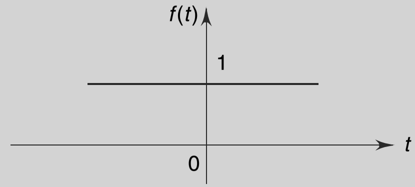
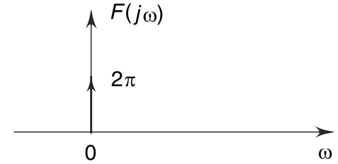
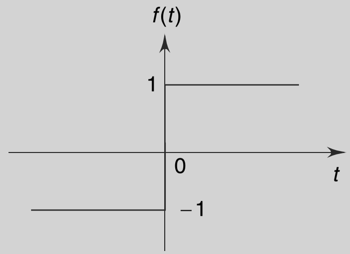
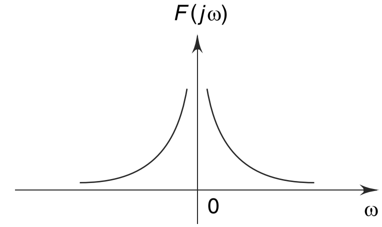
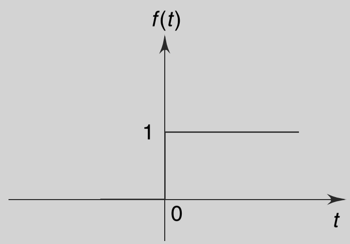
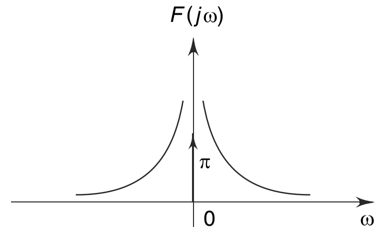


Fig. 2.4 *Single Gate Function*

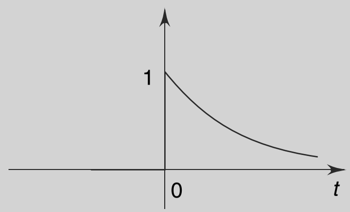
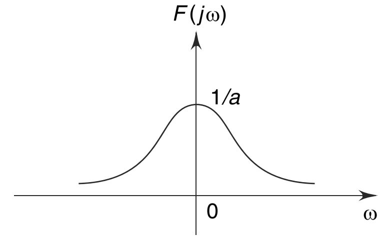
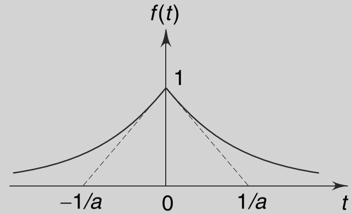
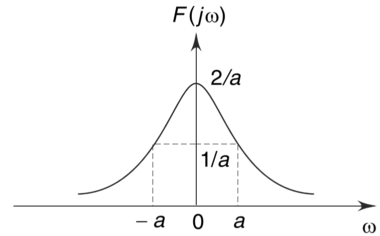
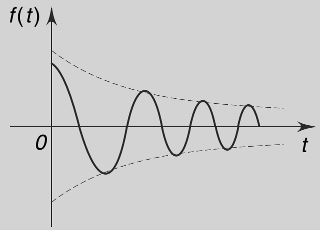
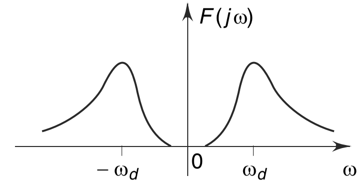
Table 2.2 *Fourier transform of some important signals*

Sl. No	*Time domain f(t)*		*Frequency domain F (jω)*	
1.		$\delta(t-t_0)$	$e^{-j\omega t_0}$	
2.		$e^{j\omega_0 t}$	$2\pi\delta(\omega-\omega_0)$	
3.		$\cos\omega_0 t$	$\pi[\delta(\omega+\omega_0)+\delta(\omega-\omega_0)]$	
4.		$\sin\omega_0 t$	$\frac{\pi}{j}[\delta(\omega+\omega_0)+\delta(\omega-\omega_0)]$	

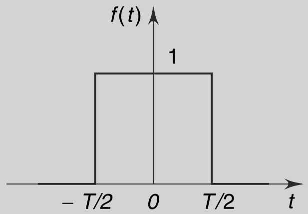
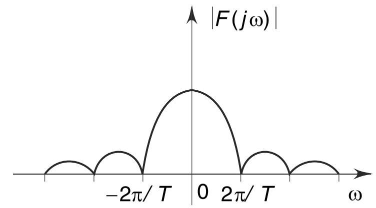
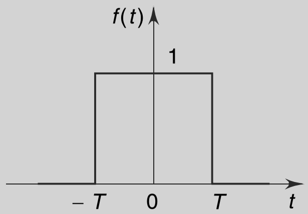
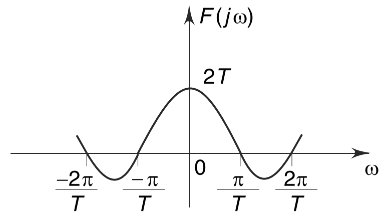
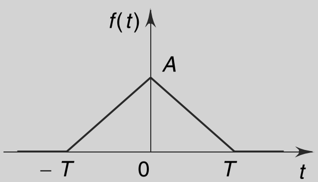
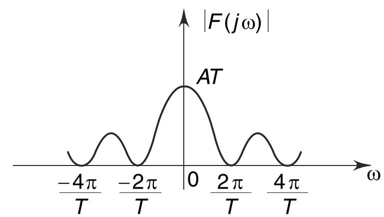
(*Contd.*)

Sl. No	Time domain $f(t)$		Frequency domain $F(j\omega)$	
5.		1	$2\pi\delta(\omega)$	
6.		$f(t) = \text{sgn}(t) = \frac{t}{\lvert t \rvert}$	$\frac{2}{j\omega}$	
7.		$u(t)$	$\pi\delta(\omega) + \frac{1}{j\omega}$	

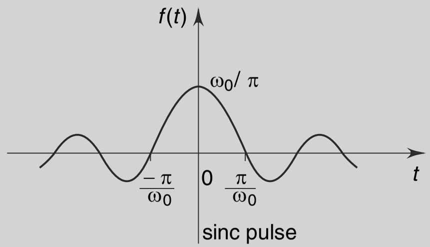
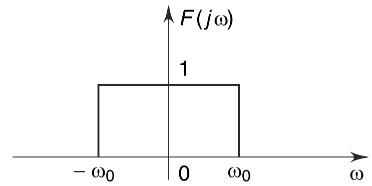
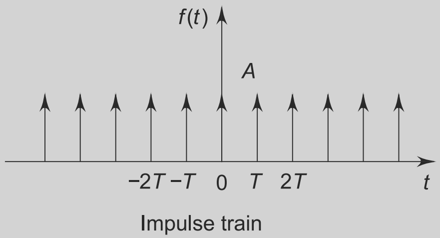
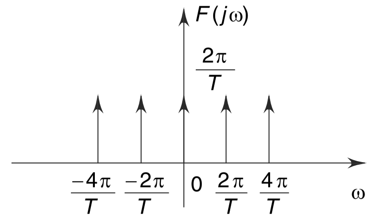
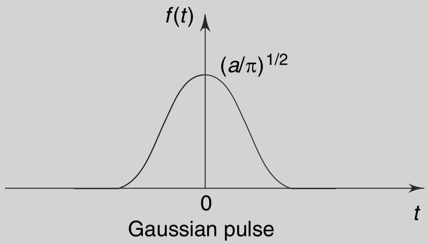
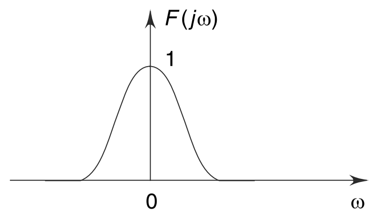
(*Contd.*)

Sl.No	Time domain $f(t)$		Frequency domain $F(j\omega)$	
8.		$e^{-at}u(t)$	$\dfrac{1}{a+j\omega}$	
9.		$e^{-a\|t\|}$	$\dfrac{2a}{a^2+\omega^2}$	
10.		$e^{-at}\cos\omega_0 t\cdot u(t)$	$\dfrac{a+j\omega}{(a+j\omega)^2+\omega_0^2}$	

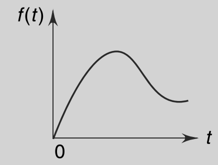
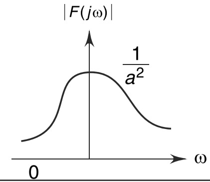
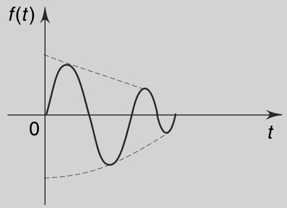
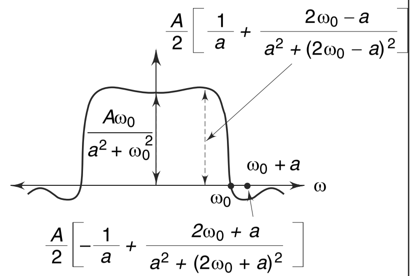
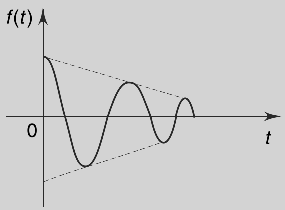
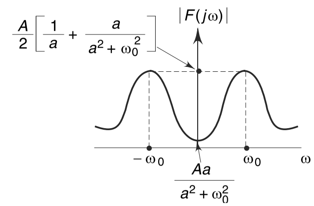
(*Contd.*)

Sl.No	*Time domain f(t)*		*Frequency domain F (j ω)*	
11.		$u\left(t+\frac{T}{2}\right)-u\left(t-\frac{T}{2}\right)$	$T\frac{\sin\frac{\omega T}{2}}{\frac{\omega T}{2}}$	
12.		$f(t) = 1$, for $\|t\| \le T$ $= 0$, otherwise $f(t) = u(t+T) - u(t-T)$	$2T\frac{\sin(\omega T)}{\omega T} = 2T \operatorname{sinc}(\omega T)$	
13.	 Triangular Pulse	$f(t) = A\left[1-\frac{\|t\|}{T}\right],\ \|t\| \le T$ $= 0$, elsewhere	$AT\left[\frac{\sin\frac{\omega T}{2}}{\frac{\omega T}{2}}\right]^2 = AT \operatorname{sinc}^2\left(\frac{\omega T}{2}\right)$	

(*Contd.*)

Sl.No	Time domain f(t)		Frequency domain F (j ω)	
14.		$\frac{\omega_0}{\pi}\,\mathrm{sinc}\left[\frac{\omega_0 t}{\pi}\right]$	$u(\omega+\omega_0) - u(\omega-\omega_0)$	
15.		$\sum_{n=-\infty}^{\infty} \delta(t-nT)$	$\frac{2\pi}{T}\sum_{n=-\infty}^{\infty} \delta\left(\omega-\frac{2\pi v}{T}\right)$	
16.		$\left(\frac{a}{\pi}\right)^{\frac{1}{2}} \exp(-at)^2$	$\exp\left(\frac{-\pi^2 f^2}{a}\right)$	

(*Contd.*)

Sl.No	Time domain $f(t)$		Frequency domain $F(j\omega)$	
17.		$t \exp(-at)u(t)$	$\frac{1}{(a+j\omega)^2}$	
18.		$\frac{t^{n-1}}{(n-1)!}\exp(-at)u(t)$	$\frac{1}{(a+j\omega)^n}$	
19.		$A \exp(-at) \sin(\omega_0 t)u(t)$	$\frac{A\omega_0}{(a+j\omega)^2+\omega_0^2}$	
20.		$A \exp(-at) \cos(\omega_0 t)u(t)$	$A\frac{a+j\omega}{(a+j\omega)^2+\omega_0^2}$	
21.		$\cos\omega_0 t\,[u(t+T)-u(t-T)]$	$T\left[\frac{\sin(\omega-\omega_0)T}{(\omega-\omega_0)T}+\frac{\sin(\omega+\omega_0)T}{(\omega+\omega_0)T}\right]$	

The Fourier transform of $f(t)$ is

$$F(j\omega) = \mathcal{F}[f(t)] = \int_{-\infty}^{\infty} f(t) e^{-j\omega t}\, \mathrm{d}t$$

$$= \int_{-T/2}^{T/2} 1 \cdot e^{-j\omega t}\, dt = \frac{1}{-j\omega}[e^{-j\omega t}]_{-T/2}^{T/2}$$

$$= \frac{1}{-j\omega}\left[e^{-j\omega T/2} - e^{-j\omega T/2}\right] = T \cdot \frac{\sin\left(\frac{\omega T}{2}\right)}{\left(\frac{\omega T}{2}\right)} = T \operatorname{sinc}\left(\frac{\omega T}{2}\right)$$

Hence, the amplitude spectrum is $|F(j\omega)| = T\left|\operatorname{sinc}\left(\frac{\omega T}{2}\right)\right|$

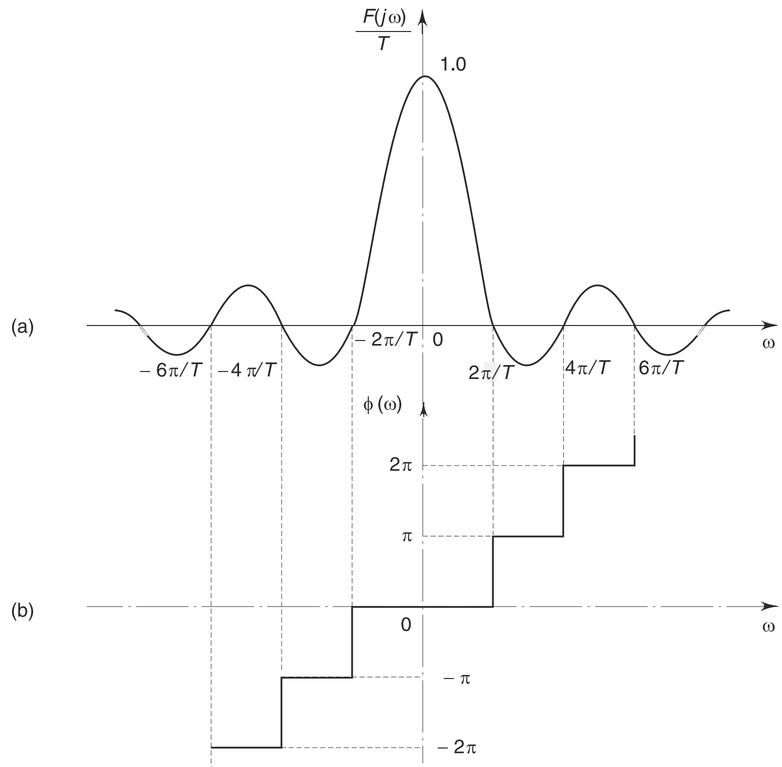


Fig. 2.5 (a) *Amplitude Spectrum and*
(b) *Phase Spectrum of the Single Gate Function*

and the phase spectrum is $\underline{|F(j\omega)} = \begin{cases} 0, & \operatorname{sinc}\left(\frac{\omega T}{2}\right) > 0 \\ \pi & \operatorname{sinc}\left(\frac{\omega T}{2}\right) < 0 \end{cases}$

The amplitude and phase spectra are shown in Fig. 2.5.

Comments

(1) $\underset{x\to 0}{Lt}\ \frac{\sin x}{x} = 1$, where $x = \frac{\omega T}{2}$. The phase value of the amplitude is one.

(2) The zeros occur when $\frac{\sin x}{x} = 0$

Therefore, sin $x = 0$ or $x = n\,\pi$ where $n = 1, 2$, etc...

and $x = \frac{\omega T}{2} = \pi, 2\pi, \ldots$ (or) $f = \frac{1}{T}, \frac{2}{T}, \ldots$

(3) There will be a shift in phase by π radians when the phase curve changes polarity from positive to negative or vice versa.

Example 2.16 *Determine the Fourier transform of the double gate function shown in Fig. E2.16, using time-shifting theorem.*

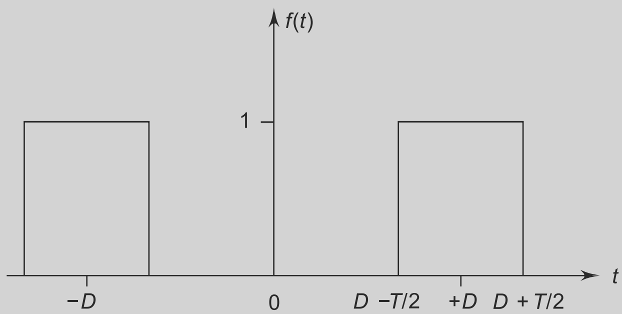


Fig. E2.16

Solution

We know that the Fourier transform of the single gate function is $T\,\text{sinc}\,\frac{\omega T}{2}$. Here in the double gate function, the single gate function is shifted by $+D$ and $-D$ at both sides, i.e. one gate function is centered about $+D$ and the other is centered about $-D$. Hence, by applying the shifting theorem of Fourier transform, we get

$$\mathcal{F}[f(t)] = e^{-j\omega D} T\,\text{sinc}\left(\frac{\omega T}{2}\right) + e^{j\omega D} T\,\text{sinc}\left(\frac{\omega T}{2}\right)$$

$$= 2T\cos(\omega D)\,\text{sinc}\left(\frac{\omega T}{2}\right)$$

2.8.2 Rectangular Pulse

Consider the rectangular pulse shown in Fig. 2.6. The analytic expression for the given pulse is

$$f(t) = \begin{cases} 1, & \text{for} \quad 0 < t < T \\ 0, & \text{otherwise} \end{cases}$$

The Fourier transform of $f(t)$ becomes

$$F(j\omega) = \mathcal{F}[f(t)] = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t$$

$$T = \int_0^T e^{-j\omega t}\,\mathrm{d}t = \left[\frac{e^{-j\omega t}}{-j\omega}\right]_0^T = \frac{e^{-j\omega T} - 1}{-j\omega}$$

$$= \frac{e^{-j\omega T/2}}{-j\omega}\left[e^{j\omega T/2} - e^{-j\omega T/2}\right]$$

$$= \frac{2e^{-j\omega T/2}}{\omega}\left[\frac{e^{-j\omega T/2} - e^{-j\omega T/2}}{2j}\right]$$

$$= Te^{-j\omega T/2}\left[\frac{\sin\left(\frac{\omega T}{2}\right)}{\left(\frac{\omega T}{2}\right)}\right] = Te^{-j\omega T/2}\,\mathrm{sinc}\left(\frac{\omega T}{2}\right)$$

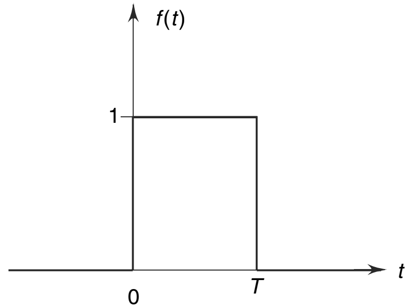


Fig. 2.6

Its amplitude and phase spectra are plotted in Fig. 2.7.

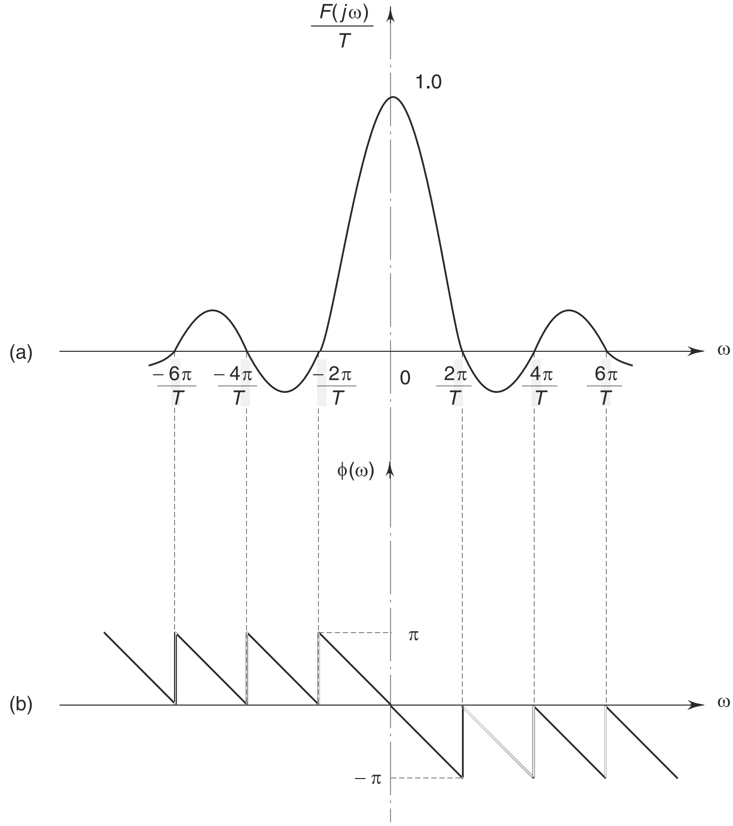


Fig. 2.7 *(a) Amplitude Spectrum and (b) Phase Spectrum of Rectangular Pulse*

Comments

(1) The phase of the amplitude spectrum is exactly the same as that in the previous case given in Fig. 2.7.

(2) There is an additional uniform phase shift factor $e^{-j\omega T/2}$ which changes the phase spectrum of the previous case.

(3) By using the time-shift theorem, $\mathcal{F}\left[f\left(t-\frac{T}{2}\right)\right] = F(j\omega)e^{-j\omega T/2}$ where $F(j\omega) = T\,\text{sinc}\left(\frac{\omega T}{2}\right)$, the above result can readily be obtained.

Example 2.17 *Find the Fourier transform of a rectangular pulse 2 seconds long with a magnitude of 10 volts as shown in Fig. E2.17.*

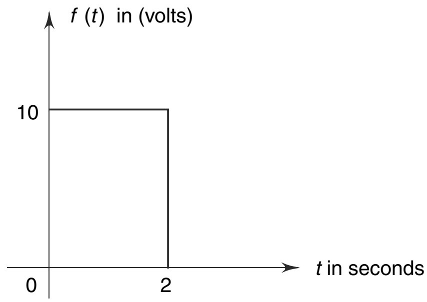


Fig. E2.17

Solution Fourier transform $F(j\omega)$ of the given pulse is given by

$$F(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t$$

$$= \int_0^2 10e^{-j\omega t}\,\mathrm{d}t = 10\left[\frac{e^{-j\omega t}}{-j\omega}\right]_0^2$$

$$= 10\left(\frac{e^{-j2\omega}-1}{-j\omega}\right)$$

$$= 10\frac{e^{-j\omega}}{-j\omega}\left[e^{-j\omega}-e^{j\omega}\right]$$

$$= 20\frac{e^{-j\omega}}{\omega}\left[\frac{e^{j\omega}-e^{-j\omega}}{2j}\right] = 20e^{-j\omega}\frac{\sin\omega}{\omega} = 20e^{-j\omega}\,\text{sinc}\,\omega$$

Alternate Method

$$F(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t$$

$$= \int_0^2 10\,e^{-j\omega t}\,\mathrm{d}t = 10\left[\frac{e^{-j\omega t}}{-j\omega}\right]_0^2$$

$$= 10\left(\frac{e^{-j2\omega}-1}{-j\omega}\right) = j10\left(\frac{e^{-j2\omega}-1}{\omega}\right)$$

$$= j10\left(\frac{\cos 2\omega - j\sin 2\omega - 1}{\omega}\right) = \frac{10}{\omega}[\sin 2\omega + j(\cos 2\omega - 1)]$$

Therefore,

$$|F(j\omega)| = \frac{10}{\omega}\sqrt{\sin^2 2\omega + (\cos 2\omega - 1)^2}$$

$$\angle F(j\omega) = \Phi(\omega) = \tan^{-1}\left(\frac{\cos 2\omega - 1}{\sin 2\omega}\right)$$

2.8.3 Exponential Pulse

The analytic expression for the exponential pulse shown in Fig. 2.8 is

$$f(t)=\begin{cases} e^{-at}, & \text{for } t \geq 0 \\ 0, & \text{for } t \geq 0 \end{cases}$$

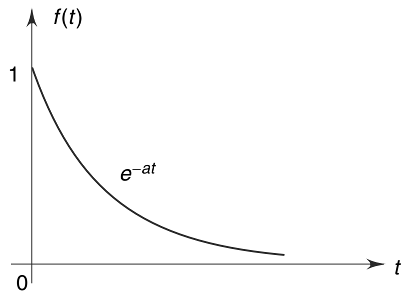


Fig. 2.8

Its Fourier transform is

$$F(j\omega)=\int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t$$

$$=\int_{-\infty}^{0} f(t)e^{-j\omega t}\,\mathrm{d}t+\int_{0}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t$$

$$=0+\int_{0}^{\infty} e^{-at}e^{-j\omega t}\,\mathrm{d}t, \text{ since } f(t)=0 \text{ for } t<0$$

$$=\int_{0}^{\infty} e^{-(a+j\omega)t}\,\mathrm{d}t=\frac{1}{a+j\omega}=\frac{1}{\sqrt{a^2+\omega^2}}\underline{\left|-\tan^{-1}\left(\frac{\omega}{a}\right)\right.}$$

Therefore, $$|F(j\omega)|=\frac{1}{(a^2+\omega^2)^{1/2}} \text{ and } \Phi(\omega)=-\tan^{-1}\left(\frac{\omega}{a}\right)$$

The amplitude and phase spectra of the exponential pulse are shown in Fig. 2.9(a) and (b).

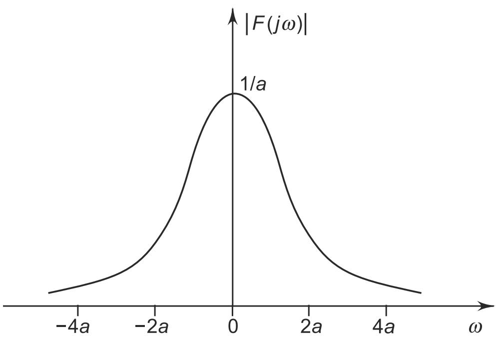


Fig. 2.9(a) *Amplitude Spectrum of Exponential Pulse*

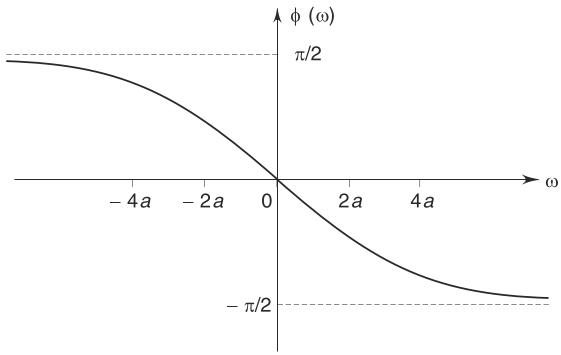


Fig. 2.9(b) *Phase Spectrum of the Exponential Pulse*

The energy spectral density is $$S(\omega)=\frac{1}{\pi}|F(j\omega)|^2=\frac{1}{\pi}\left|\frac{1}{a+j\omega}\right|^2=\frac{1}{\pi(a^2+\omega^2)}$$

Example 2.18 *Determine the Fourier transform for the double exponential pulse in Fig. E2.18(a) whose function is given by $f(t)=e^{-a|t|}$*

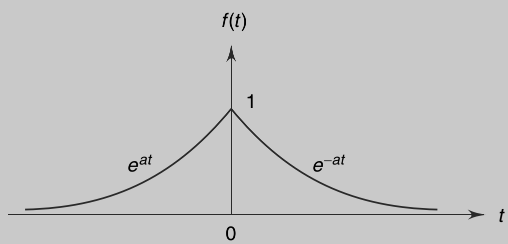


Fig. E2.18(a) *Double Exponential Pulse*

Solution

The expression for the double exponential pulse is $f(t) = e^{-a|t|}$

Hence,

$$f(t) = \begin{cases} e^{at}, & \text{for } t < 0 \\ e^{-at} & \text{for } t \geq 0 \end{cases}$$

$$F(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,dt$$

$$= \int_{-\infty}^{0} e^{at}e^{-j\omega t}\,dt + \int_{0}^{\infty} e^{-at}e^{-j\omega t}\,dt$$

$$= \int_{-\infty}^{0} e^{(a-j\omega)}\,dt + \int_{0}^{\infty} e^{-(a+j\omega)t}\,dt$$

$$= \frac{1}{(a-j\omega)}\left[e^{(a-j\omega t)}\right]_{-\infty}^{0} + \left[\frac{e^{-(a+j\omega)t}}{-(a+j\omega)}\right]_{0}^{\infty}$$

$$= \frac{1}{a-j\omega} + \frac{1}{a+j\omega}$$

Therefore,

$$F(j\omega) = \frac{2a}{a^2+\omega^2}.$$

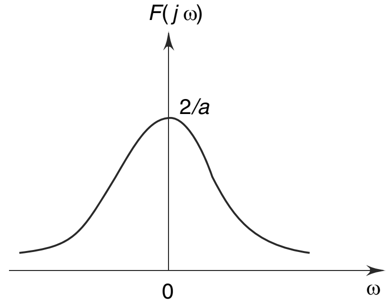


Fig. E2.18(b) *Spectrum of the Double Exponential Pulse*

The spectrum of the double exponential pulse is shown in Fig. E2.18(b).

2.8.4 Triangular Pulse

Consider the triangular pulse shown in Fig. 2.10.

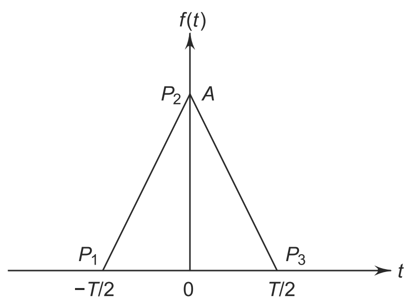


Fig. 2.10 *Triangular Pulse*

Equation of line $P_1 P_2$ is

$$f(t)_{P_1P_2} = \frac{A}{T/2}t + A = A\left(1 + \frac{2}{T}t\right)$$

Equation of line $P_2 P_3$ is

$$f(t)_{P_2P_3} = -\frac{A}{T/2}t + A = A\left(1 - \frac{2}{T}t\right)$$

Therefore,

$$f(t) = \begin{cases} A\left(1+\frac{2}{T}t\right) & \text{for } \frac{-T}{2} < t \leq 0 \\ A\left(1-\frac{2}{T}t\right) & \text{for } 0 \leq t \leq \frac{T}{2} \end{cases}$$

Now,

$$F(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,dt = \int_{-T/2}^{T/2} f(t)e^{-j\omega t}\,dt$$

$$= \int_{-T/2}^{0} f(t)e^{-j\omega t}\,dt + \int_{0}^{T/2} f(t)e^{-j\omega t}\,dt$$

$$= \int_{-T/2}^{0} A\left(1+\frac{2}{T}t\right)e^{-j\omega t}\,\mathrm{d}t + \int_{0}^{T/2} A\left(1-\frac{2}{T}t\right)e^{-j\omega t}\,\mathrm{d}t$$

$$= A\int e^{-j\omega t}\,\mathrm{d}t + A\int e^{-j\omega t}\,\mathrm{d}t + \frac{2A}{T}\int_{-T/2}^{0} te^{-j\omega t}\,\mathrm{d}t - \frac{2A}{T}\int_{0}^{T/2} te^{-j\omega t}\,\mathrm{d}t$$

(The first integral is printed with upper limit 0 and no lower limit; the second with upper limit $T/2$ and no lower limit.)

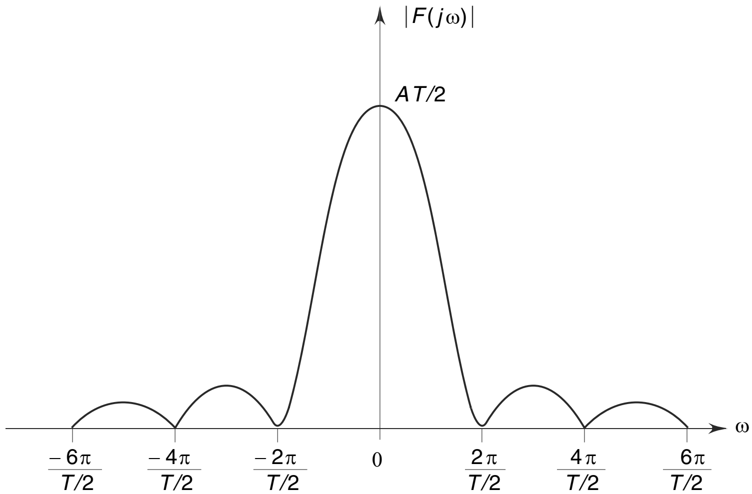


Fig. 2.11 *The Frequency Spectrum of a Single Triangular Pulse*

Changing t to $-t$ in the first and third integrals, we obtain

$$F(j\omega) = A\left[\int_{0}^{T/2} e^{j\omega t}\,\mathrm{d}t + \int_{0}^{T/2} e^{-j\omega t}\,\mathrm{d}t\right] - \frac{2A}{T}\left[\int_{0}^{T/2} te^{j\omega t}\,\mathrm{d}t + \int_{0}^{T/2} te^{-j\omega t}\,\mathrm{d}t\right]$$

$$= A\int_{0}^{T/2} 2\cos\omega t\,\mathrm{d}t - \frac{2A}{T}\int_{0}^{T/2} 2t\cos\omega t\,\mathrm{d}t$$

$$= 2A\frac{\sin\frac{\omega t}{2}}{\omega} - \frac{4A}{T}\left[\frac{T}{2}\frac{\sin\frac{\omega T}{2}}{\omega} - \int_{0}^{T/2}\frac{\sin\omega t}{\omega}\,\mathrm{d}t\right]$$

$$= 2A\frac{\sin\frac{\omega t}{2}}{\omega} - \frac{4A}{T}\left[\frac{T}{2}\frac{\sin\frac{\omega T}{2}}{\omega} + \left\{\frac{\cos\omega t}{\omega^2}\right\}_{0}^{T/2}\right]$$

$$= 2A\frac{\sin\frac{\omega t}{2}}{\omega} - \frac{4A}{T}\left[\frac{T}{2}\frac{\sin\frac{\omega T}{2}}{\omega} + \frac{\cos\frac{\omega T}{2}}{\omega^2} - \frac{1}{\omega^2}\right]$$

$$= \frac{2A}{\omega}\sin\frac{\omega T}{2} - \frac{2A}{\omega}\sin\frac{\omega T}{2} - \frac{4A}{T\omega^2}\left(\cos\frac{\omega T}{2} - 1\right)$$

$$= \frac{4A}{T\omega^2}\left(1 - \cos\frac{\omega T}{2}\right)$$

$$= \frac{4A}{T\omega^2}\left[2\sin^2\frac{\omega T}{4}\right] = \frac{8A}{T\omega^2}\sin^2\frac{\omega T}{4}$$

$$= \frac{8A}{T\omega^2}\left(\frac{\omega T}{4}\right)^2 \frac{\sin^2\frac{\omega T}{4}}{\left(\frac{\omega T}{4}\right)^2} = \frac{AT}{2}\operatorname{sinc}^2\left(\frac{\omega T}{4}\right)$$

The frequency spectrum of the single triangular pulse is shown in Fig. 2.11.

2.8.5 Gaussian Pulse

The pulse defined by $f(t) = e^{-\pi t^2}$ is called ***Gaussian Pulse,*** which is plotted in Fig. 2.12.

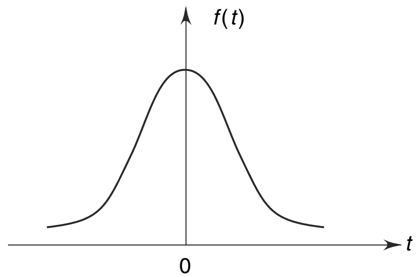


Fig. 2.12 *Gaussian Pulse*

The Fourier transform of this function is

$$\mathcal{F}[e^{-\pi t^2}] = \int_{-\infty}^{\infty} e^{-\pi t^2} e^{-j\omega t}\,\mathrm{d}t$$

$$= \int_{-\infty}^{\infty} e^{-(\pi t^2 + j\omega t)}\,\mathrm{d}t$$

Substituting

$$\pi t^2 + j\omega t = \left(\sqrt{\pi}t + \frac{j\omega}{2\sqrt{\pi}}\right)^2 + \frac{\omega^2}{4\pi}, \text{ we get}$$

$$\mathcal{F}[e^{-\pi t^2}] = \int_{-\infty}^{\infty} e^{-\left(\sqrt{\pi}t + \frac{j\omega}{2\sqrt{\pi}}\right)^2} e^{-\frac{-\omega^2}{4\pi}}\,\mathrm{d}t = e^{-\frac{-\omega^2}{4\pi}}\,\mathrm{d}t\int_{-\infty}^{\infty} e^{-\left(\sqrt{\pi}t + \frac{j\omega}{2\sqrt{\pi}}\right)^2}\,\mathrm{d}t$$

Putting $u = \sqrt{\pi}t + \frac{j\omega}{2\sqrt{\pi}}, \mathrm{d}u = \sqrt{\pi}\,\mathrm{d}t$ and hence, $\mathrm{d}t = \frac{\mathrm{d}u}{\sqrt{\pi}}$

$$\mathcal{F}[e^{-\pi t^2}] = e^{\frac{\omega^2}{4\pi}} 2\int_0^{\infty} e^{-u^2}\frac{du}{\sqrt{\pi}}$$

$$= e^{-\frac{\omega^2}{4\pi}} \cdot \frac{2}{\sqrt{\pi}} \cdot \frac{\sqrt{\pi}}{2}, \quad \text{since} \int_0^{\infty} e^{-u^2}\,\mathrm{d}u = \frac{\sqrt{\pi}}{2} = e^{-\frac{\omega^2}{4\pi}} = e^{-\pi/f^2}$$

Hence, $\mathcal{F}\left[e^{-\pi t^2}\right] = e^{-\pi f^2}$

The area under the Gaussian pulse is

$$\int_{-\infty}^{\infty} f(t)\,\mathrm{d}t = \int_{-\infty}^{\infty} e^{\pi t^2}\,\mathrm{d}t = F(0) = \left[e^{-\pi f^2}\right]_{at\, f=0} = 1$$

When both the area under the curve and the central ordinate of a pulse are unity, then we say that the pulse is ***normalised***. Hence, we conclude that the normalised Gaussian pulse is its own Fourier transform as derived by

$$e^{-\pi t^2} \Leftrightarrow e^{-\pi f^2}$$

Example 2.19 *Find the Fourier transform of the Gaussian pulse $f(t) = e^{-a^2t^2}$*

Solution $$\mathcal{F}\left[e^{-a^2t^2}\right] = \int_{-\infty}^{\infty} e^{-a^2t^2} e^{-j\omega t}\,\mathrm{d}t \int_{-\infty}^{\infty} e^{-(a^2t^2 + j\omega t)}\,\mathrm{d}t$$

Substituting $a^2t^2 + j\omega t = \left(at + \frac{j\omega}{2a}\right)^2 + \frac{\omega^2}{4a^2}$, we get

$$\mathcal{F}\left[e^{-a^2t^2}\right] = \int_{-\infty}^{\infty} e^{-\left(at+\frac{j\omega}{2a}\right)^2} e^{\frac{-\omega^2}{4a^2}}\,\mathrm{d}t = e^{\frac{-\omega^2}{4a^2}} \int_{-\infty}^{\infty} e^{-\left(at+\frac{j\omega}{2a}\right)^2}\,\mathrm{d}t$$

Putting $u = at + \frac{j\omega}{2a}$, $\mathrm{d}u = a\,\mathrm{d}t$ and so $\mathrm{d}t = \frac{\mathrm{d}u}{a}$

$$\mathcal{F}[e^{-a^2t^2}] = e^{-\frac{\omega^2}{4a^2}} 2\int_0^{\infty} e^{-u^2} \frac{\mathrm{d}u}{a}. \quad \text{Here} \quad \int_0^{\infty} e^{-u^2}\,\mathrm{d}u = \frac{\sqrt{\pi}}{2}$$

$$= e^{-\frac{\omega^2}{4a^2}} \frac{2}{a}\frac{\sqrt{\pi}}{2} = \frac{\sqrt{\pi}}{a} e^{-\frac{\omega^2}{4a^2}} = \frac{\sqrt{\pi}}{a} e^{-\left(\frac{\omega}{2a}\right)^2}$$

Therefore, $$\mathcal{F}\left[e^{-a^2t^2}\right] = \frac{\sqrt{\pi}}{a} e^{-\left(\frac{\pi f}{a}\right)^2}$$

2.8.6 Impulse Function (Unit Impulse)

The impulse function is called the **Dirac delta function** $\delta(t)$ which has an infinite amplitude and is infinitely narrow. This is defined as

(i) $\delta(t) = 0$ for all values except at $t = 0$.

(ii) $\int_{-\infty}^{\infty} \delta(t)\,\mathrm{d}t = 1$, i.e. the area within the pulse is unity.

The Fourier transform of the impulse function $\delta(t)$ is obtained as

$$F(j\omega) = \mathcal{F}[\delta(t)] = \int_{-\infty}^{\infty} \delta(t) e^{-j\omega t}\,\mathrm{d}t = 1$$

Hence, we have the pair $\delta(t) \Leftrightarrow 1$.

The frequency spectrum of the impulse function $\delta(t)$ shown in Fig. 2.13(a) has a constant amplitude and extends over positive and negative frequencies.

Using the time-shift theorem, we get $\delta(t - t_0) \Leftrightarrow e^{-j\omega t_0}$

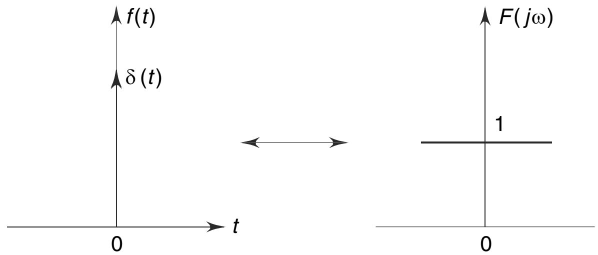


Fig. 2.13(a) *Impulse Function and its Spectrum*

The shifted impulse and its amplitude and phase spectra are shown in Fig. 2.13(b).

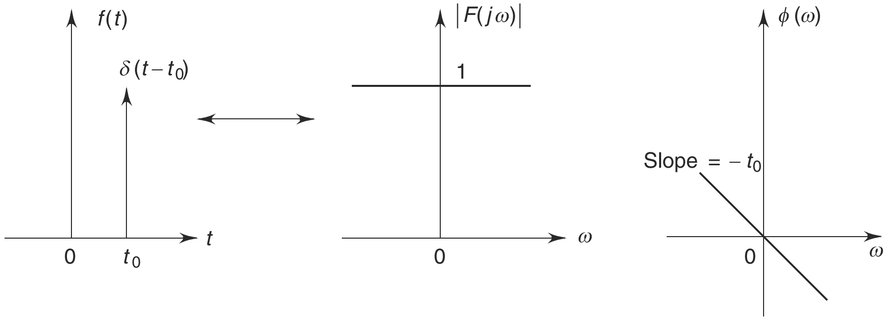


Fig. 2.13(b) *Shifted Impulse and its Spectrum*

The inverse Fourier transform of the unit impulse is given by

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \delta(\omega)\, e^{j\omega t}\, d\omega$$

$$= \frac{1}{2\pi}\angle \omega t$$

or $\quad |f(t)| = \dfrac{1}{2\pi}$. Therefore $\dfrac{1}{2\pi} \Leftrightarrow \delta(\omega)$ or $1 \Leftrightarrow 2\pi\delta(\omega)$

Hence, the Fourier transform of the constant function is an impulse at the origin with an area equal to 2π, as illustrated in Fig. 2.13 (c).

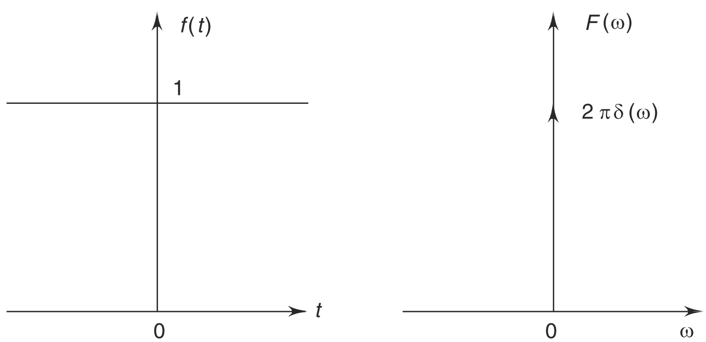


Fig. 2.13 (c) *Constant Function and its Spectrum*

2.8.7 Unit Step Function

The unit step function is obtained by suddenly closing a switch of a d.c. circuit. For easier analysis, the waveform of the unit step function is split into two component waveforms as shown in Fig. 2.14.

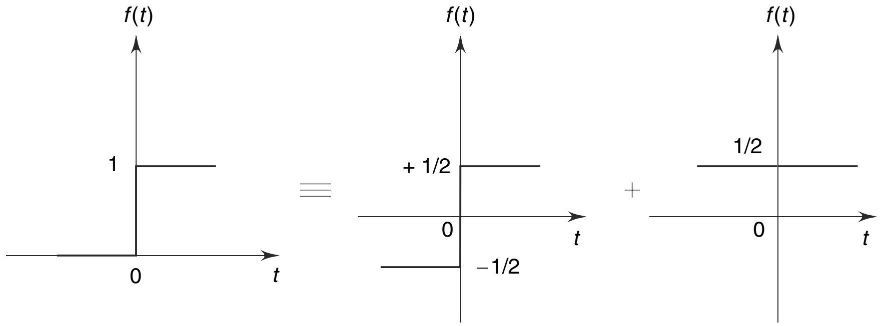


Fig. 2.14 *Unit Step Function*

The first waveform is similar to the signum function with half amplitude. Therefore, the Fourier transform function is given by

$$F_1(j\omega) = \frac{1}{2}\left(\frac{2}{j\omega}\right) = \frac{1}{j\omega}$$

The second waveform is related to the unit impulse function and hence its Fourier transform is given by

$$F_2(j\omega) = \frac{1}{2}\{2\pi\delta(\omega)\} = \pi\delta(\omega)$$

Therefore, the Fourier transform of the step function becomes

$$F(j\omega) = F_1(j\omega) + F_2(j\omega)$$

$$= \frac{1}{j\omega} + \pi\delta(\omega)$$

The Fourier transform of the unit step function is shown in Fig. 2.15.

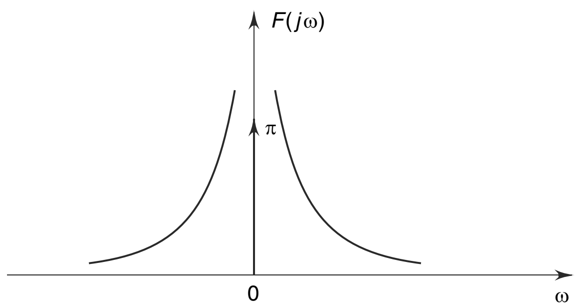


Fig. 2.15 *Fourier Transform of the Unit Step Function*

2.8.8 Sinusoidal Functions

The Fourier transforms of the sinusoidal functions cos $\omega_0 t$ and sin $\omega_0 t$ are obtained as given below.

$$\mathcal{F}[\cos\omega_0 t] = \mathcal{F}\left[\frac{e^{j\omega_0 t} + e^{-j\omega_0 t}}{2}\right]$$

Using the transform pair $e^{j\omega_0 t} \Leftrightarrow 2\pi\delta(\omega - \omega_0)$, we get

$$\mathcal{F}[\cos\omega_0 t] = \pi[\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]$$

Similarly, $$\mathcal{F}[\sin\omega_0 t] = \mathcal{F}\left[\frac{e^{j\omega_0 t} + e^{-j\omega_0 t}}{2j}\right] = j\pi[\delta(\omega + \omega_0) - \delta(\omega - \omega_0)]$$

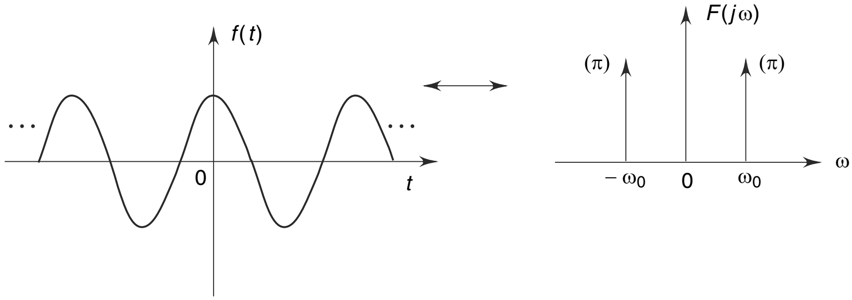


Fig. 2.16(a) *Cosine Wave and its Spectrum*

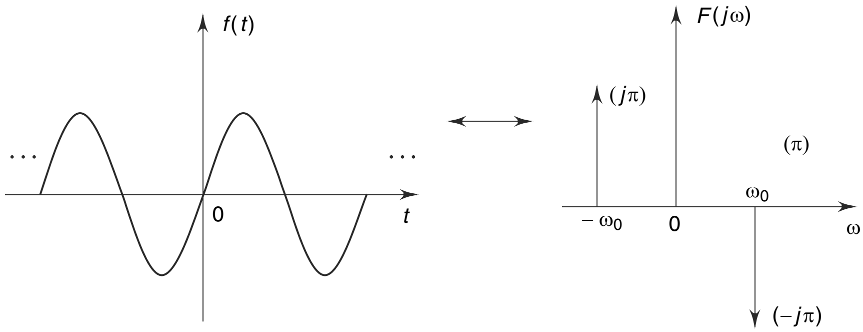


Fig. 2.16(b) *Sinusoidal Wave and its Spectrum*

These transform pairs are shown in Figs. 2.16 (a) and (b). Using the frequency convolution theorem we have

$$\mathcal{F}[f(t)\cos\omega_0 t] = \frac{1}{2\pi}\Big[F(j\omega) * \{\delta(\omega-\omega_0)+\delta(\omega+\omega_0)\}\Big]$$

$$= F(j\omega) * \frac{1}{2}[\delta(\omega+\omega_0)+\delta(\omega-\omega_0)]$$

$$= \frac{1}{2}[F(\omega + j\omega_0) + F(j\omega-\omega_0)]$$

2.8.9 Signum Function

The signum function denoted by sgn (t), is defined by

$$\text{sgn}\,(t) = \begin{cases} -1, & \text{if} \quad t < 0 \\ 0, & \text{if} \quad t = 0 \\ 1, & \text{if} \quad t > 0 \end{cases}$$

or sgn (t) = $u(t) - u(-t)$

It is plotted in Fig. 2.17.

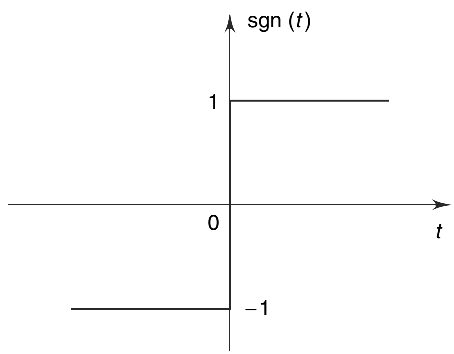


Fig. 2.17 *Signum Function*

$$F(j\omega) = \mathcal{F}[\text{sgn}\,(t)] = \int_{-\infty}^{\infty} \text{sgn}\,(t)\,e^{j\omega t}\,\mathrm{d}t$$

$$= \int_{-\infty}^{0} (-1)e^{-j\omega t}\,\mathrm{d}t + \int_{0}^{\infty} (1)\,e^{-j\omega t}\,\mathrm{d}t$$

$$= -\left[\frac{e^{-j\omega t}}{-j\omega}\right]_{-\infty}^{0} + \left[\frac{e^{-j\omega t}}{-j\omega}\right]_{0}^{\infty} = \frac{1}{j\omega} + \frac{1}{j\omega} = \frac{2}{j\omega}$$

Therefore, $\mathcal{F}[\text{sgn}(t)] = \dfrac{2}{j\omega}$

Hence, we have the transform pair $\text{sgn}(t) \Leftrightarrow \dfrac{2}{j\omega}$

The amplitude and phase spectra of the signum function are shown in Fig. 2.18 (a) and (b) respectively.

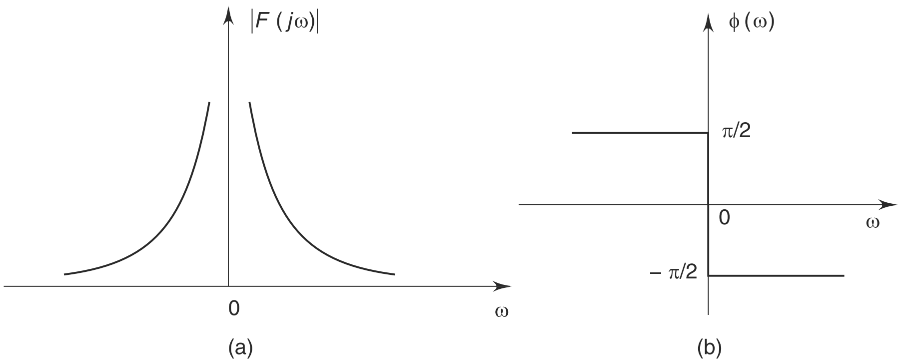


Fig. 2.18 *(a) Amplitude and (b) Phase Spectra of the Signum Function*

Example 2.20 *Determine the Fourier transform of $f(t) = e^{-a|t|}$ sgn (t).*

Solution

$$f(t) = e^{-a|t|}\,\text{sgn}(t)$$

$$= \begin{cases} -e^{-at}, \text{ for } t < 0 \\ \;\; e^{-at}, \text{ for } t > 0 \end{cases}$$

Therefore, $\mathcal{F}[f(t)] = \displaystyle\int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,dt$

$$= -\int_{-\infty}^{0} e^{at}e^{-j\omega t}\,dt + \int_{0}^{\infty} e^{-at}e^{-j\omega t}\,dt = -\int_{-\infty}^{0} e^{(a-j\omega)t}\,dt + \int_{0}^{\infty} e^{-(a+j\omega)t}\,dt$$

$$= -\left[\frac{e^{(a-j\omega)t}}{(a-j\omega)}\right]_{-\infty}^{0} + \left[\frac{e^{-(a-j\omega)t}}{-(a+j\omega)}\right]_{0}^{\infty}$$

$$= \left[\frac{1}{a-j\omega}\right] + \left[0 + \frac{1}{a+j\omega}\right]$$

$$= \left[\frac{1}{j\omega - a} + \frac{1}{j\omega + a}\right] = \frac{j2\omega}{(j\omega)^2 - a^2} = \frac{-j2\omega}{a^2 + \omega^2}$$

Hence, $F(j\omega) = \mathcal{F}\left[e^{-a|t|}\,\text{sgn}(t)\right] = \dfrac{-j2\omega}{a^2 + \omega^2}$

It is a pure imaginary function of ω.

Since $F(-j\omega) = \dfrac{-j2(-\omega)}{a^2 + (-\omega^2)} = \dfrac{j2\omega}{a^2 + \omega^2} = -F(j\omega),$ it is also an odd function of ω. Hence, this is proved that if $f(t)$ is real and odd, then $F(j\omega)$ is pure imaginary and odd.

The phase of $F(j\omega)$ is 90° for all $\omega < 0$ and –90° for all $\omega > 0$.

The amplitude of $F(j\omega)$ is $|F(j\omega)| = \dfrac{2|\omega|}{a^2 + \omega^2}$.

2.8.10 Pulse Train Signal or Periodic Gate Function

From Eq. (2.10), the Fourier series representation for this pulse train signal shown in Fig. 2.19 is

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t}$$

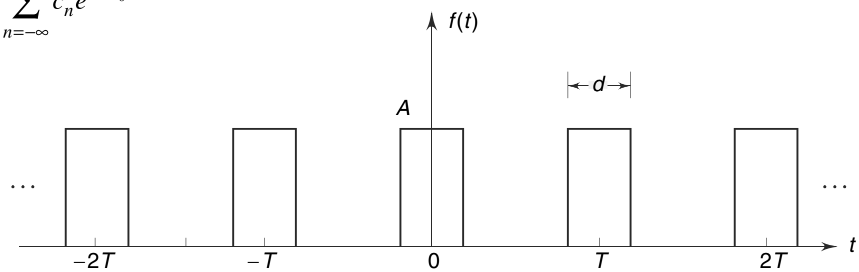


Fig. 2.19 *Pulse Train Signal*

To find the spectrum, we calculate the Fourier series coefficients as

$$c_n = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-jn\omega_0 t}\, \mathrm{d}t$$

where

$$\omega_0 = \frac{2\pi}{T} \text{ and } n \text{ is the number of the harmonic.}$$

$$c_n = \frac{1}{T} \int_{-d/2}^{d/2} A e^{-jn\omega_0 t} \mathrm{d}t$$

$$= \frac{-A}{jn\omega_0 T} \left[e^{-jn\omega_0 t} \right]_{-d/2}^{d/2} = \frac{-A}{n\omega_0 T} \left[\frac{e^{jn\omega_0 d/2} - e^{-jn\omega_0 d/2}}{2j} \right]$$

$$= \frac{2A}{n\omega_0 T} \sin\left(\frac{n\omega_0 d}{2} \right) = \frac{Ad}{T} \left[\frac{\sin\left(\frac{n\omega_0 d}{2} \right)}{\frac{n\omega_0 d}{2}} \right] = \frac{Ad}{T} \operatorname{sinc}\left(\frac{n\omega_0 d}{2} \right)$$

The sinc function oscillates with period 2π and decays with increasing x. It has zeros at $n\pi$, $n = \pm 1, \pm 2, \ldots$ and is an even function of x. Then the spectrum of the periodic gate function is

$$c_n = \frac{Ad}{T}\operatorname{sinc}\left(\frac{n\omega_0 d}{2}\right)$$

$$= \frac{Ad}{T}\operatorname{sinc}\left(\frac{n\left(\frac{2\pi}{T}\right)d}{2}\right) = \frac{Ad}{T}\operatorname{sinc}\left(\frac{n\pi d}{T}\right)$$

Thus we can represent $f(t)$ as

$$f(t) = \sum_{n=-\infty}^{\infty} \frac{Ad}{T}\operatorname{sinc}\left(\frac{n\pi d}{T}\right)e^{jn\frac{2\pi}{T}t}$$

$$F[f(t)] = \frac{2\pi Ad}{T}\sum_{n=-\infty}^{\infty}\operatorname{sinc}\left(\frac{n\pi d}{T}\right)\delta(\omega - n\omega_0)$$

From the above equation, we find that the values c_n are real.

Example 2.21 *What percentage of the total power is contained within the first zero crossing of the spectrum envelope for $f(t)$ as given in Fig. E2.21 (a).*

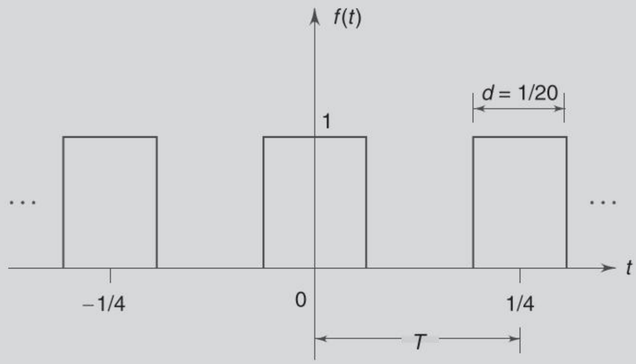


Fig. E2.21 *(a) Periodic Pulse Train Signal f(t)*

Solution $f(t) = \sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t}$

where $c_n = \frac{1}{T}\int_{-d/2}^{d/2} f(t)e^{-jn\omega_0 t}\,dt = \frac{Ad}{T}\operatorname{sinc}\left(\frac{n\pi d}{T}\right)$

$$= \frac{1}{5}\operatorname{sinc}\left(\frac{n\pi}{5}\right), \text{ since } A=1,\ T=1/4 \text{ and } d = \frac{1}{20}$$

The total power in $f(t)$ is

$$P = \frac{1}{T}\int_{-d/2}^{d/2} f^2(t)\,dt = 4\int_{-\frac{1}{40}}^{\frac{1}{40}} (1)^2 dt = 4\left(\frac{1}{40}+\frac{1}{40}\right) = 0.2$$

The spectrum *f*(*t*) is shown in Fig. E2.21(b). The first zero crossing occurs at 40π rad/sec, and there are four harmonics plus the d.c. value within the first zero crossing. Thus, the power contained within the first zero crossing of the spectrum envelope is

$$P_{fzc} = |c_o|^2 + 2\left[|c_1|^2 + |c_2|^2 + |c_3|^2 + |c_4|^2\right]$$

$$= \left(\frac{1}{5}\right)^2 + \frac{2}{5^2}\left[\text{sinc}^2\left(\frac{\pi}{5}\right) + \text{sinc}^2\left(\frac{2\pi}{5}\right) + \text{sinc}^2\left(\frac{3\pi}{5}\right) + \text{sinc}^2\left(\frac{4\pi}{5}\right)\right]$$

$$= \left(\frac{1}{5}\right)^2 + \frac{2}{5^2}[0.875 + 0.573 + 0.255 + 0.055] \approx 0.04 + 0.141 = 0.181$$

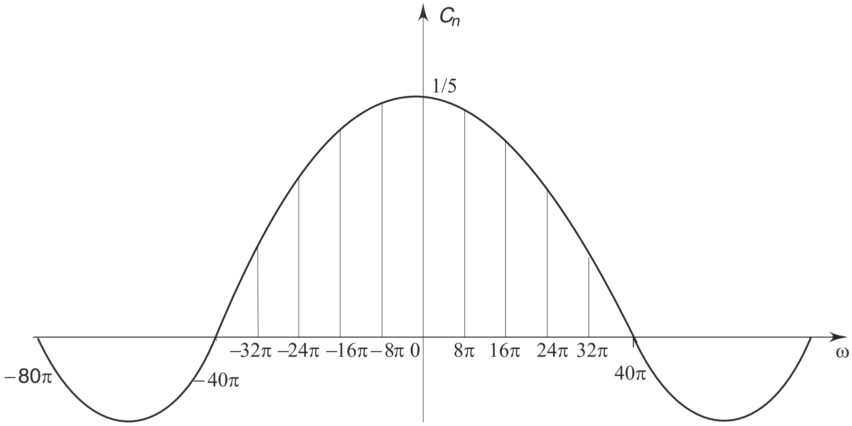


Fig. E2.21(b) *Spectrum of the Function*

Comparing the power P and P_{fzc}, we see that (0.181/0.2) × 100 = 90.5% of the total power in $f(t)$ is contained within the first zero crossing of the spectrum for $f(t)$.

2.8.11 Unit Impulse Train

Since $f(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT)$ is a periodic function, as shown in Fig. 2.20, the Fourier series representation of this unit impulse train is $f(t) = \sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t}$

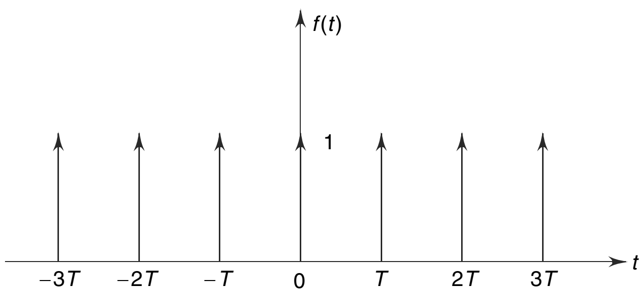


Fig. 2.20 *Unit Impulse Train*

where $$c_n = \frac{1}{T}\int_{-T/2}^{T/2} f(t)e^{-jn\omega_0 t}\,dt = \frac{1}{T}\int_{-T/2}^{T/2} \delta(t)e^{-jn\omega_0 t}\,dt = \frac{1}{T}$$

Substituting this result in the above Fourier series representation, we obtain

$$f(t) = \frac{1}{T}\sum_{n=-\infty}^{\infty} e^{jn\omega_0 t}$$

Therefore, $$F(j\omega) = \mathcal{F}[f(t)] = \mathcal{F}\left[\frac{1}{T}\sum_{n=-\infty}^{\infty} e^{jn\omega_0 t}\right]$$

$$= \frac{2\pi}{T}\sum_{n=-\infty}^{\infty} \delta(\omega - n\omega_0), \quad \text{where } \omega_0 = \frac{2\pi}{T}$$

Hence, $$\sum_{k=-\infty}^{\infty} \delta(t-kT) \Leftrightarrow \omega_0 \sum_{n=-\infty}^{\infty} \delta(\omega - n\omega_0)$$

Each unit impulse of the unit impulse train in the time domain has a transform of an impulse train in the frequency domain. The locations of the impulses are at $n\omega_0 = n2\pi/T$ where $n = 0, \pm 1, \pm 2, \ldots$ The area for each impulse in the frequency domain is ω_0.

—FOURIER TRANSFORM OF POWER AND ENERGY SIGNALS 2.9

2.9.1 Power Signal

The average power of a signal $x(t)$ over a single period (t_1, t_1+T) is given by

$$P_{\text{av}} = \frac{1}{T}\int_{t_2}^{t_1+T} |x(t)|^2\,dt$$

where $x(t)$ is a complex periodic signal.

A signal $f(t)$ is called a power signal if the average power is expressed by

$$P_x = \underset{t\to\infty}{Lt}\,\frac{1}{2T}\int_{-T}^{T} |x(t)|^2$$

$$P_{\text{av}} = \frac{1}{T}\int_{-T/2}^{T/2} |x(t)|^2\,dt, \text{ for a continuous periodic signal with period } T;$$

or $$P_{\text{av}} = \frac{1}{N}\sum_{n=0}^{N-1} |x(n)|^2, \text{ for a digital signal with } x(n) = 0 \text{ for } n < 0;$$

is equal to a finite value, i.e. equal to the average power over a single period. If $x(t)$ *is bounded,* P_∞ *is finite. Hence, every bounded and periodic signal is a power signal. But it is true that a power signal is not necessarily a bounded and periodic signal.*

If the signal $x(t)$ contains finite signal power, i.e. $0 < P_x < \infty$, then $x(t)$ is called a power signal.

2.9.2 Energy Signal

A signal $x(t)$ is called an ***energy signal*** if its total energy over the interval $(-\infty, \infty)$ is finite, that is

$$E_x = \underset{T\to\infty}{Lt} \int_{-T}^{T} |x(t)|^2 \, dt < \infty$$

$$E_\infty = \int_{-\infty}^{\infty} |x(t)|^2 \, dt < \infty$$

If the signal $x(t)$ contains finite signal energy, i.e. $0 < E_x < \infty$, then $x(t)$ is called an energy signal.

For a digital signal, the energy is defined by

$$E = \sum_n |x(n)|^2$$

As $n \to \infty$, the energy of periodic signals becomes infinite, whereas the energy of aperiodic pulse-like signals have a finite value. So, these aperiodic pulse-like signals are called *energy signals*. Since the signal energy of a periodic signal is always either zero or infinite, any periodic signal cannot be an energy signal.

Note: There are signals for which neither P_∞ nor E_∞ are finite. If the signals contain infinite energy and zero power or infinite energy and infinite power, then the signals like impulse functions and $x(t) = t$ are neither energy nor power signals.

Example 2.22 *Determine the signal energy and signal power for*

(a) $f(t) = e^{-|t|}$ and (b) $f(t) = e^{-t}$.

Solution

(a)

$$E_\infty = \int_{-\infty}^{\infty} [e^{-|t|}]^2 \, dt$$

$$= \int_{-\infty}^{0} e^{2t} dt + \int_{0}^{-\infty} e^{-2t} dt = 2\int_{0}^{\infty} e^{-2t} dt$$

$$= -[e^{-2t}]_{t=0}^{\infty} = -[0-1] = 1$$

The signal power $P_\infty = 0$ since E_∞ is finite. Hence, the signal $f(t)$ is an energy signal.

(b)

$$E_T = \int_{-T}^{T} (e^{-t})^2 dt = \int_{-T}^{T} e^{-2t} dt = -\frac{1}{2}[e^{-2T} - e^{2T}]$$

As $T\to\infty$, E_T approaches infinity.

Its average power is

$$P_\infty = \underset{T\to\infty}{Lt} \frac{1}{2T} E_T = \underset{T\to\infty}{Lt} \frac{e^{2T} - e^{-2T}}{4T} = \infty \quad \text{[Using L' Hospital's rule]}$$

Hence, e^{-t} is neither an energy signal nor a power signal.

Example 2.23 *Compute the signal energy for* $x(t) = e^{-4t}\, u(t)$

$$u(t) = \begin{cases} 1, & \text{for} \quad t \ge 0 \\ 0, & \text{for} \quad t < 0 \end{cases}$$

Solution Since the value of the given function is zero between $-T$ to 0, the limits must be applied between 0 to T instead of $-T$ to T.

$$E_x = \underset{T\to\infty}{Lt} \int_{-T}^{T} |x(t)|^2\,dt = \underset{T\to\infty}{Lt} \int_{0}^{T} \left|e^{-4T}\right|^2 dt = \underset{T\to\infty}{Lt} \int_{0}^{T} e^{-8t}dt$$

$$= \underset{T\to\infty}{Lt} \left[\frac{e^{-8t}}{-8}\right]_0^T = \underset{T\to\infty}{Lt} \left[\frac{e^{-8t}}{-8} + \frac{1}{8}\right] = \frac{1}{8}$$

$$P_x = \underset{T\to\infty}{Lt} \frac{1}{2T} \int_{-T}^{T} |x(t)|^2\,dt$$

$$= \underset{T\to\infty}{Lt} \frac{1}{2T} \left[\frac{e^{-8t}}{-8}\right]_{-T}^{T} = \underset{T\to\infty}{Lt} \frac{(e^{-8T}-1)}{-16T} = 0.$$

Here, as the signal energy E_x is finite and the signal power $P_x = 0$, the signal is energy signal.

Example 2.24 *Determine the signal energy and signal power for the following complex-valued signal and determine whether it is an energy signal or a power signal.*

$$x(t) = Ae^{j2\pi\alpha t}$$

Solution Here, the signal period is $T = \dfrac{1}{\alpha}$. Since $x(t)$ is a periodic signal, it cannot be an energy signal. Therefore, the power signal is evaluated as

$$P_x = \frac{1}{T} \int_{t_1}^{t_1+T} |x(t)|^2\,dt$$

$$P_x = \alpha \int_{t_1}^{t_1+\left(\frac{1}{\alpha}\right)} \left|A\,e^{2\pi\alpha t}\right|^2 dt = \alpha \int_{t_1}^{t_1+\left(\frac{1}{\alpha}\right)} A^2 dt = \alpha[A^2 t]_{t_1}^{t_1+\frac{1}{\alpha}} = A^2$$

Since the signal has finite power, it is a power signal and $E_x = \infty$.

Example 2.25 *Determine the magnitude and phase spectrum of the pulse shown in Fig. E2.25(a).*

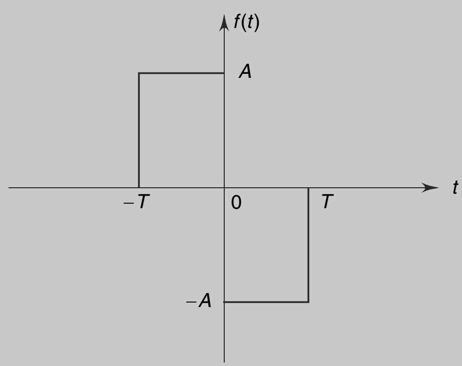


Fig. E2.25(a)

Solution Here

$$f(t)=\begin{cases} A, & \text{for} -T \le t \le 0 \\ -A, & \text{for } 0 \le t \le T \\ 0, & \text{otherwise} \end{cases}$$

$$F(j\omega)=\int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t=\int_{-\infty}^{0} f(t)e^{-j\omega t}\,\mathrm{d}t+\int_{0}^{\infty} f(t)e^{-j\omega t}\,\mathrm{d}t$$

$$=\int_{T}^{0} f(t)e^{-j\omega t}\,\mathrm{d}t+\int_{0}^{T} f(t)e^{-j\omega t}\,\mathrm{d}t=\int_{-T}^{0} A\,e^{-j\omega t}\,\mathrm{d}t+\int_{0}^{T}(-A)\,e^{-j\omega t}\,\mathrm{d}t$$

Changing t by $-t$ in the first integral, we get

$$F(j\omega)=\int_{0}^{T} A\,e^{j\omega t}\,dt-\int_{0}^{T} A\,e^{-j\omega t}\,\mathrm{d}t$$

$$=A\int_{0}^{T}\left(e^{j\omega t}-e^{-j\omega t}\right)\mathrm{d}t \;=j2A\int_{0}^{T}\sin\omega t\,\mathrm{d}t=j2A\left(\frac{\cos\omega t}{-\omega}\right)_{0}^{T}$$

$$=j2A\left(\frac{\cos\omega T}{-\omega}+\frac{1}{\omega}\right)=\frac{j2A}{\omega}[1-\cos\omega T]$$

Therefore, $$F(j\omega)\big|=\frac{2A}{\omega}(1-\cos\,\omega T)$$

Therefore, the magnitude is zero when $(1-\cos\,\omega T)=0$

i.e. $$\omega T=2n\pi,\text{ or }\omega=n\frac{2\pi}{T}$$

The magnitude spectrum is shown in Fig. E2.25(b).

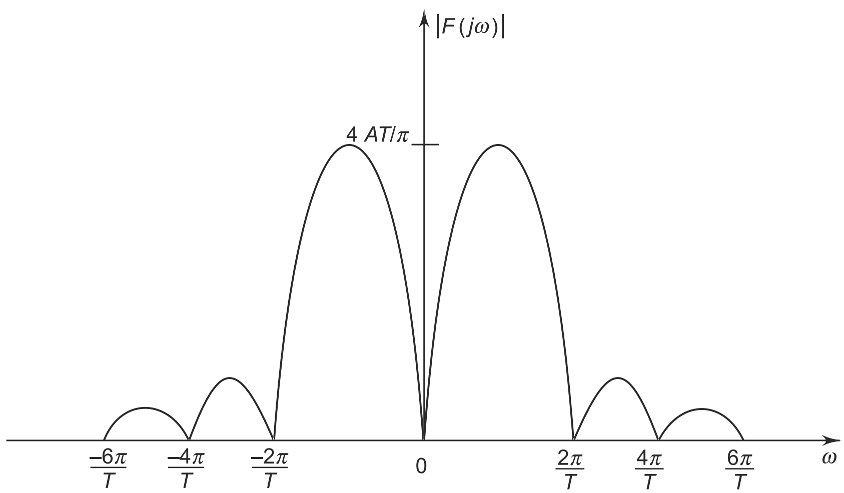


Fig. E2.25(b) *Magnitude Spectrum*

$$\tan\phi(j\omega) = \frac{\text{Imaginary part of } F(j\omega)}{\text{Real part of } F(j\omega)}$$

$$= \frac{2A/\omega(1-\cos\omega T)}{0}$$

$$= \infty, \text{ when } \omega \text{ is positive}$$

$$= -\infty, \text{ when } \omega \text{ is negative}$$

Therefore, $\phi(j\omega) = \tan^{-1}(\infty) = +\frac{\pi}{2}$, for ω positive

$$= \tan^{-1}(-\infty) = -\frac{\pi}{2}, \quad \text{for } \omega \text{ negative}$$

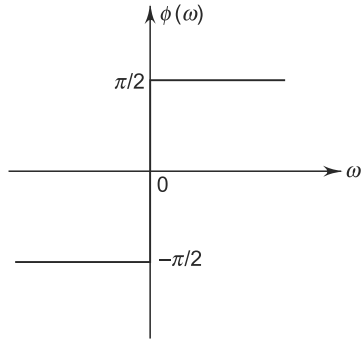


Fig. E2.25(c) *Phase Spectrum*

The phase spectrum is shown in Fig. E2.25(c).

Example 2.26 *Find the Fourier transform of the signal f(t) shown in Fig. E2.26.*

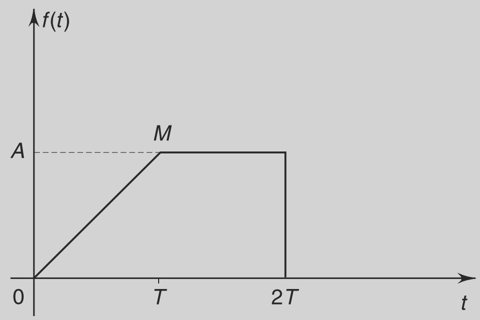


Fig. E2.26

Solution The equation of the line 0M is $\frac{y-y_1}{x-x_1} = \frac{y_1-y_2}{x_1-x_2}$

Hence, $\frac{f(t)-0}{t-0} = \frac{0-A}{0-T}$

$$f(t) = \frac{A}{T}t, \text{ for } 0 < t < T = A, \quad \text{for } T < t < 2T$$

$$F(j\omega) = \int_0^T \frac{A}{T} t e^{-j\omega t} dt + \int_T^{2T} A e^{-j\omega t} dt$$

$$= \frac{A}{T}\left[t\left\{\frac{e^{-j\omega t}}{-j\omega}\right\} - \left\{\frac{e^{-j\omega t}}{(-j\omega)^2}\right\}\right]_0^T + A\left[\frac{e^{-j\omega t}}{-j\omega}\right]_T^{2T}$$

$$= \frac{A}{T}\left[T\frac{e^{-j\omega T}}{-j\omega} - \frac{e^{-j\omega T}}{-\omega^2} - 0 + \frac{1}{-\omega^2}\right] + A\left[\frac{e^{-j2\omega T}}{-j\omega} - \frac{e^{-j\omega T}}{-j\omega}\right]$$

$$= A\left[-\frac{e^{-j\omega T}}{j\omega} + \frac{e^{-j\omega T}}{T\omega^2} - \frac{1}{T\omega^2} - \frac{e^{-j2\omega T}}{j\omega} + \frac{e^{-j\omega T}}{j\omega}\right]$$

$$= \frac{A}{T\omega^2}\left[e^{-j\omega T} - 1\right] + j\frac{A}{\omega}e^{-j2\omega T}$$

$$= \frac{A}{T\omega^2}e^{-j\omega T/2}\left[e^{-j\omega T/2} - e^{j\omega T/2}\right] + j\frac{A}{\omega}e^{-j2\omega T}$$

$$= -j\frac{2A}{T\omega^2}e^{-j\omega T/2}\sin\frac{\omega T}{2} + j\frac{A}{\omega}e^{-j2\omega T}$$

$$= -j\frac{A}{\omega}e^{-j\omega T/2}\,\text{sinc}\,\frac{\omega T}{2} + j\frac{A}{\omega}e^{-j2\omega T}$$

$$= \frac{A}{j\omega}\left[e^{-j\omega T/2}\,\text{sinc}\,\frac{\omega T}{2} - e^{-j2\omega T}\right]$$

Example 2.27 *The magnitude $|F(j\omega)|$ and the phase $\phi(\omega)$ of the Fourier transform of a signal $f(t)$ are shown in Figs. E2.27(a) and (b). Find $f(t)$.*

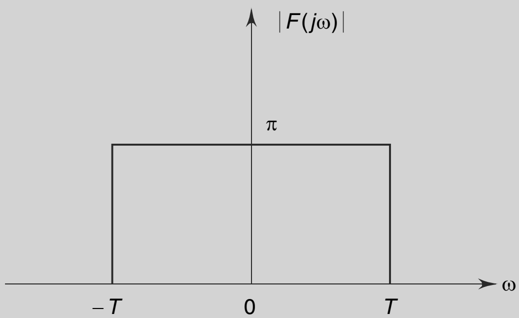


Fig. E2.27(a)

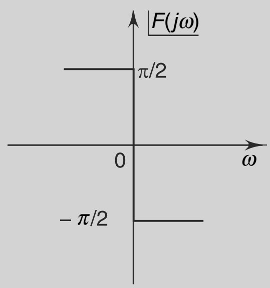


Fig. E2.27(b) *Phase Spectrum*

Solution The magnitude $|F(j\omega)| = \pi$, for $-T \le \omega \le T$

$$\phi(\omega) = \underline{|F(j\omega)} = \begin{cases} +\frac{\pi}{2}, & \text{for } \omega < 0 \\ -\frac{\pi}{2}, & \text{for } \omega > 0 \\ 0, & \text{for } \omega = 0 \end{cases}$$

For the limits $\quad -T \le \omega \le 0,\ F(j\omega) = \pi e^{j\pi/2}$

For the limits $\quad 0 \le \omega \le T,\ F(j\omega) = \pi e^{-j\pi/2}$

$$f(t)=\frac{1}{2\pi}\int_{-\infty}^{\infty}F(j\omega)e^{j\omega t}\,d\omega$$

$$=\frac{1}{2\pi}\left[\int_{-T}^{0}\pi e^{j\pi/2}e^{j\omega t}\,d\omega+\int_{0}^{T}\pi e^{-j\pi/2}e^{j\omega t}\,d\omega\right]$$

$$=\frac{1}{2}e^{j\pi/2}\left[\frac{e^{j\omega t}}{jt}\right]_{-T}^{0}+\frac{1}{2}e^{-j\pi/2}\left[\frac{e^{j\omega t}}{jt}\right]_{0}^{T}$$

$$=\frac{1}{2}\frac{e^{j\pi/2}}{jt}\left[1-e^{-jTt}\right]+\frac{1}{2}\frac{e^{-j\pi/2}}{jt}\left[e^{jTt}-1\right]$$

$$=\frac{1}{j2t}\left[e^{j\pi/2}-e^{j\pi/2}e^{-jTt}+e^{-j\pi/2}e^{jTt}-e^{-j\pi/2}\right]$$

$$=\frac{1}{t}\left[\frac{e^{j\pi/2}-e^{-j\pi/2}}{j2}\right]-\frac{1}{t}\left[\frac{e^{j(\pi/2-Tt)}-e^{-j(\pi/2-Tt)}}{j2}\right]$$

$$=\frac{1}{t}[\sin(\pi/2)-\sin(\pi/2-Tt)]$$

$$=\frac{1}{t}[1-\cos Tt]=\frac{2\sin^2\left(\frac{Tt}{2}\right)}{t}=\frac{T^2t}{2}\operatorname{sinc}^2\left(\frac{Tt}{2}\right)$$

Example 2.28 *Obtain the Fourier transform of the trapezoidal pulse shown in Fig. E2.28.*

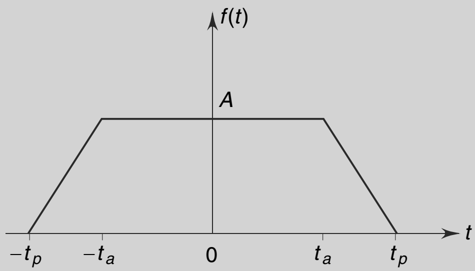


Fig. E2.28

Solution The Fourier transform of $f(t)$ is given by

$$F(j\omega)=\int_{-\infty}^{\infty}f(t)e^{-j\omega t}\,\mathrm{d}t$$

The function $f(t)$ is given by

$$f(t)=\begin{cases}\dfrac{At}{t_p-t_a}+\dfrac{At_p}{t_p-t_a}, & \text{for } -t_p<t<-t_a\\ A, & \text{for } -t_a<t<t_a\\ \dfrac{At_p}{t_p-t_a}-\dfrac{At}{t_p-t_a}, & \text{for } t_a<t<t_p\end{cases}$$

Therefore,

$$F(j\omega)=\int_{-t_p}^{-t_a}\frac{A\,te^{-j\omega t}}{t_p-t_a}\mathrm{d}t+\int_{-t_p}^{-t_a}\frac{A\,t_p e^{-j\omega t}}{t_p-t_a}\mathrm{d}t\;+\int_{-t_a}^{t_a}A\,e^{-j\omega t}\mathrm{d}t+\int_{t_a}^{t_p}\frac{A\,t_p\,e^{-j\omega t}}{t_p-t_a}\mathrm{d}t-\int_{t_a}^{t_p}\frac{A\,te^{-j\omega t}}{t_p-t_a}\mathrm{d}t$$

$$=\frac{A}{t_p-t_a}\left[\int_{-t_p}^{-t_a}te^{-j\omega t}\mathrm{d}t-\int_{t_a}^{t_p}te^{-j\omega t}\mathrm{d}t\right]+\frac{At_p}{t_p-t_a}\left[\int_{-t_p}^{-t_a}e^{-j\omega t}\mathrm{d}t+\int_{t_a}^{t_p}e^{-j\omega t}\mathrm{d}t\right]+A\int_{-t_a}^{t_a}e^{-j\omega t}\mathrm{d}t$$

$$=\frac{A}{t_p-t_a}\left[\left\{\frac{te^{-j\omega t}}{-j\omega}+\frac{e^{-j\omega t}}{\omega^2}\right\}_{-t_p}^{-t_a}-\left\{\frac{te^{-j\omega t}}{-j\omega}+\frac{e^{-j\omega t}}{\omega^2}\right\}_{t_a}^{t_p}\right]$$

$$+\frac{At_p}{t_p-t_a}\left[\left\{\frac{e^{-j\omega t}}{-j\omega}\right\}_{-t_p}^{-t_a}+\left\{\frac{e^{-j\omega t}}{-j\omega}\right\}_{t_a}^{t_p}\right]+A\left[\frac{e^{-j\omega t}}{-j\omega}\right]_{-t_a}^{t_a}$$

$$=\frac{A}{t_p-t_a}\left[\frac{-t_a e^{j\omega t_a}}{-j\omega}+\frac{e^{j\omega t_a}}{\omega^2}-\frac{t_p e^{-j\omega t_p}}{j\omega}-\frac{e^{j\omega t_p}}{\omega^2}\right]-\left[\frac{t_p e^{-j\omega t_p}}{-j\omega}+\frac{e^{-j\omega t_p}}{\omega^2}+\frac{t_a e^{-j\omega t_a}}{j\omega}-\frac{e^{-j\omega t_a}}{\omega^2}\right]$$

$$+\frac{At_p}{t_p-t_a}\left[\frac{e^{j\omega t_a}}{-j\omega}+\frac{e^{j\omega t_p}}{j\omega}-\frac{e^{-j\omega t_p}}{j\omega}+\frac{e^{-j\omega t_a}}{j\omega}\right]+A\left[\frac{e^{-j\omega t_a}}{-j\omega}+\frac{e^{-j\omega t_a}}{j\omega}\right]$$

$$=\frac{A}{t_p-t_a}\left[\frac{t_a e^{j\omega t_a}}{j\omega}+\frac{e^{j\omega t_a}}{\omega^2}-\frac{t_p e^{j\omega t_p}}{j\omega}-\frac{e^{j\omega t_p}}{\omega^2}+\frac{t_p e^{-j\omega t_p}}{j\omega}-\frac{e^{-j\omega t_p}}{\omega^2}-\frac{t_a e^{-j\omega t_a}}{j\omega}+\frac{e^{-j\omega t_a}}{\omega^2}\right]$$

$$+\frac{At_p}{t_p-t_a}\left[\frac{e^{j\omega t_a}}{-j\omega}+\frac{e^{j\omega t_p}}{j\omega}-\frac{e^{-j\omega t_p}}{j\omega}-\frac{e^{-j\omega t_a}}{j\omega}\right]+A\left[\frac{e^{-j\omega t_a}}{-j\omega}+\frac{e^{j\omega t_a}}{j\omega}\right]$$

$$=\frac{A}{t_p-t_a}\left[t_a\left\{\frac{e^{j\omega t_a}-e^{j\omega t_a}}{j\omega}\right\}+\frac{1}{\omega^2}\left\{e^{j\omega t_a}+e^{-j\omega t_a}\right\}-\frac{1}{\omega^2}\left\{e^{j\omega t_p}+e^{-j\omega t_p}\right\}\right]$$

$$-\frac{At_p}{t_p-t_a}\left[\frac{e^{j\omega t_a}-e^{-j\omega t_a}}{j\omega}\right]+A\left[\frac{e^{j\omega t_a}-e^{-j\omega t_a}}{j\omega}\right]$$

$$=\frac{A}{t_p-t_a}\left[\frac{t_a}{\omega}2\sin\omega t_a+\frac{1}{\omega^2}2\cos\omega t_a-\frac{1}{\omega^2}2\cos\omega t_p\right]-\frac{At_p}{t_p-t_a}\frac{1}{\omega}2\sin\omega t_a+\frac{A}{\omega}2\sin\omega t_a$$

$$=\frac{2A}{\omega}\sin\omega t_a\left[1-\frac{t_p}{t_p-t_a}+\frac{t_a}{t_p-t_a}\right]+\frac{2A}{\omega^2(t_p-t_a)}\left[\cos\omega t_a-\cos\omega t_p\right]$$

$$=\frac{2A}{\omega^2(t_p-t_a)}\left[\cos\omega t_a-\cos\omega t_p\right]$$

Example 2.29 *Find the Fourier transform of the signal shown in Fig. E2.29.*

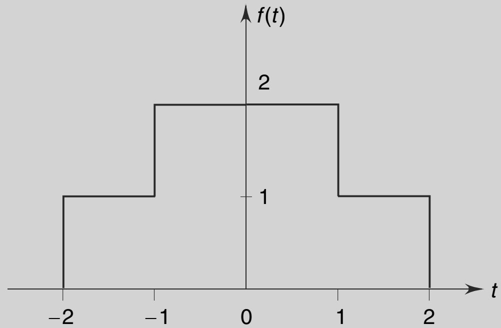


Fig. E2.28

Solution

For the given signal,
$$f(t)=\begin{cases}2, & \text{for } -1<t<1\\ 1, & \text{for } -2<t<-1\\ 1, & \text{for } 1<t<2\end{cases}$$

$$F(j\omega)=\mathcal{F}[f(t)]=\int_{-1}^{1}2.e^{-j\omega t}\,\mathrm{d}t+\int_{-2}^{-1}e^{-j\omega t}\,\mathrm{d}t+\int_{1}^{2}e^{-j\omega t}\,\mathrm{d}t$$

$$=\frac{1}{-j\omega}\left\{\left[2e^{-j\omega t}\right]_{-1}^{1}+\left[e^{-j\omega t}\right]_{-2}^{-1}+\left[e^{-j\omega t}\right]_{1}^{2}\right\}$$

$$=\frac{1}{-j\omega}\left\{2(e^{-j\omega}-e^{j\omega})\ +\ (e^{j\omega}-e^{2j\omega})\ +\ (e^{-j2\omega}-e^{-j\omega})\right\}$$

$$=\frac{1}{\omega}\left\{4\left[\frac{e^{j\omega}-e^{-j\omega}}{2j}\right]+2\left[\frac{e^{j2\omega}-e^{-j2\omega}}{2j}\right]-2\left[\frac{e^{j\omega}-e^{-j\omega}}{2j}\right]\right\}$$

$$=\frac{4}{\omega}\left\{\sin\omega+\frac{1}{2}\sin 2\omega-\frac{1}{2}\sin\omega\right\}=\frac{4}{\omega}\left\{\frac{1}{2}\sin\omega+\frac{1}{2}\sin 2\omega\right\}$$

$$=2\frac{\sin\omega}{\omega}+2\frac{\sin 2\omega}{\omega}=2\,\mathrm{sinc}\,\omega+4\,\mathrm{sinc}\,2\omega$$

Example 2.30 *Find the Fourier transform of $f(t)=e^{-at}\cos bt$.*

Solution

$$\mathcal{F}[e^{-at}\cos bt]=\frac{1}{2}\int_{0}^{\infty}e^{-at}[e^{jbt}+e^{-jbt}]e^{-j\omega t}\,\mathrm{d}t$$

$$=\frac{1}{2}\int_{0}^{\infty}\left[e^{-(a-jb+j\omega)t}+e^{-(a+jb+j\omega)t}\right]\mathrm{d}t$$

$$=\frac{1}{2}\left[\frac{e^{-(a+j\omega-jb)t}}{-(a+j\omega-jb)}+\frac{e^{-(a+j\omega-jb)t}}{-(a+j\omega+jb)}\right]_{0}^{\infty}=\frac{1}{2}\left[\frac{1}{(a+j\omega-jb)}+\frac{1}{(a+j\omega+jb)}\right]$$

$$= \frac{1}{2}\left[\frac{a + j\omega + jb + a + j\omega - jb}{(a + j\omega)^2 + b^2}\right] = \frac{a + j\omega}{(a + j\omega)^2 + b^2}$$

Example 2.31 *Find the Fourier transform of f (t) = t cos at.*

Solution

$$\mathcal{F}[t\cos at] = \mathcal{F}\left[t\left\{\frac{e^{jat} + e^{-jat}}{2}\right\}\right]$$

$$= \int_0^{\infty} t\left[\frac{e^{jat} + e^{-jat}}{2}\right]e^{-j\omega t}\mathrm{d}t = \frac{1}{2}\int_0^{\infty}(te^{-j(-a+\omega)t} + te^{-j(a+\omega)t})\mathrm{d}t$$

$$= \frac{1}{2}\left[t\left\{\frac{e^{j(-a+\omega)t}}{-j(-a+\omega)}\right\} - \frac{e^{-j(-a+\omega)t}}{[-j(-a+\omega)]^2}\right]_0^{\infty}$$

$$+\frac{1}{2}\left[t\left\{\frac{e^{-j(a+\omega)t}}{-j(a+\omega)}\right\} - \frac{e^{-j(a+\omega)t}}{[-j(a+\omega)]^2}\right]_0^{\infty}$$

$$= \frac{1}{2}\left[\frac{1}{(-j)^2(-a+\omega)^2} + \frac{1}{(-j)^2(a+\omega)^2}\right] = -\frac{1}{2}\left[\frac{1}{(-a+\omega)^2} + \frac{1}{(a+\omega)^2}\right]$$

$$= -\frac{1}{2}\frac{a^2 + \omega^2 + 2a\omega + a^2 + \omega^2 - 2a\omega}{(\omega^2 - a^2)^2} = -\frac{\omega^2 + a^2}{(\omega^2 - a^2)^2} \quad \text{or} = -\frac{a^2 + \omega^2}{(a^2 - \omega^2)^2}$$

Example 2.32 *Find the Fourier transform of f (t) = sin (ω$_c$t + θ).*

Solution

$$\mathcal{F}[f(t)] = F(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\mathrm{d}t$$

$$= \int_0^{\infty}\sin(\omega_c t + \theta)e^{-j\omega t}\mathrm{d}t = \frac{1}{j2}\int_0^{\infty}\left[e^{j(\omega_c t+\theta)} - e^{-j(\omega_c t+\theta)}\right]e^{-j\omega t}\mathrm{d}t$$

$$= \frac{1}{j2}\int_0^{\infty}\left[e^{j(\omega_c t-\omega t+\theta)} - e^{-j(\omega_c t+\omega t+\theta)}\right]\mathrm{d}t = \frac{1}{j2}\left[\frac{e^{-j(-\omega_c t+\omega t-\theta)}}{-j(-\omega_c+\omega)} - \frac{e^{-j(\omega_c t+\omega t+\theta)}}{-j(\omega_c+\omega)}\right]_0^{\infty}$$

$$= \frac{1}{2}\left[\frac{e^{-j(-\omega_c+\omega)t}e^{j\theta}}{-(\omega_c-\omega)} - \frac{e^{-j(\omega_c+\omega)t}e^{-j\theta}}{(\omega_c+\omega)}\right]_0^{\infty} = \frac{1}{2}\left[\frac{e^{j\theta}}{\omega_c - \omega} + \frac{e^{-j\theta}}{\omega_c + \omega}\right]$$

$$= \frac{1}{2}\left[\frac{e^{j\theta}(\omega_c + \omega) + e^{-j\theta}(\omega_c - \omega)}{\omega_c^2 - \omega^2}\right]$$

$$= \frac{1}{2(\omega_c^2 - \omega^2)}\left[\omega_c\left(e^{j\theta} + e^{-j\theta}\right) + \omega\left(e^{j\theta} - e^{-j\theta}\right)\right]$$

$$= \frac{1}{2(\omega_c^2 - \omega^2)}[2\omega_c \cos\theta + j2\omega\sin\theta] = \frac{\omega_c\cos\theta + j\omega\sin\theta}{\omega_c^2 - \omega^2}$$

Example 2.33 *Find Fourier transform of* $f(t) = e^{jat^2}$ *and hence find the Fourier transform of the functions* $\cos at^2$, $\sin at^2$, $e^{-b^2t^2}$ *and* e^{-bt^2}.

Solution

$$\mathcal{F}[e^{jat^2}] = \int_{-\infty}^{\infty} e^{jat^2} e^{-j\omega t}\,dt = \int^{\infty} e^{j(at^2 - \omega t)}\,dt$$

Here,
$$at^2 - \omega t = a\left(t^2 - \frac{\omega}{a}t\right)$$

$$= a\left[t^2 - \frac{2\omega}{2a}t + \frac{\omega^2}{4a^2}\right] - \frac{\omega^2}{4a}$$

$$= a\left[t - \frac{\omega}{2a}\right]^2 - \frac{\omega^2}{4a} = \left[\sqrt{a}\,t - \frac{\omega}{2\sqrt{a}}\right]^2 - \frac{\omega^2}{4a} = y^2 - \frac{\omega^2}{4a}$$

where
$$y = \sqrt{a}\,t - \frac{\omega}{2\sqrt{a}}$$

Therefore,
$$dy = \sqrt{a}\,dt \text{ and } dt = \frac{dy}{\sqrt{a}}$$

$$\mathcal{F}[e^{jat^2}] = \int_{-\infty}^{\infty} e^{j\left[y^2 - \frac{\omega^2}{4a}\right]} \frac{dy}{\sqrt{a}}$$

$$= \frac{1}{\sqrt{a}} e^{-j\left(\frac{\omega^2}{4a}\right)} \int_{-\infty}^{\infty} e^{jy^2}\,dy = \frac{2}{\sqrt{a}} e^{-j\left(\frac{\omega^2}{4a}\right)} \int_{0}^{\infty} e^{jy^2}\,dy$$

Now,
$$r(a) = \int_0^{\infty} x^{(a-1)} e^{-x}\,dx$$

$$r\left(\frac{1}{2}\right) = \int_0^{\infty} x^{-1/2} e^{-x}\,dx$$

Putting $x = y^2$ in the above equation, we get

$$r\left(\frac{1}{2}\right) = \int_0^{\infty} y^{-1} e^{-y^2}\, 2y\,dy = 2\int_0^{\infty} e^{-y^2}\,dy = \sqrt{\pi}$$

Replacing y by $\sqrt{-jx}$ in the above integral, we get

$$2\int_0^\infty e^{jx^2}\sqrt{(-j)}dx = \sqrt{\pi}$$

$$2\int_0^\infty e^{jx^2}dx = \sqrt{\frac{\pi}{-j}} = \sqrt{j\pi}$$

$$\mathcal{F}[e^{jat^2}] = \sqrt{\frac{j\pi}{a}}\; e^{\frac{-j\omega^2}{4a}} \tag{1}$$

But
$$j = \cos\frac{\pi}{2} + j\sin\frac{\pi}{2} = e^{j\frac{\pi}{2}} \text{ and } \sqrt{j} = e^{j\frac{\pi}{4}}$$

$$\mathcal{F}[e^{jat^2}] = \sqrt{\frac{\pi}{a}}e^{j\left[\frac{\pi}{4}-\frac{\omega^2}{4a}\right]} \tag{2}$$

From Eq. (2), we can derive Fourier's transform of some trigonometric functions.

$$\mathcal{F}[e^{jat^2}] = \int_{-\infty}^{\infty}(\cos at^2 + j\sin at^2)e^{-j\omega t}\mathrm{d}t$$

$$= \sqrt{\frac{\pi}{a}}\left[\cos\left(\frac{\pi}{4}-\frac{\omega^2}{4a}\right) + j\sin\left(\frac{\pi}{4}-\frac{\omega^2}{4a}\right)\right]$$

Equating real and imaginary parts from both the sides,

$$\mathcal{F}[\cos at^2] = \int_{-\infty}^{\infty}\cos at^2 e^{-j\omega t}dt = \sqrt{\frac{\pi}{a}}\cos\left(\frac{\pi}{4}-\frac{\omega^2}{4a}\right)$$

$$\mathcal{F}[\sin at^2] = \int_{-\infty}^{\infty}\sin at^2 e^{-j\omega t}dt = \sqrt{\frac{\pi}{a}}\sin\left(\frac{\pi}{4}-\frac{\omega^2}{4a}\right)$$

Putting $a = jb^2$ in Eq. (1), we get

$$\mathcal{F}\left[e^{-b^2t^2}\right] = \sqrt{\frac{j\pi}{jb^2}}e^{-j\frac{\omega^2}{4jb^2}} = \frac{\sqrt{\pi}}{b}e^{-\left(\frac{\omega}{2b}\right)^2}$$

Therefore,
$$\mathcal{F}[e^{-b^2t^2}] = \frac{\sqrt{\pi}}{b}e^{-\left(\frac{\omega}{2b}\right)^2}$$

Putting $a = jb$ in Eq. (1), we get

$$\mathcal{F}[e^{-bt^2}] = \sqrt{\frac{j\pi}{jb}}e^{-\omega^2/4b}$$

Therefore,
$$\mathcal{F}[e^{-bt^2}] = \sqrt{\frac{\pi}{b}}e^{-\omega^2/4b}$$

Example 2.34 *Determine the Fourier transform of the sinusoidal pulse shown in Fig. E2.34.*

Solution $$F(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,dt$$

Here $f(t) = \begin{cases} A\sin\omega_0 t & \text{for } 0 < t \le T/2, \quad \text{where } \omega_0 = 2\pi/T \\ 0, & \text{otherwise} \end{cases}$

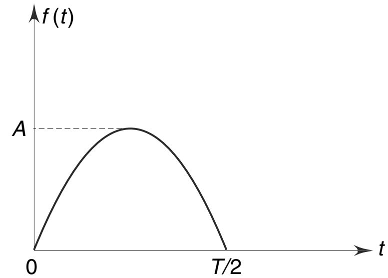


Fig. E2.34

$$F(j\omega) = \int_0^{T/2} A\sin\omega_0 t\, e^{-j\omega t}\,dt$$

$$= A\int_0^{T/2}\left(\frac{e^{j\omega_0 t} - e^{-j\omega_0 t}}{2j}\right)e^{-j\omega t}\,dt$$

$$= \frac{A}{2j}\int_0^{T/2}\left[e^{j(\omega_0-\omega)t} - e^{-j(\omega_0-\omega)t}\right]dt$$

$$= \frac{A}{2j}\left[\frac{e^{j(\omega_0-\omega)t}}{j(\omega_0-\omega)} - \frac{e^{-j(\omega_0-\omega)t}}{-j(\omega_0-\omega)}\right]_0^{T/2}$$

$$= -\frac{A}{2}\left[\frac{e^{j(\omega_0-\omega)T/2}-1}{(\omega_0-\omega)} + \frac{e^{-j(\omega_0+\omega)T/2}-1}{(\omega_0+\omega)}\right]$$

$$= -\frac{A}{2}\left[\frac{(e^{j(\omega_0-\omega)T/2}-1)(\omega_0+\omega) + (e^{-j(\omega_0-\omega)T/2}-1)(\omega_0+\omega)}{(\omega_0^2-\omega^2)}\right]$$

$$= \frac{-A}{2(\omega_0^2-\omega^2)}\left[\omega_0(e^{j(\omega_0-\omega)T/2} + e^{-j(\omega_0-\omega)T/2}) + \omega(e^{j(\omega_0-\omega)T/2} - e^{-j(\omega_0-\omega)T/2}) - 2\omega_0\right]$$

$$= \frac{-A}{2(\omega_0^2-\omega^2)}\left[\omega_0 e^{-j\omega T/2}(e^{j\pi}+e^{-j\pi}) + \omega e^{-j\omega T/2}(e^{j\pi}-e^{-j\pi}) - 2\omega_0\right]$$

$$= \frac{-A}{2(\omega_0^2-\omega^2)}\left[-e^{-j\omega T/2}2\omega_0 - 2\omega_0\right] = \frac{2A\omega_0}{2(\omega_0^2-\omega^2)}\left[1+e^{-j\omega T/2}\right]$$

$$= \frac{A\omega_0}{\omega_0^2-\omega^2}[e^{j\omega T/4} + e^{-j\omega T/4}]e^{-j\omega T/4} = \left(\frac{2A\omega_0}{\omega_0^2-\omega^2}\cos\frac{\omega T}{4}\right)e^{-j\omega T/4}$$

REVIEW QUESTIONS

2.1 Write down the trigonometric form of the Fourier series representation of a periodic signal.
2.2 What do you understand by odd and even functions?
2.3 State the necessary and sufficient conditions for the existence of the Fourier series representation for a signal.

2.4 Write a short note on Dirichlet's conditions.
2.5 Explain the different forms of Fourier series.
2.6 Obtain the trigonometric Fourier series of the waveform given in Fig. Q2.6.

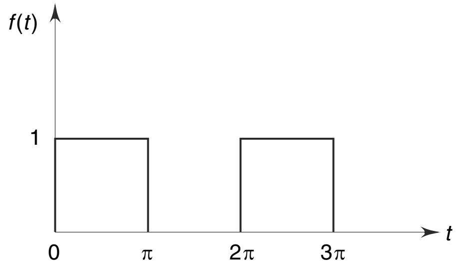


Fig. Q2.6

Ans: $f(t) = \frac{1}{2} + \frac{2}{\pi}\sum_{n=1}^{\infty}\frac{\sin(2n-1)t}{(2n-1)}$

2.7 Obtain the trigonometric Fourier series for the waveform shown in Fig. Q2.7.

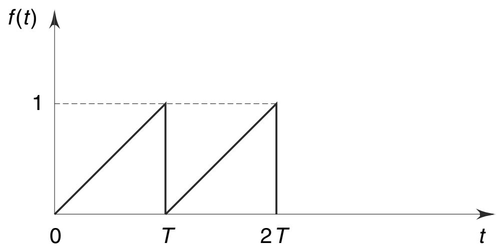


Fig. Q2.7

Ans: $f(t) = \frac{1}{2} + \frac{1}{\pi}\sum_{n=1}^{\infty}\frac{\sin(n\omega t)}{n}$

2.8 Obtain the trigonometric Fourier series for the signal shown in Fig. Q2.8.

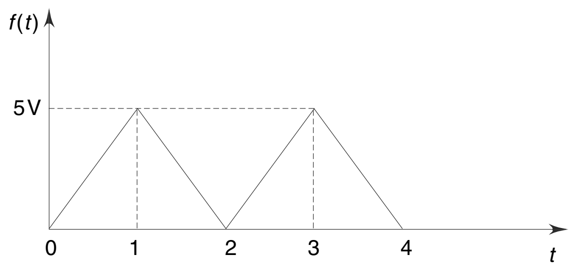


Fig. Q2.8

Hint: $f(t) = \begin{cases} 5t, & for \quad 0 < t < 1 \\ -5(t-2), & for \quad 1 < t < 2 \end{cases}$

Ans: $$f(t)=\frac{5}{2}-\frac{20}{\pi^2}\sum_{n=1}^{\infty}\frac{\cos(2n-1)\pi t}{(2n-1)^2}$$

2.9 Obtain the trigonometric Fourier series for the signal shown in Fig. Q2.9.

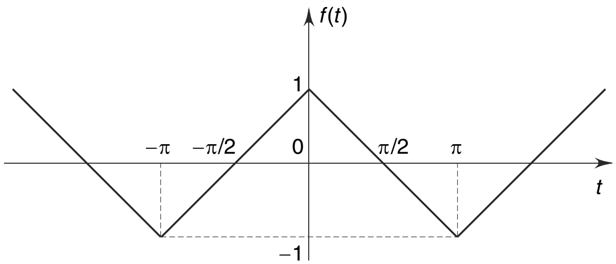


Fig. Q2.9

Hint: $$f(t)=\begin{cases}\left(1+\dfrac{2t}{\pi}\right), & \text{for } -\pi<t<0\\ \left(1-\dfrac{2t}{\pi}\right), & \text{for } 0<t<\pi\end{cases}$$

Ans: $$f(t)=\frac{8}{\pi^2}\sum_{n=1}^{\infty}\frac{\cos(2n-1)t}{(2n-1)^2}$$

2.10 Obtain the trigonometric Fourier series for the full wave rectified sine wave shown in Fig. Q2.10.

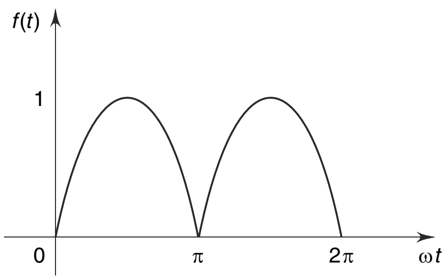


Fig. Q2.10

Hint: $f(t)=\sin t$, for $0<t<\pi$

Ans: $$f(t)=\frac{2}{\pi}-\frac{4}{\pi}\sum_{n=1}^{\infty}\frac{\cos 2nt}{4n^2-1}$$

2.11 Obtain the trigonometric Fourier series for the waveform shown in Fig. Q2.11.

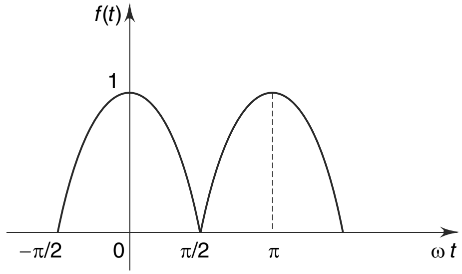


Fig. Q2.11

Hint: $f(t) = \cos t$, for $-\pi/2 < t < \pi/2$

Ans: $f(t) = \frac{2}{\pi} + \frac{4}{\pi}\sum_{n=1}^{\infty}(-1)^{n+1}\frac{1}{4n^2-1}\cos 2n\ t$

2.12 Obtain the trigonometric Fourier series for the half wave rectified sine waveform shown in Fig. Q2.12.

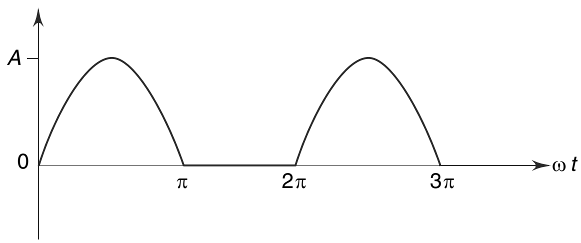


Fig. Q2.12

Ans: $f(t) = \frac{A}{\pi}\left(1 + \frac{\pi}{2}\sin\omega t - \frac{2}{3}\cos 2\omega t - \frac{2}{15}\cos 4\omega t - \frac{2}{35}\cos 6\omega t - \ldots\right)$

2.13 Obtain the Fourier series for the full wave rectified sine waves shown in Fig. Q2.13 (a) and (b).

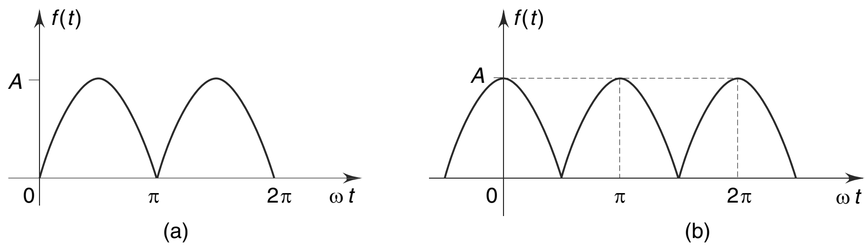


Fig. Q2.13

Ans: (a) $f(t) = \frac{2A}{\pi}\left(1 + \frac{2}{3}\cos 2\omega t - \frac{2}{15}\cos 4\ \omega t - \frac{2}{35}\cos 6\omega t - \ldots\right)$

(b) $f(t) = \frac{2A}{\pi}\left(1 + \frac{2}{3}\cos 2\omega t - \frac{2}{15}\cos 4\ \omega t + \frac{2}{35}\cos 6\omega t - \ldots\right)$

2.14 Obtain the Trigonometric Fourier series expansion of the periodic signal

$$x(t) = \begin{cases} A & \text{for} \quad kT < t \le (k+1)T \\ -A & \text{for} \quad (k+1)T < t \le -kT \end{cases}$$

with k taking the values 0, 2, 4, 6, …

2.15 A periodic triangular waveform starts at the origin with zero value and increases linearly with respect to time. After a time T, it becomes zero. Obtain its Fourier series.

2.16 With regard to Fourier series representation, justify the following statement:

(i) Odd functions have only sine terms

(ii) Even functions have no sine terms

(iii) Functions with half-wave symmetry have only odd harmonics.

2.17 Obtain the exponential Fourier series for the waveforms shown in Figs. Q2.17(a) and (b).

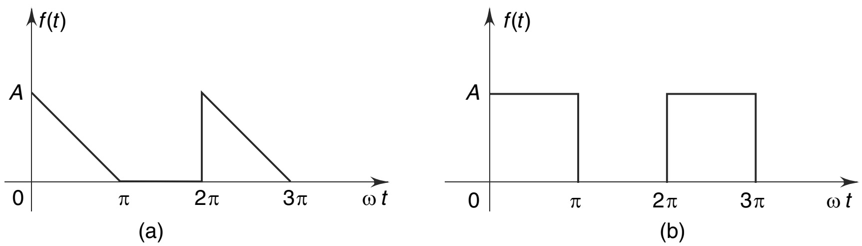


Fig. Q2.17

$$\textbf{Ans}: (a) f(t) = A\left\{\cdots + \left(\frac{1}{9\pi^2} + j\frac{1}{6\pi}\right)e^{-3\omega t} + j\frac{1}{4\pi}e^{-j2\omega t} + \left(\frac{1}{\pi^2} + j\frac{1}{2\pi}\right)e^{-j\omega t} + \frac{1}{4} + \left(\frac{1}{\pi^2} - j\frac{1}{2\pi}\right)e^{j\omega t} - j\frac{1}{4\pi}e^{j2\omega t} + \left(\frac{1}{9\pi^2} - j\frac{1}{6\pi}\right)e^{j\omega t} + \cdots\right\}$$

$$(b)\ f(t) = A\left\{\cdots + j\frac{1}{3\pi}e^{-j3\omega t} + j\frac{1}{\pi}e^{-j3\omega t} + \frac{1}{2} - j\frac{1}{\pi}e^{j\omega t} - j\frac{1}{3\pi}e^{j3\omega t} - \ldots\right\}$$

2.18 Expand the following function over the interval (–4, 4) by a complex Fourier series

$$f(t) = 1, \ -2 \le t \le 2$$
$$= 0, \text{ elsewhere}$$

2.19 Determine the complex exponential Fourier series for the periodic signal

$$f(t) = A\,|\sin 2\pi / T|$$

2.20 For the periodic signal $f(t)$ shown in Fig. Q2.20, show that the complex exponential Fourier series coefficients are given by

$$c_n = \begin{cases} 2/n^2\pi^2, & \text{for} \quad n \quad \text{odd} \\ 0, & \text{for} \quad n \quad \text{even and} \ne 0 \end{cases}$$

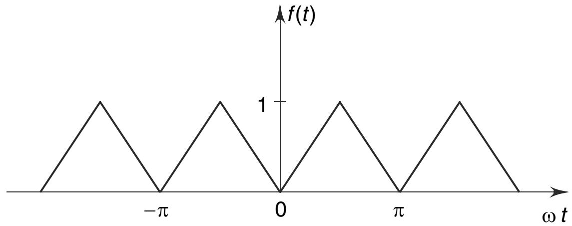


Fig. Q2.20

2.21 State and prove Parseval's theorem for complex exponential Fourier series.

2.22 Distinguish between the exponential form of Fourier series and Fourier transform. What is the nature of the "transform pairs" in the above two cases?

2.23 How would you obtain Fourier integral from Fourier series?

2.24 Define Fourier transform pair.

2.25 Define the Fourier transform of a time function and explain under what conditions it exists.

2.26 Explain how non-periodic signals can be represented by Fourier transform.

2.27 State and explain Parseval's theorem for Fourier transform.

2.28 What is the Fourier transform of a non-periodic signal $f(t)$ and explain the same with an example.

2.29 Explain in detail with suitable examples the various properties of Fourier transform.

2.30 Describe the symmetry property of Fourier transform.

2.31 State and prove that convolution theorem in relation of Fourier transform.

2.32 A signal with the spectrum $X(j\omega) = \dfrac{1}{1+0.5j\omega}$ is applied to the input of a first order high-pass filter having the frequency response $H(j\omega) = \dfrac{j\omega}{1+j\omega}$. Sketch the magnitude and phase of the output signal spectrum over the range $0<\omega<10$, using linear scales.

2.33 Determine Fourier transform of the signal shown in Fig. Q2.33.

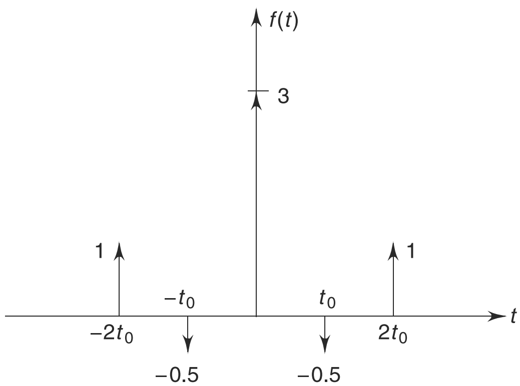


Fig. Q2.33

Ans: $F(j\omega) = 3 - \cos \omega t_0 + 2 \cos 2\omega t_0$

2.34 Find the Fourier transform of a gate pulse of unit height, unit width and centered $t = 0$.

2.35 Determine the inverse Fourier transform of the spectrum shown in Fig. Q2.35.

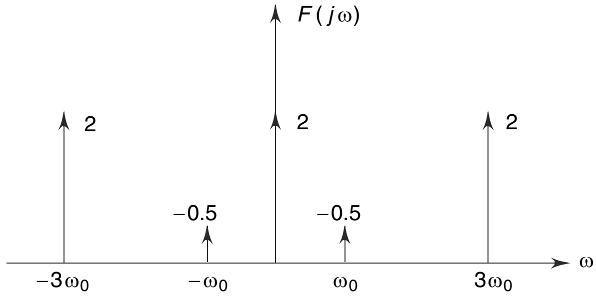


Fig. Q2.35

Ans: $f(t) = \frac{1}{2\pi}\{2 + \cos\omega_0 t + 4\cos 3\omega_0 t\}$

2.36 Find the Fourier transform of the time function

$$f(t) = 5[u(t+3) + u(t+2) - u(t-2) - u(t-3)]$$

Ans : $\frac{10}{\omega}(\sin 3\omega + \sin 2\omega)$

2.37 A certain function of time, $f(t)$ has a Fourier transform

$$F(j\omega) = \frac{1}{(\omega^2+1)} e^{2\omega^2/(\omega^2+1)}$$

Write down the Fourier transform of

(i) $f(2t)$, (ii) $f(t-2)e^{jt}$ and (iii) $3\frac{df(t)}{dt}$

2.38 Obtain the Fourier transform of the signal

$$f(t) = \frac{1}{2}\left[\delta(t+1) + \delta\left(t+\frac{1}{2}\right) + \delta\left(t-\frac{1}{2}\right) + \delta(t-1)\right]$$

Ans : $F(j\omega) = \cos\omega + \cos\omega/2$

2.39 Obtain the Fourier transform of the signal $f(t) = e^{-a|t|}\cos\omega_0 t$.

2.40 What is the difference between energy signal and power signal?

2.41 Determine the energy of the signal $x(t) = 10\left(\frac{\sin 10\pi t}{10\pi t}\right)$.

2.42 Find the energy signal of the signal $x(t) = e^{(-\alpha t)}u(t)$, contained in frequencies $|f|, \alpha/2\pi$. What percentage of the total energy is this?
Note : α is positive.

2.43 Determine the Fourier transform of the following signal and also explain what will happen to the spectrum if τ decreases but $A\,\tau$ remains constant.

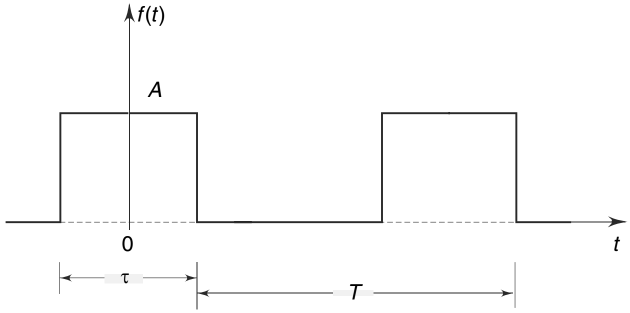


Fig. Q2.43

2.44 Defining $x(t)$ and $y(t)$ as

$$x(t) = \begin{cases} e^{-1} \text{ for } t \geq 0 \\ \quad\quad\quad \text{and} \\ 0 \quad \text{for } t < 0 \end{cases}$$

$$y(t) = \begin{cases} \alpha e^{-at} \text{ for } t \geq 0 \\ 0 \quad\quad \text{for } t < 0 \end{cases}$$

Find $f(t) = x(t)*y(t)$.

2.45 With suitable waveform, explain the convolution of $x(t)$, the input and $h(t)$, the system transfer function to get $y(t)$, the output.

2.46 Evaluate the Fourier transform of a signal unit pulse of 1 volt.

2.47 Obtain the Fourier transform of a signal symmetrical triangular pulse.

2.48 Show that a time shift in the time domain is equal to a phase shift in the frequency domain.

2.49 Find the Fourier transform of $x(t) = Ae^{-t/T}\, u(t)$ and sketch its magnitude and phase as functions of frequency.

2.50 Determine the Fourier transform of a two-sided exponential pulse $x(t) = e^{-|t|}$.

2.51 Find the Fourier transform of $x(t) = A \cos(\omega_c t + \theta)$.

2.52 Find the Fourier transform of $f(t) = e^{-at} \sin bt$.

2.53 Find the Fourier transform of the following functions

(a) $5 \sin^2 3t$ (b) $\cos(8t + 0.1\pi)$

Ans: (a) $2.5\pi\,[2\delta(\omega) + \delta(\omega + 6) + \delta(\omega - 6)]$

(b) $18.85 \angle 18^\circ\, \delta(\omega - 8) + 18.85 \angle -18^\circ\, \delta(\omega + 8)$

2.54 Find the Fourier transform of the single triangular pulse with period $T = 8$ sec and amplitude $A = 10$V.

Ans: $40 \text{ sinc}^2 2\omega$

2.55 Determine the Fourier transform of a one-cycle sine wave.

Chapter 3 Applications of Laplace Transform to System Analysis

INTRODUCTION 3.1

The Laplace transform is closely related to the Fourier transform and the z-transform. An important difference between the Fourier transform and the Laplace transform is that the Fourier transform uses a summation of waves of positive and negative frequencies, while the Laplace transform employs damped waves through the use of an additional factor $e^{-\sigma}$ where σ is a positive number. The Fourier transform and the Laplace transform are operations that convert the time domain function $f(t)$ to the frequency domain function $F(j\omega)$ and $F(s)$ respectively.

The relation between the output signal $y(t)$ and the input signal $x(t)$ in a *RLC* circuit is a linear differential equation of the form given by

$$\sum_{k=1}^{N} a_k \frac{\mathrm{d}^k y(t)}{\mathrm{d}t^k} = \sum_{k=0}^{M} b_k \frac{\mathrm{d}^k x(t)}{\mathrm{d}t^k}$$

The Laplace transform gives the total solution to the differential equation and corresponding intial and final value problems.

Laplace transform is an important and powerful tool in system analysis and design. This transform is widely used for describing continuous circuits and systems, including automatic control systems and also for analysing signal flow through causal linear time invariant systems with non-zero initial conditions. The z-transform, to be discussed in the next chapter, is suitable for dealing with discrete signals and systems. We can conclude the Laplace transform and the z-transform are complementary to the Fourier transform.

DEFINITION 3.2

For a periodic or non-periodic time function $f(t)$, which is zero for $t \leq 0$ and defined for $t > 0$, the Laplace transform of $f(t)$, denoted as $\mathcal{L}\{f(t)\}$, is defined by

$$\mathcal{L}\{f(t)\} = F(s) = \int_0^{\infty} f(t) e^{-(\sigma + j\omega)t} \mid \mathrm{d}t$$

Putting $s = \sigma + j\omega$, we have

$$F(s) = \int_0^{\infty} f(t) e^{-st} t \qquad (3.1)$$

The condition for the Laplace transform to exist is $\int_{-\infty}^{\infty} |f(t)e^{-st}|\,dt < \infty$ for some finite σ. Laplace transform thus converts the time domain function *f*(*t*) to the frequency domain function *F*(*s*). This transform defined to the positive-time functions is called single-sided or unilateral, because it does not depend on the history of *x*(*t*) prior to *t* = 0. In the double-sided or bilateral Laplace transform, the lower limit of integration is *t* = −∞, where *x*(*t*) covers over all time.

Due to the convergence factor $e^{-\sigma t}$, the ramp, parabolic functions, etc. are Laplace transformable. In transient problems, the Laplace transform is preferred to the Fourier transform, as the Laplace transform directly takes into account the initial conditions at *t* = 0, due to the lower limit of integration.

Inverse Laplace Transform

The inverse Laplace transform is used to convert a frequency domain function *F*(*s*) to the time domain function *f*(*t*), as defined by

$$f(t) = \mathcal{L}^{-1}\{F(s)\} = \frac{1}{2\pi j}\int_{\sigma-j\omega}^{\sigma+j\omega} F(s)e^{st}\,ds \tag{3.2}$$

Here, the path of integration is a straight line parallel to the *jω*-axis, such that all the poles of *F*(*s*) lie to the left of the line.

In practice, it is not necessary to carry out this complicated integration. With the help of partial fractions and existing tables of transform pairs, we can obtain the solution of different expressions.

REGION OF CONVERGENCE (ROC) 3.3

For the existence of the Laplace transform, the integral $F(s) = \int_0^{\infty} f(t)e^{-st}\,dt$ must be converging. The limits of the variable $s = \sigma + j\omega$ to a part of the complex plane is called the *Region of Convergence* (ROC). For example, $F(s) = \mathcal{L}\{e^{3t}\} = \frac{1}{s-3}$. Here, the Laplace transform is defined only for $R_e(s) > 3$. The region $R_e(s) > 3$, i.e. $\sigma > 3$ is called the Region of Convergence, which is shown in Fig. 3.1. ROC is required in computing the Laplace transform and inverse Laplace transform. If the ROC is not specified, the inverse Laplace transform is not unique.

In the one-sided Laplace transform, all time functions are assumed to be positive and there is a one-to-one correspondence between the Laplace transform and its inverse. Hence, no ambiguity will arise, even if the ROC is not specified in the one-sided Laplace transform. But, in the two-sided Laplace transform, the specification of ROC is essential.

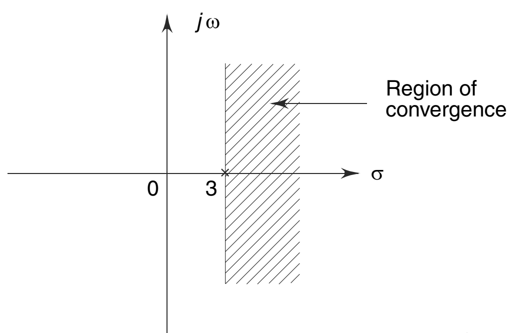


Fig. 3.1 *Region of Convergence for* $F(s) = \frac{1}{(s-3)}$

3.4 LAPLACE TRANSFORMS OF SOME IMPORTANT FUNCTIONS

1. Unit Step Function

$$f(t) = 1,\ 0 < t < \infty$$

$$\mathcal{L}\{u(t)\} = \int_0^\infty 1\cdot e^{-st}\mathrm{d}t = -\frac{1}{s}[e^{-st}]_0^\infty = -\frac{1}{s}[0-1] = \frac{1}{s} \tag{3.3}$$

2. Exponential Function

$$f(t) = Ae^{-at}$$

$$\mathcal{L}\{Ae^{-at}\} = \int_0^\infty Ae^{-at}e^{-st}\mathrm{d}t = A\int_0^\infty e^{-(a+s)t}\ \mathrm{d}t = \frac{A}{a+s}\left[e^{-(a+s)t}\right]_0^\infty = \frac{A}{(s+a)}$$

Hence, $$\mathcal{L}\{Ae^{-at}\} = \frac{A}{(s+a)} \tag{3.4}$$

3. Sine Function

$$f(t) = \sin\omega_0 t$$

Using Euler's identity, we have

$$\sin\omega_0 t = \frac{1}{2j}(e^{j\omega_0 t} - e^{-j\omega_0 t})$$

Hence, $$\mathcal{L}\{\sin\omega_0 t\} = \frac{1}{2j}\left[\mathcal{L}(e^{j\omega_0 t}) - \mathcal{L}(e^{-j\omega_0 t})\right] = \frac{1}{2j}\left[\frac{1}{s-j\omega_0} - \frac{1}{s+j\omega_0}\right] = \frac{\omega_0}{s^2+\omega_0^2}$$

Hence, $$\mathcal{L}\{\sin\omega_0 t\} = \frac{\omega_0}{s^2+\omega_0^2} \tag{3.5}$$

4. Cosine Function

$$f(t) = \cos\omega_0 t$$

We know that $\cos\omega_0 t = \frac{1}{2}(e^{j\omega_0 t} - e^{-j\omega_0 t})$

$$\mathcal{L}\{\cos\omega_0 t\} = \frac{1}{2}\left[\mathcal{L}(e^{j\omega_0 t}) + \mathcal{L}(e^{-j\omega_0 t})\right] = \frac{1}{2}\left[\frac{1}{s-j\omega_0} + \frac{1}{s+j\omega_0}\right] = \frac{s}{s^2+\omega_0^2}$$

Hence, $$\mathcal{L}\{\sin\omega_0 t\} = \frac{s}{s^2+\omega_0^2} \tag{3.6}$$

5. Hyperbolic Sine and Cosine Functions

$$\sinh\omega_0 t = \frac{1}{2}\left[e^{\omega_0 t} - e^{-\omega_0 t}\right]$$

$$\cosh\omega_0 t = \frac{1}{2}\left[e^{\omega_0 t} + e^{-\omega_0 t}\right]$$

$$\mathcal{L}\{\sinh \omega_0 t\} = \frac{1}{2}\left[\mathcal{L}(e^{\omega_0 t}) - \mathcal{L}(e^{-\omega_0 t})\right] = \frac{1}{2}\left[\frac{1}{s-\omega_0} - \frac{1}{s+\omega_0}\right] = \frac{\omega_0}{s^2 - \omega_0^2}$$

$$\mathcal{L}\{\sinh \omega_0 t\} = \frac{\omega_0}{s^2 - \omega_0^2} \tag{3.7}$$

Similarly, $\mathcal{L}\{\cosh \omega_0 t\} = \frac{1}{2}\left[\mathcal{L}\left(e^{\omega_0 t}\right) + \mathcal{L}\left(e^{-\omega_0 t}\right)\right] = \frac{1}{2}\left[\frac{1}{s-\omega_0} + \frac{1}{s+\omega_0}\right] = \frac{s}{s^2 - \omega_0^2}$

$$\mathcal{L}\{\sinh \omega_0 t\} = \frac{\omega_0}{s^2 - \omega_0^2} \tag{3.8}$$

6. Damped Sine and Cosine Functions

$$\mathcal{L}\{e^{-at} \sin \omega_0 t\} = \mathcal{L}\left\{e^{-at}\left(\frac{e^{j\omega_0 t} - e^{-j\omega_0 t}}{2j}\right)\right\}$$

$$= \frac{1}{2j}\left[\mathcal{L}\left\{e^{-(a-j\omega_0)t} - e^{-(a+j\omega_0)t}\right\}\right]$$

$$= \frac{1}{2j}\left[\frac{1}{s+(a-j\omega_0)} - \frac{1}{s+(a+j\omega_0)}\right]$$

$$= \frac{1}{2j}\left[\frac{1}{(s+a)-j\omega_0} - \frac{1}{(s+a)+j\omega_0}\right] = \frac{\omega_0}{(s+a)^2 + \omega_0^2}$$

Hence, $$\mathcal{L}\{e^{-at} \sin \omega_0 t\} = \frac{\omega_0}{(s+a)^2 + \omega_0^2} \tag{3.9}$$

Similarly, $$\mathcal{L}\{e^{-at} \cos \omega_0 t\} = \mathcal{L}\left\{e^{-at}\left(\frac{e^{j\omega_0 t} + e^{-j\omega_0 t}}{2}\right)\right\} = \frac{s+a}{(s+a)^2 + \omega_0^2}$$

Hence, $$\mathcal{L}\{e^{-at} \cos \omega_0 t\} = \frac{s+a}{(s+a)^2 + \omega_0^2} \tag{3.10}$$

7. Damped Hyperbolic Sine and Cosine Functions

$$\mathcal{L}\{e^{-at} \sinh \omega_0 t\} = \mathcal{L}\left\{e^{-at}\left(\frac{e^{\omega_0 t} - e^{-\omega_0 t}}{2}\right)\right\}$$

$$= \frac{1}{2}\left[\mathcal{L}(e^{-(a-\omega_0)t}) - \mathcal{L}(e^{-(a+\omega_0)t})\right] = \frac{1}{2}\left[\frac{1}{s+a-\omega_0} - \frac{1}{s+a+\omega_0}\right]$$

Hence, $$\mathcal{L}\{e^{-at} \sinh \omega_0 t\} = \frac{\omega_0}{(s+a)^2 - \omega_0^2} \tag{3.11}$$

Similarly, $$\mathcal{L}\{e^{-at} \cosh \omega_0 t\} = \frac{s+a}{(s+a)^2 - \omega_0^2} \tag{3.12}$$

8. t^n Function

$$\mathcal{L}\{t^n\} = \int_0^\infty t^n e^{-st}\,\mathrm{d}t = \int_0^\infty t^n \,\mathrm{d}\left(\frac{e^{-st}}{-s}\right) = \left[\frac{t^n e^{-st}}{-s}\right]_0^\infty - \int_0^\infty \frac{e^{-st}}{-s} n t^{n-1}\,\mathrm{d}t$$

$$= \frac{n}{s}\int_0^\infty e^{-st} t^{n-1}\,\mathrm{d}t = \frac{n}{s}\mathcal{L}(t^{n-1})$$

Similarly, $\mathcal{L}\{t^{n-1}\} = \dfrac{n-1}{s}\mathcal{L}(t^{n-2})$

By taking Laplace transformations of t^{n-2}, t^{n-3}, … and substituting in the above equation, we get

$$\mathcal{L}\{t^n\} = \frac{n}{s}\frac{n-1}{s}\frac{n-2}{s}\cdots\frac{2}{s}\frac{1}{s}\mathcal{L}(t^{n-n)})$$

$$= \frac{n!}{s^n}\mathcal{L}(t^0) = \frac{n!}{s^n}\times\frac{1}{s} = \frac{n!}{s^{n+1}}, \quad \text{when } n \text{ is a positive integer.}$$

Therefore, $\mathcal{L}\{t^n\} = \dfrac{n!}{s^{n+1}}$ (3.13)

Substituting $n = 1$, we have $\mathcal{L}\{t^n\} = \dfrac{1}{s^2}$

Example 3.1 *Find the Laplace transform of the following functions*

(a) $f(t) = t^3 + 3t^2 - 6t + 4$ (b) $f(t) = \cos^3 3t$

(c) $f(t) = \sin at \cos bt$ (d) $f(t) = t \sin at$

(e) $f(t) = \dfrac{1-e^t}{t}$ (f) $f(t) = \delta(t^2 - 3t + 2)$

Solution

(a) $\mathcal{L}\{f(t)\} = \mathcal{L}[t^3 + 3t^2 - 6t + 4] = \dfrac{3!}{s^4} + 3\dfrac{2!}{s^3} - 6\dfrac{1!}{s^2} + \dfrac{4}{s} = \dfrac{6}{s^4} + \dfrac{6}{s^3} - \dfrac{6}{s^2} + \dfrac{4}{s}$

(b) $f(t) = \cos^3 3t$

We know that $\cos 3A = 4\cos^3 A - 3\cos A$

Therefore, $\mathcal{L}\{\cos^3 3t\} = \mathcal{L}\left[\dfrac{\cos 9t + 3\cos 3t}{4}\right] = \dfrac{1}{4}\left[\dfrac{s}{s^2+9^2} + 3\dfrac{s}{s^2+3^2}\right] = \dfrac{1}{4}\left[\dfrac{s}{s^2+81} + \dfrac{3s}{s^2+9}\right]$

(c) $\mathcal{L}\{\sin at \cos bt\} = \mathcal{L}\left[\dfrac{1}{2}\{\sin(a+b)t + \sin(a-b)t\}\right] = \dfrac{1}{2}\left[\dfrac{a+b}{s^2+(a+b)^2} + \dfrac{a-b}{s^2+(a-b)^2}\right]$

(d) $\mathcal{L}\{t\sin at\} = -\dfrac{\mathrm{d}}{\mathrm{d}s}\mathcal{L}[\sin at] = -\dfrac{\mathrm{d}}{\mathrm{d}s}\left[\dfrac{a}{s^2+a^2}\right] = -a\dfrac{\mathrm{d}}{\mathrm{d}s}[(s^2+a^2)^{-1}]$

$$= -a\left[-\frac{1}{(s^2+a^2)^2}\cdot 2s\right] = \frac{2as}{(s^2+a^2)^2}$$

(e)
$$f(t)=\left[\frac{1-e^t}{t}\right]$$

Here, $$\mathcal{L}\{1-e^t\}=\frac{1}{s}-\frac{1}{(s-1)}$$

$$\mathcal{L}\left\{\frac{1-e^t}{t}\right\}=\int_s^{\infty}\left[\frac{1}{s}-\frac{1}{(s-1)}\right]\mathrm{d}s=\left[\log s-\log(s-1)\right]_s^{\infty}=\left[\log\frac{s}{s-1}\right]_s^{\infty}$$
$$=\left[-\log\frac{s-1}{s}\right]_s^{\infty}=\left[-\log\left(1-\frac{1}{s}\right)\right]_s^{\infty}=\log\left(1-\frac{1}{s}\right)=\log\left(\frac{s-1}{s}\right)$$

(f) The given impulse function is $f(t)=\delta(t^2-3t+2)=\delta\left[(t-1)(t-2)\right]$
$$=\delta(t-1)u(t-1)+\delta(t-2)u(t-2)=\delta(t-1)+\delta(t-2)$$

Therefore, $$=\mathcal{L}[\delta(t-1)+\mathcal{L}[\delta(t-2)]$$
$$F(s)=e^{-s}+e^{-2s}$$

Example 3.2 *Determine the Laplace transform of the rectangular pulse shown in Fig. E3.2.*

Solution
$$F(s)=\mathcal{L}\{f(t)\}=\int_0^T f(t)\,e^{-st}\,\mathrm{d}t$$
$$=\int_0^T 1\cdot e^{-st}\,\mathrm{d}t=\left[\frac{e^{-st}}{-s}\right]_0^T=\frac{1}{-s}[e^{-sT}-1]$$
$$=\frac{1}{s}[1-e^{-sT}]$$

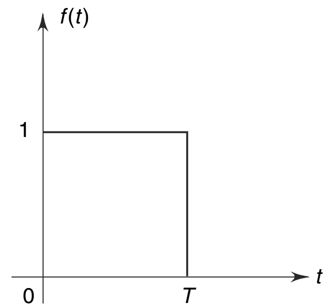


Fig. E3.2

Alternate method

The given pulse is represented in terms of the step function as
$$f(t) = u(t) - u(t-T)$$
Taking Laplace transform, we obtain
$$F(s)=\frac{1}{s}[1-e^{-sT}]$$

Example 3.3 *Find the Laplace transform of a single sawtooth pulse shown in Fig. E3.3.*

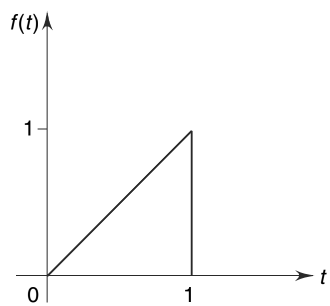


Fig. E3.3

Solution The function for the given waveform is
$$f(t)=\begin{cases}t & \text{for } 0<t\le 1\\ 0 & \text{for } t>1\end{cases}$$
$$\mathcal{L}\{f(t)\}=\int_0^{\infty}f(t)e^{-st}\,\mathrm{d}t=\int_0^1 te^{-st}\,\mathrm{d}t=\left[t\frac{e^{-st}}{-s}-1\frac{e^{-st}}{s^2}\right]_0^1$$
$$=\left[\left(\frac{e^{-s}}{-s}-\frac{e^{-s}}{s^2}\right)-\left(0-\frac{1}{s^2}\right)\right]=\frac{1}{s^2}-e^{-s}\left[\frac{1}{s}+\frac{1}{s^2}\right]$$

Example 3.4 *Find the Laplace transform of the triangular pulse shown in Fig. E3.4.*

Solution For the given triangular waveform,

$$f(t) = \begin{cases} \frac{2}{T}t, & \text{for } 0 \le t \le T/2 \\ 2 - \frac{2}{T}t, & \text{for } T/2 \le t \le T \end{cases}$$

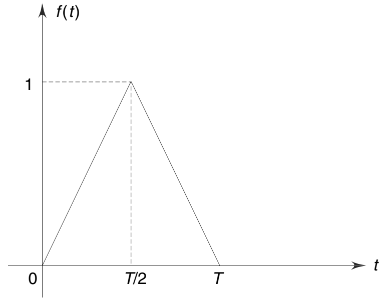


Fig. E3.4

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^{T/2} \left(\frac{2}{T}t\right) e^{-st} \mathrm{d}t + \int_{T/2}^{T} \left(2 - \frac{2}{T}t\right) e^{-st} \, \mathrm{d}t$$

$$= \frac{2}{T}\int_0^{T/2} te^{-st}\mathrm{d}t + 2\int_{T/2}^{T} e^{-st}\mathrm{d}t - \frac{2}{T}\int_0^{T/2} te^{-st}\mathrm{d}t$$

$$= \frac{2}{T}\left[\left\{t\frac{e^{-st}}{-s}\right\} - \left\{\frac{e^{-st}}{s^2}\right\}\right]^{T/2} + 2\left[\frac{e^{-st}}{-s}\right]_{T/2}^{T} - \frac{2}{T}\left[\left\{t\frac{e^{-st}}{-s}\right\} - \left\{\frac{e^{-st}}{s^2}\right\}\right]_{T/2}^{T}$$

$$= \frac{2}{T}\left[\frac{T}{2}\frac{e^{-sT/2}}{-s} + 0 - \frac{e^{-sT/2}}{s^2} + \frac{1}{s^2}\right] - \frac{2}{s}\left[e^{-sT} - e^{sT/2}\right]$$

$$- \frac{2}{T}\left[T\frac{e^{-sT}}{-s} + \frac{T}{2}\frac{e^{-sT/2}}{s} - \frac{e^{-sT}}{s^2} + \frac{e^{-sT/2}}{s^2}\right]$$

$$= \frac{e^{-sT/2}}{s} - \frac{2}{T}\frac{e^{-sT/2}}{s^2} + \frac{2}{T}\frac{1}{s^2} - \frac{2}{s}e^{-sT} + \frac{2}{s}e^{-sT/2}$$

$$+ 2\frac{e^{-sT}}{s} - \frac{e^{-sT/2}}{s} + \frac{2}{T}\frac{e^{-sT}}{s^2} - \frac{2}{T}\frac{e^{-sT/2}}{s^2}$$

$$= \frac{2}{T}\frac{1}{s^2} - \frac{4}{T}\frac{e^{-sT/2}}{s^2} + \frac{2}{T}\frac{e^{-sT}}{s^2}$$

Example 3.5 *Find the Laplace transform of the square wave shown in Fig. E3.5.*

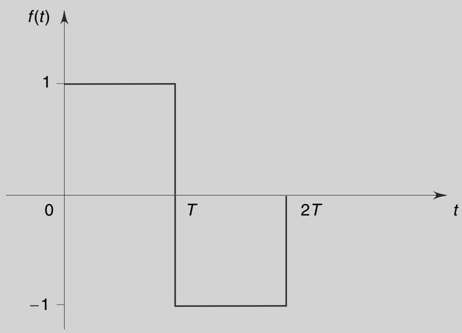


Fig. E3.5

Solution For the given waveform,

$$f(t) = \begin{cases} 1, & \text{for } 0 < t < T \\ -1, & \text{for } T < t < 2T \end{cases}$$

$$\mathcal{L}\{f(t)\} = \int_0^\infty f(t)e^{-st}\,\mathrm{d}t$$

$$= \int_0^T 1.e^{-st}\mathrm{d}t + \int_T^{2T} (-1)e^{-st}\mathrm{d}t = \left[\frac{e^{-st}}{-s}\right]_0^T - \left[\frac{e^{-st}}{-s}\right]_T^{2T}$$

$$= -\frac{1}{s}\left[e^{-sT} - 1 - e^{-2sT} + e^{-sT}\right] = \frac{1}{s}\left[1 - 2e^{-sT} + e^{-2sT}\right] = \frac{1}{s}\left(1 - e^{-sT}\right)^2$$

Example 3.6 *For the waveform shown in Fig. E3.6, find the Laplace transform.*

Solution The function for the given waveform is

$$f(t) = \begin{cases} A\sin t, & \text{for } 0 < t < \pi \\ 0, & \text{for } t > \pi \end{cases}$$

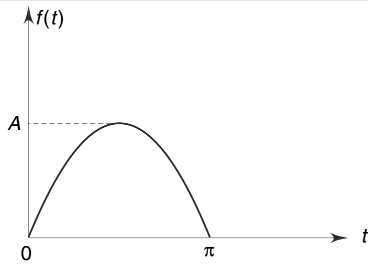


Fig. E3.6

By definition, we have

$$\mathcal{L}\{f(t)\} = \int_0^\infty f(t)e^{-st}\mathrm{d}t = \int_0^\pi A\sin t\, e^{-st}\mathrm{d}t$$

$$= A\int_0^\pi \sin t\, e^{-st}\mathrm{d}t = \frac{A}{\left(s^2+1\right)}\left[e^{-sT}(-s\sin t - \cos t)\right]_0^\pi = A\frac{e^{-s\pi}+1}{\left(s^2+1\right)}$$

INITIAL AND FINAL VALUE THEOREMS 3.5

3.5.1 Initial Value Theorem

If the function $f(t)$ and its derivative $f'(t)$ are Laplace transformable, then

$$\underset{t\to 0}{\mathrm{Lt}}\, f(t) = \underset{s\to\infty}{\mathrm{Lt}}\, sF(s)$$

Proof We know that

$$\mathcal{L}\{f'(t)\} = s[\mathcal{L}\{f(t)\}] - f(0)$$

By taking the limit $s \to \infty$ on both sides

$$\underset{s\to\infty}{\mathrm{Lt}}\, \mathcal{L}\{f'(t)\} = \underset{s\to\infty}{\mathrm{Lt}}\, [sF(s) - f(0)]$$

$$\underset{s\to\infty}{\mathrm{Lt}} \int_0^\infty f'(t)e^{-st}\mathrm{d}t = \underset{s\to\infty}{\mathrm{Lt}}\, [sF(s) - f(0)]$$

As $s \to \infty$, the integration of LHS becomes zero

i.e. $$\int_0^\infty \underset{s\to\infty}{\text{Lt}} [f'(t)e^{-st}]\,dt = 0$$

$$\underset{s\to\infty}{\text{Lt}}\ sF(s) - f(0) = 0$$

Therefore $$\underset{s\to\infty}{\text{Lt}}\ sF(s) = f(0) = \underset{t\to 0^+}{\text{Lt}}\ f(t) \tag{3.14}$$

3.5.2 Final Value Theorem

If $f(t)$ and $f'(t)$ are Laplace transformable, then

$$\underset{t\to\infty}{\text{Lt}}\ f(t) = \underset{s\to 0}{\text{Lt}}\ sF(s) \tag{3.15}$$

Proof We know that

$$\mathcal{L}\{f'(t)\} = sF(s) - f(0)$$

Taking the limits $s \to 0$ on both sides, we get

$$\underset{s\to 0}{\text{Lt}}\ \mathcal{L}\{f'(t)\} = \underset{s\to 0}{\text{Lt}}\ [sF(s) - f(0)]$$

$$\underset{s\to 0}{\text{Lt}} \int_0^\infty f'(t)e^{-st}\,dt = \underset{s\to 0}{\text{Lt}}\ [sF(s) - f(0)]$$

Therefore, $$\int_0^\infty f'(t)\,dt = \underset{s\to 0}{\text{Lt}}\ [sF(s) - f(0)]$$

$$[f(t)]_0^\infty = \underset{t\to\infty}{\text{Lt}}\ f(t) - \underset{t\to 0}{\text{Lt}}\ f(t) = \underset{s\to 0}{\text{Lt}}\ sF(s) - f(0)$$

Since $f(0)$ is not a function of s, it gets cancelled from both sides of the above equation.

Therefore, $$\underset{t\to\infty}{\text{Lt}}\ f(t) = \underset{s\to 0}{\text{Lt}}\ sF(s)$$

3.6 CONVOLUTION INTEGRAL

If $X(s)$ and $H(s)$ are the Laplace transforms of $x(t)$ and $h(t)$, then the product of $X(s)\,H(s) = Y(s)$, where $Y(s)$ is the Laplace transform of $Y(t)$ given by

$$y(t) = x(t) * h(t) = \int_0^t x(\tau)\,h(t-\tau)\,d\tau \tag{3.16}$$

$$Y(s) = X(s)H(s) \tag{3.17}$$

Proof Let $$y(t) = \int_0^t x(\tau)\,h(t-\tau)\,d\tau$$

$$Y(s) = \mathcal{L}[y(t)] = \int_0^\infty e^{-st}y(t)\,dt = \int_0^\infty e^{-st}\int_0^t x(\tau)\,h(t-\tau)\,d\tau\,dt$$

$$= \int_0^\infty \int_0^t e^{-st}x(\tau)\,h(t-\tau)\,d\tau\,dt$$

Changing the order of integration, the above equation becomes

$$\mathcal{L}\{y(t)\} = \int_0^\infty \int_\tau^\infty e^{-st} x(\tau)\, h(t-\tau)\, \mathrm{d}t\, \mathrm{d}\tau = \int_0^\infty x(\tau) \left[\int_\tau^\infty e^{-st}\, h(t-\tau)\, \mathrm{d}t \right] \mathrm{d}\tau$$

Putting $a = t - \tau$, then $t = a + \tau$ and $\mathrm{d}a = \mathrm{d}t$, we get

$$Y(s) = \mathcal{L}[y(t)] = \int_0^\infty x(\tau) \left[\int_0^\infty e^{-s(a+\tau)} h(a)\, \mathrm{d}a \right] \mathrm{d}\tau$$

$$= \int_0^\infty x(\tau) e^{-s\tau} d\tau \left[\int_0^\infty e^{-sa} h(a)\, \mathrm{d}a \right] = X(s) \cdot H(s)$$

Therefore, $Y(s) = X(s) \cdot H(s)$

The convolution of the signals in the time domain is equal to the multiplication of their individual Laplace transforms in the frequency domain.

$y(t) = \int_0^t x(\tau) h(t-\tau)\, \mathrm{d}\tau$ defines the convolution of functions $x(t)$ and $h(t)$ and is expressed symbolically as

$$y(t) = x(t) * h(t)$$

This theorem is very useful in frequency domain analysis.

Example 3.7 *Determine the convolution integral when $f_1(t) = e^{-2t}$ and $f_2(t) = e^{-2t}$.*

Solution We have $f_1(t) * f_2(t) = \int_0^t f_1(\tau) f_2(t-\tau)\, \mathrm{d}\tau$

Then, $f_1(t) * f_2(t) = \int_0^t 2\tau e^{-2(t-\tau)}\, \mathrm{d}\tau = e^{-2t} \int_0^t 2\tau e^{2\tau}\, \mathrm{d}\tau$

$$= 2e^{-2t} \left[\tau \frac{e^{2\tau}}{2} - \int 1.\frac{e^{2\tau}}{2} \mathrm{d}\tau \right]_0^t = 2e^{-2t} \left[\frac{te^{2t}}{2} - \frac{e^{2t}}{2} + \frac{1}{4} \right] = \left[t - \frac{1}{2} + \frac{e^{-2t}}{2} \right] u(t)$$

Example 3.8 *Determine the output response y(t) of the RC network shown in Fig. E3.8 due to a unit step as the input by using the convolution integral.*

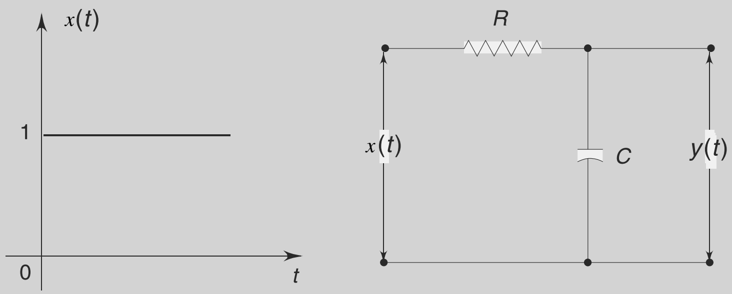


Fig. E3.8

Solution The transfer function of the network is

$$H(s) = \frac{1/Cs}{R + 1/Cs} = \frac{1}{RCs + 1} = \frac{1}{RC} \cdot \frac{1}{(s + 1/RC)}$$

Therefore, the impulse response of the network is

$$h(t) = \frac{1}{RC} e^{-t/RC} \qquad \text{for } t \geq 0$$

As the input signal is a unit step, $x(t) = \begin{cases} 0, & \text{for } t < 0 \\ 1, & \text{for } t \geq 0 \end{cases}$

Therefore, $$y(t) = \int_{-\infty}^{\infty} x(\tau) h(t-\tau)\, d\tau = \int_0^t (1/RC)\, e^{\frac{-(t-\tau)}{RC}}\, d\tau = \frac{e^{-t/Rc}}{RC} \int_0^t e^{\tau/Rc}\, d\tau$$

$$= \frac{e^{-t/RC}}{RC} \left[RC.e^{\tau/RC} \right]_0^t = e^{-t/RC} \left[e^{t/RC} - 1 \right] = 1 - e^{-t/RC}$$

Hence, $y(t) = 1 - e^{-t/RC}$ for $t \geq 0$

When a battery of 1 volt is switched into the network, we will get the same response.

Example 3.9 *Determine the output response y(t) of the RC low-pass network shown in Fig. E3.9 due to an input* $x(t) = te^{-t/RC}$ *by convolution.*

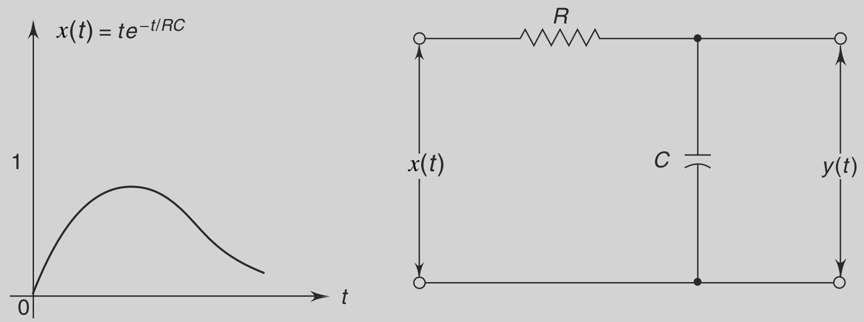


Fig. E3.9

Solution The transfer function of the network is

$$H(s) = \frac{\frac{1}{Cs}}{R + \frac{1}{Cs}} = \frac{1}{(RCs + 1)} = \frac{1}{RC} \cdot \frac{1}{\left(s + \frac{1}{RC}\right)}$$

The given input time function is $x(t) = te^{-t/RC}$

Therefore, $$X(s) = \mathcal{L}\{te^{-t/RC}\} = \frac{1}{\left(s + \frac{1}{RC}\right)^2}$$

We know that $Y(s) = X(s)\, H(s)$

Hence, $$Y(s) = \frac{1}{\left(s+\frac{1}{RC}\right)^2} \cdot \frac{1}{RC} \frac{1}{\left(s+\frac{1}{RC}\right)} = \frac{1}{RC} \cdot \frac{1}{\left(s+\frac{1}{RC}\right)^3}$$

Taking inverse Laplace transform, we obtain

$$y(t) = \frac{1}{RC} \frac{t^2 e^{-t/RC}}{2} u(t)$$

TABLE OF LAPLACE TRANSFORMS 3.7

Table 3.1 presents some functions and their corresponding Laplace transforms. Table 3.2 lists the properties of the Laplace transform. Table 3.3 gives the elements needed to develop the s-domain image of a given time domain circuit.

Table 3.1 *Laplace transforms pairs*

S.No.	*f(t)*	*F(s)*
1.	$\delta(t)$	1
2.	$\delta(t-a)$	e^{-as}
3.	$u(t)$	$\frac{1}{s}$
4.	$u(t-a)$	$\frac{e^{-as}}{s}$
5.	$\frac{t^n}{n!}u(t)$, n positive integer	$(-1)^n \frac{1}{s^{n+1}}$
6.	$e^{-at}\, u(t)$	$\frac{1}{s+a}$
7.	$\frac{t^n e^{-at}}{n!} u(t)$	$\frac{1}{(s+a)^{n+1}}$
8.	$\sin(\omega_0 t)\, u(t)$	$\frac{\omega_0}{s^2+\omega_0^2}$
9.	$\cos(\omega_0 t)\, u(t)$	$\frac{s}{s^2+\omega_0^2}$
10.	$t\cos(\omega_0 t)\, u(t)$	$\frac{s^2-\omega_0^2}{\left(s^2+\omega_0^2\right)^2}$

(*Contd.*)

S.No.	$f(t)$	$F(s)$
11.	$t \sin(\omega_0 t)\, u(t)$	$\frac{2\omega_0 s}{\left(s^2+\omega_0^2\right)^2}$
12.	$e^{-at} \sin(\omega_0 t)\, u(t)$	$\frac{\omega_0}{(s+a)^2+\omega_0^2}$
13.	$e^{-at} \cos(\omega_0 t)\, u(t)$	$\frac{s+a}{(s+a)^2+\omega_0^2}$
14.	$\sinh \omega_0 t$	$\frac{\omega_0}{s^2-\omega_0^2}$
15.	$\cosh \omega_0 t$	$\frac{s}{s^2-\omega_0^2}$
16.	$e^{-at} \sinh \omega_0 t$	$\frac{\omega_0}{(s+a)^2-\omega_0^2}$
17.	$e^{-at} \cosh \omega_0 t$	$\frac{s+a}{(s+a)^2-\omega_0^2}$
18.	$\sin(\omega_0 t + \theta)$	$\frac{s\sin\theta+\omega_0\cos\theta}{s^2+\omega_0^2}$
19.	$\cos(\omega_0 t + \theta)$	$\frac{s\cos\theta-\omega_0\sin\theta}{s^2+\omega_0^2}$

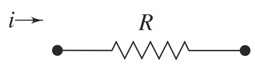
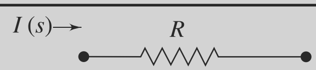
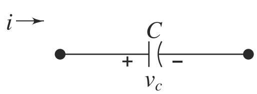
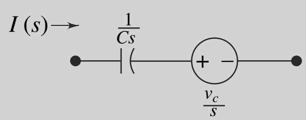
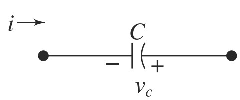
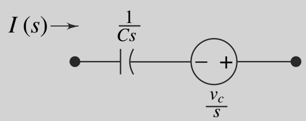
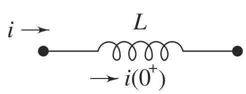
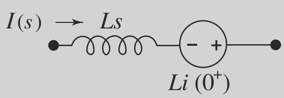
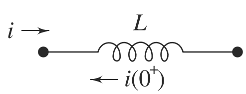
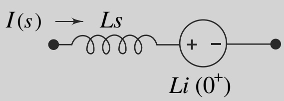
Table 3.2 *Properties of Laplace transform*

S.No.	Property	Time domain	Frequency domain
1.	Linearity	$a f_1(t) \pm b f_2(t)$ a and b are constants	$a F_1(s) \pm b F_2(s)$
2.	Scalar multiplication	$kf(t)$	$kF(s)$
3.	Scale change	$f(a t),\ a \geq 0$	$\frac{1}{a}F\left(\frac{s}{a}\right)$
4.	Time delay	$f(t-a),\ a \geq 0$	$F(s)\, e^{-as}$
5.	Frequency-shift	$e^{-at}f(t)$	$F(s + a)$
6.	Multiplication by t^n	$t^n f(t),\ n = 1, 2, \ldots.$	$(-1)^n \frac{d^n F(s)}{ds^n}$
7.	Time differentiation	$f'(t)$ $f''(t)$ $f^n(t')$	$s F(s) - f(0)$ $s^2F(s) - s f(0) - f'(0)$ $-s^{n-2} f'(0) - \ldots - f^{n-1}(0)$

(*Contd.*)

8.	Time integration	$\int_0^t \frac{(t-u)^{n-1}}{(n-1)!} f(u)\,\mathrm{d}u$	$\frac{F(s)}{s^n}$
9.	Frequency differentiation	$(-1)^n t^n f(t)$ $-t f(t)$ $t^2 f(t)$	$F^n(s)$ $F'(s) = \frac{\mathrm{d}F(s)}{\mathrm{d}s}$ $F''(s)$
10.	Frequency integration	$\frac{f(t)}{t}$	$\int_s^\infty F(s)\mathrm{d}s$
11.	Convolution	$f_1(t) * f_2(t) = \int_0^t f_1(\tau) f_2(t-\tau)\,\mathrm{d}\tau$	$F_1(s)F_2(s)$
12.	Final value	$f(\infty) = \lim_{t\to\infty} f(t)$	$\lim_{s\to 0} sF(s)$
13.	Initial value	$f(0^+) = \lim_{t\to 0^+} f(t)$	$\lim_{s\to\infty} sF(s)$
14.	Time periodicity	$f(t) = f(t + nT)$ $n = 1, 2, \ldots\ldots\ldots$	$\frac{1}{1-e^{-sT}} F_1(s)$ where $F_1(s) = \int_0^T f(t) e^{st} dt$

Table 3.3 *Representation of Laplace transform circuit elements*

Time domain	*s-domain*	*Voltage in s-domain*
		$RI(\text{s})$
		$\frac{I(s)}{Cs} + \frac{v_c}{s}$
		$\frac{I(s)}{Cs} - \frac{v_c}{s}$
		$LsI(s) - Li(0^+)$
		$LsI(s) + Li(0^+)$

PARTIAL FRACTION EXPANSIONS 3.8

For the given function $F(s) = \frac{N(s)}{D(s)}$, the inverse Laplace transform can be determined by expanding it into partial fractions. The degree of the numerator polynomial $N(s)$ must be lower than that of denominator polynomial $D(s)$. If the degree of the numerator is greater than or equal to the degree of the denominator, the numerator $N(s)$ is divided by the denominator $D(s)$ so that the remainder can be expanded more easily into partial fractions.

Case 1: For simple and real roots

$$F(s) = \frac{N(s)}{(s - p_0)(s - p_1)(s - p_2)}$$

where p_0, p_1 and p_2 are real roots and the degree of $N(s) < 3$.

Expanding $F(s)$ into partial fractions, we have

$$F(s) = \frac{A_0}{s - p_0} + \frac{A_1}{s - p_1} + \frac{A_2}{s - p_2}$$

The constant A_0 can be evaluated by multiplying $F(s)$ with $(s - p_0)$ and substituting $s = p_0$, as given below

$$A_0 = (s - p_0)F(s)|_{s=p_0}$$

Similarly, the other constants can be evaluated with the help of a general solution given by

$$A_i = (s - p_i)F(s)|_{s=p_i}$$

Case 2: For complex roots

Consider the function
$$F(s) = \frac{N(s)}{D_1(s)(s - \alpha + j\beta)(s - \alpha - j\beta)}$$
$$= \frac{A_1}{(s - \alpha + j\beta)} + \frac{A_2}{(s - \alpha - j\beta)} + \frac{N_1(s)}{D_1(s)}$$

where $\frac{N_1(s)}{D_1(s)}$ is the remainder term.

Hence the constants A_1 and A_2 are evaluated as

$$A_1 = F(s)(s - \alpha + j\beta)|_{s=\alpha-j\beta}$$
$$A_2 = F(s)(s - \alpha - j\beta)|_{s=\alpha+j\beta}$$

In general, $A_2 = A_1^*$, where A_1^* is the complex conjugate of A_1.

Case 3: Multiple or repeated roots

Here
$$F(s) = \frac{N_1(s)}{(s - p_i)^n D_1(s)}$$

The partial fraction expansion of $F(s)$ is

$$F(s) = \frac{A_0}{(s - p_i)^n} + \frac{A_1}{(s - p_i)^{n-1}} + \frac{A_2}{(s - p_i)^{n-2}} + \ldots + \frac{A_{n-1}}{(s - p_i)} + \frac{N_1(s)}{D_1(s)}$$

where $\frac{N_1(s)}{D_1(s)}$ is the remainder term.

For evaluating the constants $A_0, A_1, A_2 \ldots .A_{n-1}$, we have to multiply both sides of the above equation by $(s-p_i)^n$.

Hence, $$(s-p_i)^n\ F(s) = A_0 + A_1(s-p_i) + A_2(s-p_i)^2 + \ldots + A_{n-1}(s-p_i)^{n-1} + \frac{N_1(s)}{D_1(s)}(s-p_i)^n$$

Substituting $s = p_i$, we get

$$A_0 = (s - p_i)^n F(s)\,|_{s=pi}$$

Differentiating $F_1(s)$ with respect to s, we get

$$\frac{\mathrm{d}}{\mathrm{d}_s} F_1(s) = A_1 + 2A_2(s-p_i) + \ldots. + A_{n-1}(n-1)(s-p_i)^{n-2} + \frac{\mathrm{d}}{\mathrm{d}s}\left(\frac{N_1(s)}{D_1(s)}(s-p_i)^n\right)$$

Substituting $s = p_i$ in the above equation, we get

$$A_1 = \frac{\mathrm{d}}{\mathrm{d}s} F_1(s)\,|_{s=p_i}$$

Similarly, $$A_2 = \frac{1}{2!}\frac{\mathrm{d}^2}{\mathrm{d}s^2} F_1(s)\,|_{s=p_i}$$

Generally, $$A_n = \frac{1}{n!}\frac{\mathrm{d}^n}{\mathrm{d}s^n} F_1(s)\,|_{s=p_i} \qquad \text{where } n = 0,1,2,\ \ldots\ n-1$$

NETWORK TRANSFER FUNCTION 3.9

The transfer function $H(s)$ of the LTI system, as shown in Fig. 3.2, is equal to the ratio of the Laplace transform $Y(s)$ of the output signal to the Laplace transform $X(s)$ of the input signal when initial conditions are zero. Thus

$$H(s) = \frac{Y(s)}{X(s)} = \left.\frac{\text{Laplace transform of output}}{\text{Laplace transform of input}}\right|_{\text{All initial conditions are zero}} \tag{3.18}$$

The transfer function of a system $H(s)$ is the Laplace transform of the impulse response $h(t)$. The transfer function $H(s)$ is strictly analogous to the frequency response used in Fourier analysis. There is a considerable similarity between the transfer functions of Laplace transform and Fourier transform.

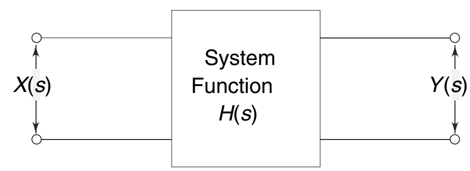


Fig. 3.2 *Transfer Function of a System*

The main advantage of transformed functions is that the time-domain convolution is replaced by frequency-domain multiplication. Hence,

$$Y(s) = X(s)\,H(s) \tag{3.19}$$

If the Laplace transform $Y(s)$ of the output signal is determined, then its equivalent time domain function $y(t)$ can be determined by taking the inverse Laplace transform.

3.9.1 Step and Impulse Responses

We know that the Laplace transform of a unit impulse $\delta(t)$ is unity, i.e. $\mathcal{L}\{\delta(t)\} = 1$. If the unit impulse is given as the system excitation, i.e. $X(s) = 1$, then the output response will be

$$\begin{aligned} Y(s) &= X(s)\,H(s) \\ &= H(s) \end{aligned}$$

Thus, it is shown that the impulse response $h(t)$ and the transfer function of the system $H(s)$ constitute a transform pair, i.e.

$$\begin{aligned} \mathcal{L}\{h(t)\} &= H(s) \\ \mathcal{L}^{-1}\{H(s)\} &= h(t) \end{aligned}$$

This implies that the step and impulse responses can be directly obtained from the system function.

The step response is the integral of the impulse response. Hence, the integral property of the Laplace transform is used to obtain the step response $\alpha(t)$ as

$$\alpha(t) = \mathcal{L}^{-1}\left\{\frac{H(s)}{s}\right\}$$

The unit ramp response $\gamma(t)$ is obtained from the equation given by

$$\gamma(t) = \mathcal{L}^{-1}\left\{\frac{H(s)}{s^2}\right\}$$

s-PLANE POLES AND ZEROS 3.10

The transfer function of a linear time-invariant system may be expressed as

$$H(s) = \frac{Y(s)}{X(s)} = \frac{a_n s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}{b_m s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + b_0} \qquad (3.20)$$

The system function is a rational function of s and is expressed as

$$H(s) = \frac{N(s)}{D(s)} = k\frac{(s-z_1)(s-z_2)(s-z_3)\cdots}{(s-p_1)(s-p_2)(s-p_3)\cdots} \qquad (3.21)$$

where k is a real number. The constants $z_1, z_2, z_3, \ldots$ are referred to as the zeros of $H(s)$, as they are values of s at which $H(s)$ is zero. Conversely, $p_1, p_2, p_3 \ldots$ are called the poles of $H(s)$, as they are values of s at which $H(s)$ becomes infinity.

The poles and zeros may be plotted in a complex plane diagram, referred to as the **s-plane**. In the complex s-plane, a pole is denoted by a small cross, and a zero by a small circle. This diagram gives a good indication of the degree of stability of a system.

3.10.1 Stability in the s-Domain

Stability is an important characteristic of the causal time-invariant system. The stability can be determined in the frequency or s-domain. Also, it does not depend on the input excitation applied to the system. The roots of the denominator polynomial or the poles of the transfer function of the linear time-invariant system will determine whether the system is stable, unstable or marginally stable, provided that the degree of the denominator polynomial is greater than or equal to the degree of the numerator polynomial.

(i) For a *stable* system, all the poles of the transfer function must lie in the left half of the s-plane.
(ii) A system is said to be *unstable*, if any poles of $H(s)$ are located in the right half of the s-plane.
(iii) A system is said to be *marginally stable*, if $H(s)$ has any poles on the $j\omega$-axis in the s-plane provided the other poles of $H(s)$ lie in the left half of the s-plane.

By finding the location of the poles, i.e. the roots of the denominator polynomial of the transfer function, the stability of a linear time-invariant system can be determined. Though computer programs are available for finding the roots of the denominator polynomial of order higher than three, it is difficult to find the range of a parameter for stability. In such cases, especially in control system design, an analytical procedure called the *Routh-Hurwitz stability criterion* is used.

3.10.2 Magnitude and Phase Responses

The transfer function of the analog filter is evaluated on the imaginary axis of the s-plane. The geometric evaluation of the magnitude and phase responses of the system with a greater number of poles and zeros can be obtained from the pole-zero diagram. Suppose we have the transfer function of the system as

$$H(s) = k\frac{(s-z_1)(s-z_2)(s-z_3)\cdots(s-z_n)}{(s-p_1)(s-p_2)(s-p_3)\cdots(s-p_m)}$$

Hence,

$$H(j\omega) = M(\omega)\Phi(\omega) = k\frac{(j\omega-z_1)(j\omega-z_2)(j\omega-z_3)\cdots(j\omega-z_n)}{(j\omega-p_1)(j\omega-p_2)(j\omega-p_3)\cdots(j\omega-p_m)}$$

$$= k\frac{\prod_{i=1}^{n}(j\omega-z_i)}{\prod_{i=1}^{m}(j\omega-p_i)} \tag{3.22}$$

The vectors are drawn from each pole and zero to any point on the imaginary axis. Therefore, the factors $(j\omega - z_i)$ and $(j\omega - p_i)$ may be expressed in polar form as

$$(j\omega-z_i) = M_{zi}e^{j\Phi z_i} \quad \text{and} \quad (j\omega-p_i) = M_{pi}e^{j\Phi p_i}$$

Hence, we have the magnitude response as

$$M(\omega) = |H(j\omega)| = \frac{k\prod_{i=1}^{n}M_{zi}}{\prod_{i=1}^{m}M_{pi}}$$

$$= \frac{k\prod_{i=1}^{n}(\text{vector magnitudes from the zeros } z_i \text{ to the point on the } j\omega \text{ axis})}{\prod_{i=1}^{m}(\text{vector magnitudes from the poles } p_i \text{ to the point on the } j\omega \text{ axis})} \tag{3.23}$$

That is, the magnitude of the system function $M(\omega)$ equals the product of all zero vector lengths divided by the product of all pole vector lengths.

Similarly, the phase response is given as

$$\Phi(\omega) = \angle H(j\omega)$$

$$= \sum_{i=1}^{n} \Phi_{zi} - \sum_{i=1}^{m} \Phi_{pi}$$

where Φ_{zi} = Angles of the vectros from the zeros z_i to the $j\omega$ axis

and Φ_{pi} = Angles of the vectros from the zeros p_i to the $j\omega$ axis (3.24)

That is, the phase of the system function is equal to the sum of all zero vector phases minus the sum of all pole vector phases.

Time delay (τ)

Let us consider the transfer function of pure time delay τ, i.e., $\mathcal{L}\{\delta(t-\tau)\} = e^{-s\tau}$ or in general $\mathcal{L}\{h(t-\tau)\} = e^{-s\tau}\ \mathcal{L}\{h(t)\}$.

Hence, for the pure delay, $H(j\omega) = e^{-j\omega\tau}$

The magnitude response of the pure delay is $|H(j\omega)| = 1$ and the phase response of the pure delay is $\Phi(j\omega) = -\omega\tau$.

Therefore, the time delay τ is equal to minus the derivative of the phase response, i.e.

$$\tau = -\frac{d\Phi(j\omega)}{d\omega}$$

Example 3.10 *Draw the poles and zero for the current I(s) in a network given by* $I(s) = \frac{3s}{(s+2)(s+4)}$ *and hence obtain i(t).*

Solution The zero occurs at $s = 0$ and the poles at $s = -2$ and $s = -4$ as shown in Fig. E3.10.

The given function $I(s)$ can be expanded by partial fraction as

$$I(s) = \frac{A_1}{(s+2)} + \frac{A_2}{(s+4)}$$

The coefficients A_1 and A_2 may be evaluated from the pole-zero diagram.

$$A_1 = k\frac{\text{Magnitude and phase angle of phasor from zero at } z_0 \text{ to pole at } p_1}{\text{Magnitude and phase angle of phasor from zero at } p_2 \text{ to pole at } p_1}$$

$$= 3\cdot\frac{2\ \ 180^\circ}{2\ \ 0^\circ} = 3\ \ 180^\circ = 3.\ (\cos 180^\circ + j \sin 180^\circ) = 3 \times -1 = -3$$

Similarly,

$$A_2 = 3\cdot\frac{4\ \ \ \ ^\circ}{2\ \ 180^\circ} = 6$$

Substituting the values t of A_1 and A_2, we get

$$I(s) = -\frac{3}{(s+2)} + \frac{6}{(s+4)}$$

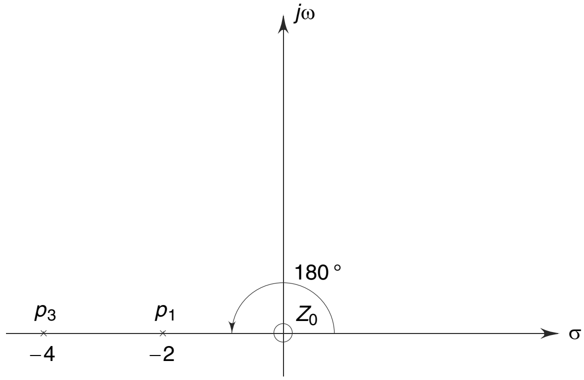


Fig. E3.10 *Pole Zero Plot of I(s)*

Taking inverse Laplace transform, we get

$$i(t) = -3e^{-2t} + 6e^{-4t}$$

Example 3.11 *Plot the poles and zeros for* $F(s) = 4 \cdot \frac{(s+1)(s+3)}{(s+2)(s+4)}$ *and hence obtain f(t).*

Solution The poles and zeros are plotted in Fig. E3.11(a). For evaluating *f*(*t*), the degree of the numerator polynomial must be one degree less than the degree of the denominator polynomial.

Dividing numerator polynomial by denominator polynomial, we get

$$F(s) = 4\frac{(s+1)(s+3)}{(s+2)(s+4)} = 4\left[1 - \frac{2s+5}{(s+2)(s+4)}\right]$$

$$= 4 - 8\left[\frac{s+2.5}{(s+2)(s+4)}\right]$$

$$= 4 - 8\left[\frac{A_1}{(s+2)} + \frac{A_2}{(s+4)}\right]$$

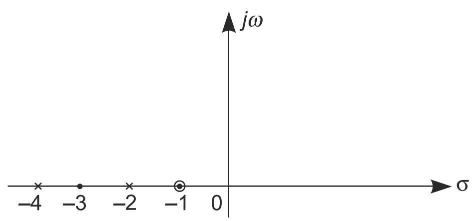


Fig. E3.11(a)

To Find the Coefficients A_1 and A_2

The poles and zero for the given function $\left[\frac{s+2.5}{(s+2)(s+4)}\right]$ are plotted in Fig. E3.11(b) from which the coefficients A_1 and A_2 can be calculated.

$$A_1 = \frac{0.5 \quad \circ}{2 \quad \circ} = \frac{1}{4}$$

and

$$A_2 = \frac{1.5 \; 180^\circ}{2 \quad \circ} = \frac{3}{4}$$

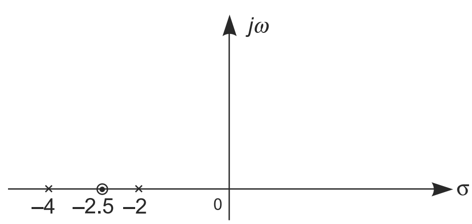


Fig. E3.11(b)

Therefore, $F(s) = 4 - 8\left[\frac{1/4}{(s+2)} + \frac{3/4}{(s+4)}\right]$

Taking inverse Laplace transform, we get

$$f(t) = 4\delta(t) - 2e^{-2t} - 6e^{-4t}$$

Example 3.12 *The transfer function of a system is given by*

$$H(s) = \frac{10s}{s^2 + 2s + 2}$$

Plot the pole-zero diagram and hence (a) determine h(t) and (b) evaluate the magnitude and phase responses at ω = 2 using the geometrical method.

Solution

$$H(s) = \frac{10s}{s^2 + 2s + 2}$$

$$= \frac{10s}{(s + 1 - j1)(s + 1 + j1)}$$

There is a zero at $s = 0$ and poles at $s = -1 + j1$ and $s = -1 - j1$. The poles and zero are plotted in Fig. E3.12(a).

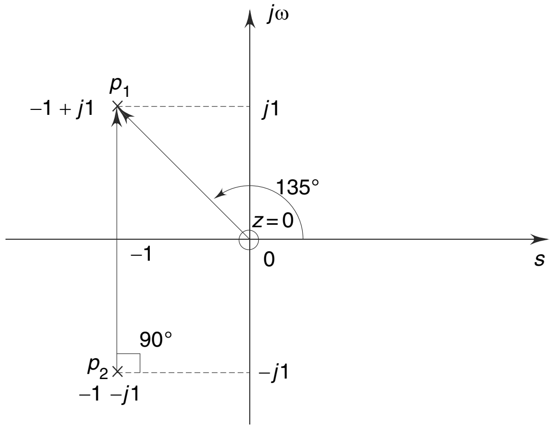


Fig. E3.12(a) *Evaluation of A_1*

(a) $$H(s) = \frac{10s}{(s + 1 - j1)(s + 1 + j1)}$$

$$= \frac{A_1}{(s + 1 - j1)} + \frac{A_1^*}{(s + 1 + j1)}$$

$$A_1 = 10\frac{\sqrt{2}\ 135°}{2\ 90°} = 5\sqrt{2} \qquad ° = 7.07e^{j\pi/4}$$

$$h(t) = A_1 e^{-(1-j1)t} + A_1^* e^{-(1+j1)t}$$
$$= 7.07\, e^{j\pi/4}\, e^{-t}\, e^{jt} + 7.07 e^{-j\pi/4}\, e^{-t} e^{-jt}$$
$$= 7.07\, e^{-t}[e^{j(1 + \pi/4)t} + e^{-j(1+\pi/4)t}]$$
$$= 14.14\, e^{-t} \cos(\pi/4 + 1)t$$

(b) At $\omega = 2$, the phasors from poles and zero are drawn to the testing point A at $j2$ as shown in Fig. E3.12(b)

Magnitude $$M(j2) = 10 \times \frac{2}{\sqrt{2} \times \sqrt{10}}$$
$$= \sqrt{20} = 4.47$$

and Phase $$\Phi(j2) = \frac{\angle 90°}{\angle \tan^{-1}(1)\ \angle \tan^{-1}(3)}$$
$$= \frac{\angle 90°}{\angle 45°\ \angle 71.8°}$$
$$= 90° - 45° - 71.8° = -26.8°$$

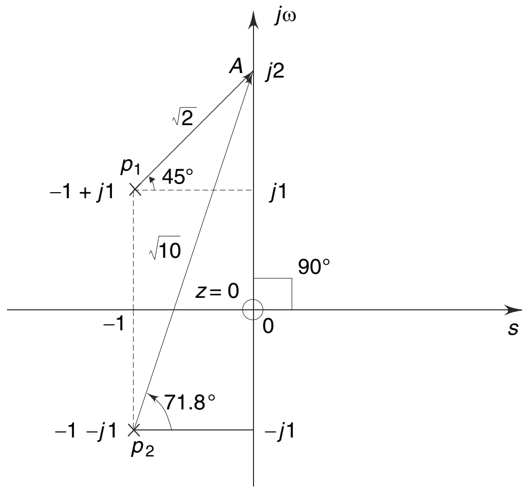


Fig. E3.12(b)

LAPLACE TRANSFORM OF PERIODIC FUNCTIONS 3.11

The time-shift theorem is useful in determining the transform of periodic time functions. Let function $f(t)$ be a causal periodic waveform which satisfies the condition $f(t) = f(t + nT)$ for all $t > 0$ where T is the period of the function and $n = 0, 1, 2,\ldots$

$$F(s) = \int_0^\infty f(t)e^{-st}\,dt$$

$$= \int_0^T f(t)e^{-st}\,dt + \int_T^{2T} f(t)e^{-st}\,dt + \cdots + \int_{nT}^{(n+1)T} f(t)e^{-st}\,dt + \cdots$$

As $f(t)$ is periodic, the above equation becomes

$$= \int_0^T f(t)e^{-st}\,dt + e^{-sT}\int_0^T f(t)e^{-st}\,dt + \cdots + e^{-nsT}\int_0^T f(t)e^{-st}\,dt + \cdots$$

$$= \left[1 + e^{-sT} + e^{-2sT} + \cdots + e^{-nsT} + \cdots\right]\int_0^T f(t)e^{-st}\,dt$$

$$= \left[1 + e^{-sT} + \left(e^{-sT}\right)^2 + \cdots + \left(e^{-sT}\right)^n + \cdots\right]F_1(s)$$

where $\quad F_1(s) = \int_0^T f(t)e^{-st}\,dt$

Here, $F_1(s) = \mathcal{L}\{[u(t) - u(t-T)]f(t)\}$, which is the transform of the first period of the time function, and $\{[u(t) - u(t - T)]f(t)\}$ has non-zero value only in the first period of $f(t)$.

When we apply the binomial theorem to the bracketed expression, it becomes $1/(1 - e^{-sT})$.

$$F(s) = \frac{1}{1 - e^{-sT}}\int_0^T f(t)e^{-st}\,dt = \frac{F_1(s)}{1 - e^{-sT}}$$

Example 3.13 *Find the Laplace transform of the periodic rectangular waveform shown in Fig. E3.13.*

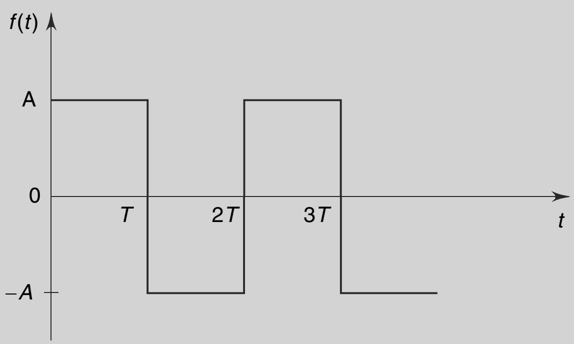


Fig. E3.13

Solution Here the period is $2T$

Therefore, $\mathcal{L}\{f(t)\} = \dfrac{1}{1-e^{-2sT}}\left[\int_0^{2T} f(t)e^{-st}\,dt\right]$

$$= \frac{1}{1-e^{-2sT}}\left[\int_0^{T} A\,e^{-st}\,dt + \int_T^{2T} (-A)\,e^{-st}\,dt\right]$$

$$= \frac{1}{1-e^{-2sT}}\left[\frac{-A}{s}\left(e^{-st}\right)_0^T + \frac{A}{s}\left(e^{-st}\right)_T^{2T}\right]$$

$$= \frac{1}{1-e^{-2sT}}\left[-\frac{A}{s}\left(e^{-sT}-1\right) + \frac{A}{s}\left(e^{-2sT}-e^{-sT}\right)\right]$$

$$= \frac{1}{1-e^{-2sT}}\left[\frac{A}{s}\left(1-e^{-sT}\right)^2\right] = \frac{A}{s}\left(\frac{\left(1-e^{-sT}\right)^2}{\left(1-e^{-sT}\right)\left(1+e^{-sT}\right)}\right) = \frac{A}{s}\left(\frac{1-e^{-sT}}{1+e^{-sT}}\right) = \frac{A}{s}\tanh\left(\frac{sT}{2}\right)$$

Example 3.14 *Find the Laplace transform of the periodic sawtooth waveform shown in Fig. E3.14.*

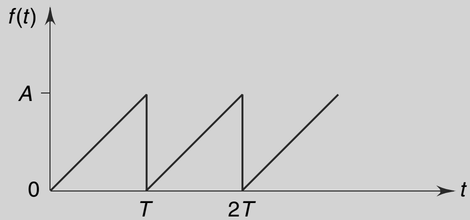


Fig. E3.14

Solution For the given transform, the period of one cycle is T. Hence,

$$\mathcal{L}\{f(t)\} = \frac{1}{1-e^{-sT}}\left[\int_0^{T} f(t)e^{-st}\,dt\right], \qquad \text{where } f(t) = \frac{A}{T}t$$

$$= \frac{1}{1-e^{-sT}}\left[\int_0^{T} \frac{A}{T}t\,e^{-st}\,dt\right] = \frac{1}{1-e^{-sT}}\,\frac{A}{T}\int_0^{T} t\,e^{-st}\,dt$$

$$= \frac{A}{T}\cdot\frac{1}{1-e^{-sT}}\left[\left\{t\frac{e^{-st}}{-s}\right\}_0^T - \left\{\frac{e^{-st}}{s^2}\right\}_0^T\right]$$

$$= \frac{A}{T}\cdot\frac{1}{1-e^{-sT}}\left[\left\{T\frac{e^{-sT}}{-s} - \frac{e^{-sT}}{s^2} + \frac{1}{s^2}\right\}\right]$$

$$= \frac{A}{Ts^2\left(1-e^{-sT}\right)}\left(1-e^{-sT} - sTe^{-sT}\right)$$

Example 3.15 *Find the Laplace transform of the full wave rectified output as shown in Fig. E3.15.*

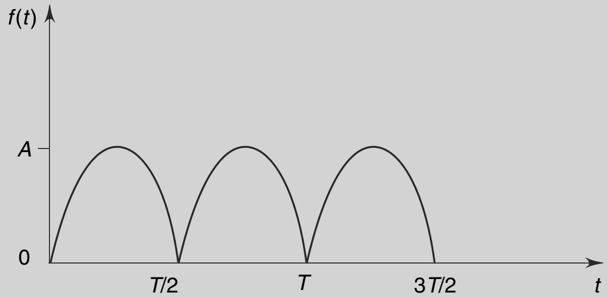


Fig. E3.15

Solution The function for the given waveform is

$$f(t) = A \sin \omega_0 t \text{ for } 0 < t < T/2$$

Hence,
$$\mathcal{L}\{f(t)\} = \frac{1}{1-e^{-sT/2}} \int_0^{T/2} f(t) e^{-st} \, dt$$

$$= \frac{A}{1-e^{-sT/2}} \int_0^{T/2} \sin \omega_0 t \, e^{-st} dt$$

$$= \frac{A}{1-e^{-sT/2}} \left[\frac{e^{-st}}{s^2+\omega_0^2} \left(-s \sin\omega_0 t - \omega_0 \cos \omega_0 t\right) \right]_0^{T/2}$$

$$= \frac{A}{1-e^{-sT/2}} \cdot \frac{1}{\left(s^2+\omega_0^2\right)} \left[\omega_0 e^{-sT/2} + \omega_0 \right]$$

$$= \frac{A\omega_0}{s^2+\omega_0^2} \frac{(1+e^{-sT/2})}{(1-e^{-sT/2})} = \frac{A\omega_0}{s^2+\omega_0^2} \frac{e^{sT/4}+e^{-sT/4}}{e^{sT/4}-e^{-sT/4}} = \frac{A\omega_0}{s^2+\omega_0^2} \coth(sT/4)$$

APPLICATION OF LAPLACE TRANSFORMATION IN ANALYSING NETWORKS 3.12

3.12.1 Step and Impulse Responses of Series RL Circuit

Step Response

In the series *RL* circuit shown in Fig. 3.3, let the switch, *S*, be closed at time $t = 0$. For the d.c. response, the input excitation is $V_i u(t)$. Applying Kirchhoff's voltage law to the circuit, we get the following differential equation.

$$L \frac{di(t)}{dt} + Ri(t) = V_i u(t) \tag{3.25}$$

Taking Laplace transform, the above equation becomes

$$L\left\{sI(s)-i(0^-)\right\}+RI(s)=\frac{V_i}{s} \tag{3.26}$$

Because of the presence of inductance L, $i(0^-) = 0$, i.e. the current through an inductor cannot change instantaneously due to the conservation of flux linkages.

Therefore, $$I(s)[sL+R]=\frac{V_i}{s}$$

Hence, $$I(s)=\frac{V_i}{sL\left(s+\frac{R}{L}\right)}$$

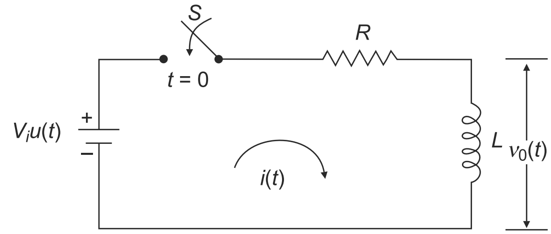


Fig. 3.3 *Series RL Circuit*

Using partial fraction expansion,

$$I(s)=\frac{V_i}{L}\left(\frac{A_1}{s}+\frac{A_2}{s+\frac{R}{L}}\right)$$

where $$A_1=sI(s)\big|_{s=0}=\frac{V_i}{L}\times\frac{1}{s+\frac{R}{L}}\bigg|_{s=0}=\frac{V_i}{R}$$

and $$A_2=\left(s+\frac{R}{L}\right)I(s)\bigg|_{s=0}=\frac{V_i}{L}\times\frac{1}{s}\bigg|_{s=-\frac{R}{L}}=-\frac{V_i}{R}$$

Therefore, $$I(s)=\frac{V_i}{L}\frac{L}{R}\left[\frac{1}{s}-\frac{1}{s+R/L}\right]=\frac{V_i}{R}\left[\frac{1}{s}-\frac{1}{s+R/L}\right]$$

Taking inverse Laplace transform, we get

$$i(t)=\frac{V_i}{R}\left[1-e^{-\frac{R}{L}t}\right] \tag{3.27}$$

In an *RL* circuit, the time constant is L/R, i.e. the time required for the current to reach from its initial value of zero to the final value V_i/R, where V_i is the source voltage. The above equation shows that there is an exponential increase in current as time t increases as shown in Fig. 3.4. As t tends to infinity, the current $i(t)$ approaches the peak value of V_i/R.

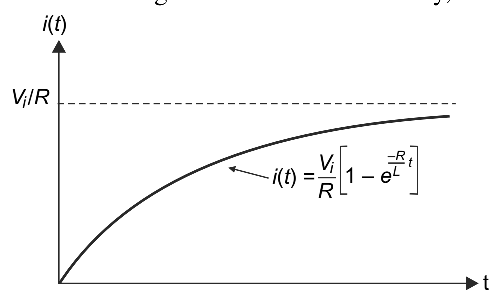


Fig. 3.4

Impulse Response

For the impulse response, the input excitation is $x(t) = \delta(t)$. Hence, the differential equation becomes

$$L\frac{\mathrm{d}i(t)}{\mathrm{d}t}+Ri(t)=\delta(t)$$

$$L\left\{sI(s)-i(0^+)\right\}+RI(s)=1$$

Since $i(0^+) = 0$,

$$I(s) = \frac{1}{Ls+R} = \frac{1}{L} \times \frac{1}{s+R/L}$$

Taking inverse Laplace transform, we get

$$i(t) = \frac{1}{L} \times e^{-(R/L)t} u(t)$$

Example 3.16 *Derive the expression for current in the circuit shown in Fig. E3.16 when the switch, S, is closed at t = 0.*

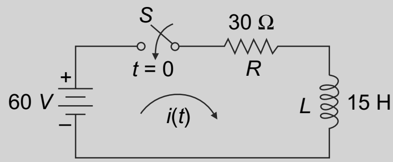


Fig. E3.16

Solution Applying Kirchhoff's voltage law to the given circuit, we get

$$15\frac{di(t)}{dt} + 30i(t) = 60$$

$$\frac{di(t)}{dt} + 2i(t) = 4$$

Taking Laplace transform, we get

$$\{sI(s) - i(0^-)\} + 2I(s) = \frac{4}{s}$$

Since there is no initial current flowing through inductor before closing the switch, S, the current remains zero at time $t = 0$, i.e. $i(0^-) = 0$. Hence, the above equation becomes

$$I(s)(s+2) = \frac{4}{s}$$

i.e. $$I(s) = \frac{4}{s(s+2)}$$

Using partial fraction,

$$I(s) = \frac{4}{s(s+2)} = \frac{A_1}{s} + \frac{A_2}{s+2}$$

where $$A_1 = sI(s)\big|_{s=0} = \frac{4}{s+2}\bigg|_{s=0} = 2$$

and $$A_2 = (s+2)I(s)\big|_{s=0} = \frac{4}{s}\bigg|_{s=-2} = -2$$

Substituting the values of A_1 and A_2, we get

$$I(s) = \frac{2}{s} - \frac{2}{s+2}$$

Taking inverse Laplace transform, we get

$$i(t) = 2\left(1 - e^{-2t}\right)u(t)$$

Example 3.17 *Determine the current i(t) flowing through the series RL circuit shown in Fig. E3.17.*

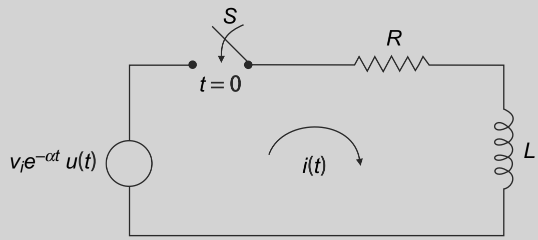


Fig. E3.17

Solution Applying Kirchhoff's voltage law to the given circuit, we get

$$L\frac{di(t)}{dt} + Ri(t) = V_i e^{-\alpha t} u(t)$$

Taking Laplace transform, we get

$$L\left\{sI(s) - i\left(0^+\right)\right\} + RI(s) = V_i \frac{1}{s+\alpha}$$

Since there is no initial current flowing through inductor before closing the switch, S, the current remains zero at time $t = 0$, i.e. $i(0^+) = 0$. Hence, the above equation becomes

$$I(s)(sL + R) = \frac{V_i}{s+\alpha}$$

i.e. $$I(s) = \frac{V_i}{L} \times \frac{1}{(s+\alpha)\left[s + \frac{R}{L}\right]}$$

Case (i): For $\alpha \neq \frac{R}{L}$

Using partial fraction expansion, we get

$$I(s) = \frac{V_i}{L} \times \frac{1}{(s+\alpha)\left[s + \frac{R}{L}\right]} = \frac{A_1}{s+\alpha} + \frac{A_2}{s + \frac{R}{L}}$$

where $$A_1 = (s+\alpha)I(s)\big|_{s=0} = \frac{V_i}{L} \times \left.\frac{1}{\left[s + \frac{R}{L}\right]}\right|_{s=-\alpha} = \frac{V_i}{R - \alpha L}$$

and $$A_2 = \left(s + \frac{R}{L}\right) I(s)\Big|_{s=0} = \frac{V_i}{L} \times \frac{1}{(s+\alpha)}\Bigg|_{s=-\frac{R}{L}} = -\frac{V_i}{R-\alpha L}$$

Substituting A_1 and A_2, we get

$$I(s) = \frac{V_i}{R-\alpha L}\left[\frac{1}{s+\alpha} - \frac{1}{s+\frac{R}{L}}\right]$$

Taking inverse Laplace transform, we get the current flowing through *RL* series circuit as

$$i(t) = \frac{V_i}{R-\alpha L}\left[e^{-\alpha t} - e^{-\frac{R}{L}t}\right]u(t)$$

Case (ii): For $\alpha = \frac{R}{L}$

$$I(s) = \frac{V_i}{L}\cdot\frac{1}{(s+\alpha)^2}$$

Taking inverse Laplace transform, we get the current flowing through *RL* series circuit as

$$i(t) = \frac{V_i}{L} t e^{-\alpha t} u(t)$$

Example 3.18 *In the circuit of Fig. E3.18, find the currents $i_1(t)$ and $i_2(t)$ and the output voltage across the 5Ω resistor when the switch is closed, and (b) also determine the initial and final values of current.*

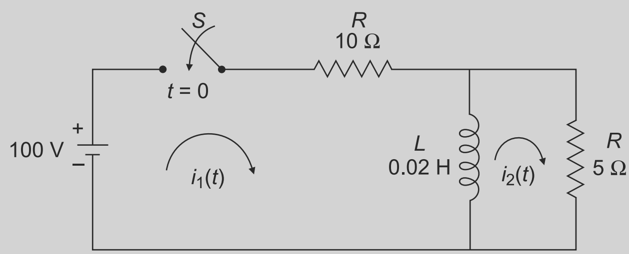


Fig. E3.18

Solution Applying Kirchhoff's voltage law to the circuit, we have

$$10i_1(t) + 0.02\frac{di_1(t)}{dt} - 0.02\frac{di_2(t)}{dt} = 100$$

$$0.02\frac{di_2(t)}{dt} + 5i_2(t) - 0.02\frac{di_1(t)}{dt} = 0$$

Taking Laplace transform for the above two equations, we get

$$(10 + 0.02s)I_1(s) - 0.02sI_2(s) = 100/s$$

$$(5 + 0.02s)I_2(s) - 0.02sI_1(s) = 0$$

Solving the equations, we have

$$I_2(s) = I_1(s)\left(\frac{s}{s+250}\right)$$

$$I_1(s) = \frac{20}{3}\left\{\frac{s+250}{s(s+500/3)}\right\}$$

$$I_2(s) = \frac{20}{3}\left(\frac{1}{s+500/3}\right) \tag{1}$$

Using partial fraction expansion, we get

$$I_1(s) = \frac{20}{3}\left\{\frac{s+250}{s(s+500/3)}\right\} = \frac{A_1}{s} + \frac{A_2}{s+500/3}$$

where $A_1 = sI_1(s)\big|_{s=0} = \frac{20}{3}\left\{\frac{s+250}{s+500/3}\right\}\Big|_{s=0} = 10$

and $A_2 = \left(s+\frac{500}{3}\right)I_1(s)\Big|_{s=0} = \frac{20}{3}\left[\frac{s+250}{s}\right]\Big|_{s=-\frac{500}{3}} = -\frac{10}{3}$

Therefore, $I_1(s) = \frac{10}{s} - \frac{10/3}{s+500/3}$ (2)

Taking inverse Laplace transform of Eq. (2) and Eq. (1), we get

$$i_1(t) = 10 - \frac{10}{3}e^{-(500/3)t}u(t)$$

$$i_2(t) = \frac{20}{3}e^{-(500/3)t}u(t)$$

The voltage across the 5 Ω resistor is $5i_2(t) = \frac{100}{3}e^{-(500/3)t}$ V

To find the initial and final values of the currents

The initial value of $i_1(t)$ is

$$i_1(0^+) = \underset{s\to\infty}{Lt}\left[sI_1(s)\right] = \underset{s\to\infty}{Lt}\left[\frac{20}{3}\left(\frac{s+250}{s+500/3}\right)\right] = \underset{s\to\infty}{Lt}\left[\frac{20}{3}\left(\frac{1+250/s}{1+500/3s}\right)\right] = \frac{20}{3}A$$

The final value of $i_1(t)$ is

$$i_1(\infty) = \underset{s\to 0}{Lt}\left[sI_1(s)\right] = \underset{s\to 0}{Lt}\left[\frac{20}{3}\left(\frac{s+250}{s+500/3}\right)\right] = 10A$$

The initial value of $i_2(t)$ is

$$i_2(0^+) = \underset{s\to\infty}{Lt}\left[sI_2(s)\right] = \underset{s\to\infty}{Lt}\left[\frac{20}{3}\left(\frac{s}{s+(500/3)}\right)\right] = \underset{s\to\infty}{Lt}\left[\frac{20}{3}\left(\frac{1}{1+(500/3s)}\right)\right] = \frac{20}{3}A$$

The final value of $i_2(t)$ is

$$i_2(\infty) = \underset{s\to 0}{Lt}\left[sI_2(s)\right] = \underset{s\to 0}{Lt}\left[\frac{20}{3}\left(\frac{s}{s+(500/3)}\right)\right] = 0$$

From the above initial and final values, it is clear that at the instant of closing the switch, S, the inductance gives an infinite impedance and hence the currents $i_1(t) = i_2(t) = 100/(10+5) = \frac{20}{3} A$. Then, in the steady state, the inductance becomes a short circuit and hence, $i_1(t) = 10\,A$ and $i_2(t) = 0A$.

Example 3.19 *In the circuit shown in Fig. E3.19(a), find the current i(t) when the switch is at position 2. The switch S is moved from the position 1 to the position 2 at time t = 0. Initially, the switch has been at the position 1 for a long time.*

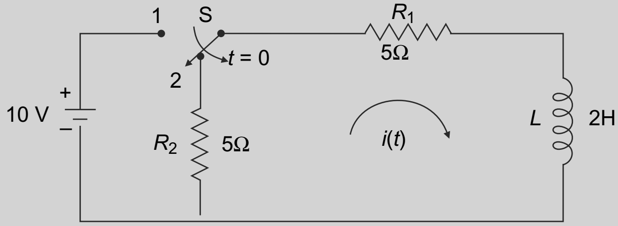


Fig. E3.19(a)

Solution The switch is initially at the position 1 for a long time as shown in Fig. E3.19(b). Hence, the steady-state current is

$$i_s = i(0^-) = \frac{10}{5} = 2A$$

The switch is moved to the position 2 as shown in Fig. E3.19(c). Applying Kirchhoff's voltage law, we get

$$2\frac{di(t)}{dt} + 10i(t) = 0$$

$$\frac{di(t)}{dt} + 5i(t) = 0$$

Taking Laplace transform, we get

$$sI(s) - i(0^-) + 5I(s) = 0$$

where $i(0^-)$ is the initial current flowing through the circuit just after the switch is at the position 2. Since the inductor does not allow sudden changes in currents, $i(0^-)$ is equal to the steady state current when the switch was at the position 1.

Substituting $i(0^-) = 2A$ in the above equation, we get

$$sI(s) - 2 + 5I(s) = 0$$

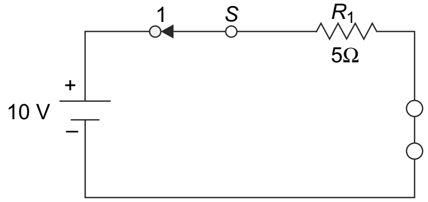


Fig. E3.19(b)

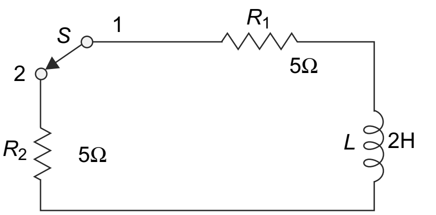


Fig. E3.19(c)

$$I(s) = \frac{2}{s+5}$$

Taking inverse Laplace transform, we have

$$i(t) = 2e^{-5t}\text{ A}$$

Example 3.20 *For the circuit shown in Fig. E3.20, determine the resultant current i(t) when the switch is moved from the position 1 to the position 2 at t = 0. Initially, the switch has been at the position 1 for a long time to get the steady-state values.*

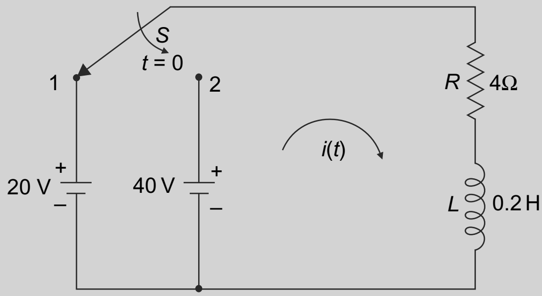


Fig. E3.20

Solution When the switch is moved to the position 2, applying Kirchhoff's voltage law, we get

$$0.2\frac{di(t)}{dt} + 4i(t) = 40$$

Taking Laplace transform, we get

$$0.2\left[sI(s) - i(0^-)\right] + 4I(s) = \frac{40}{s}$$

where $i(0^-)$ is the initial current passing through the circuit just after the switch is at the position 2. Since the inductor does not allow sudden changes in currents, $i(0^-)$ is equal to the steady state current, i_s, when the switch was at the position 1.

Therefore, $\quad i_s = i(0^-) = \frac{20}{4} = 5A$

Substituting $i(0^-) = 5$A in the above equation, we get

$$0.2\left[sI(s) - 5\right] + 4I(s) = \frac{40}{s}$$

$$I(s)[0.2s + 4] = \frac{40}{s} + 1$$

$$I(s) = \frac{5(s+40)}{s(s+20)}$$

Using partial fractions, $I(s)$ can be expanded as

$$I(s) = \frac{5(s+40)}{s(s+20)} = \frac{A_1}{s} + \frac{A_2}{s+20}$$

where $\quad A_1 = sI(s)\big|_{s=0} = \frac{5(s+40)}{(s+20)}\Big|_{s=0} = 10$

and $\quad A_2 = (s+20)I(s)\big|_{s=-20} = \frac{5(s+40)}{s}\Big|_{s=-20} = -5$

Therefore, $$I(s) = \frac{10}{s} - \frac{5}{s+20}$$

Taking inverse Laplace transform, we get

$$i(t) = 5(2 - e^{-20t})\, u(t)$$

Example 3.21 *In the circuit as shown in Fig. E3.21(a) the switch, S, is closed at t = 0 in the position 1 and changed over to the position 2 after one millisecond. Find the time at which the current is zero and reversing its direction. Assume that the change over of switch from the position 1 to 2 takes place in zero time.*

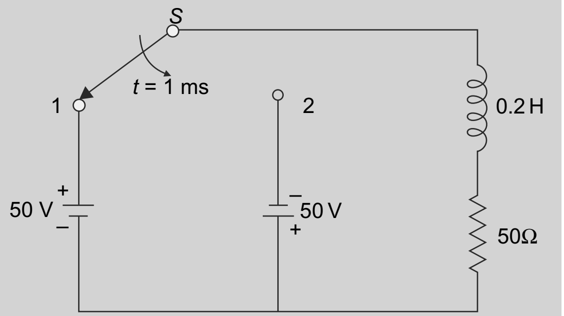


Fig. E3.21(a)

Solution Since the switch, S, is initially opened, the initial current $i(0^-) = 0$ A.

At $t = 0$, the switch is kept at the position 1, as shown in Fig. E3.21(b).

Appling KVL to the circuit in Fig. E3.21(b), we get

$$50i(t) + 0.2\frac{di(t)}{dt} = 50$$

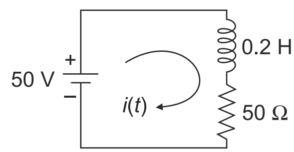


Fig. E3.21(b)

Taking Laplace transform, we get

$$50I(s) + 0.2\left[sI(s) - i\left(0^-\right)\right] = \frac{50}{s}$$

Substituting $i(0^-) = 0$ and simplifying, we get

$$50I(s) + 0.2sI(s) = \frac{50}{s}$$

$$I(s) = \frac{50}{s(0.2s+50)} = \frac{50}{0.2s\left(s + \frac{50}{0.2}\right)} = \frac{250}{s(s+250)}$$

By partial fraction expansion,

$$I(s) = \frac{250}{s(s+250)} = \frac{A_1}{s} + \frac{A_2}{s+250}$$

where $$A_1 = sI(s)\Big|_{s=0} = \left(\frac{250}{s+250}\right)\Bigg|_{s=0} = 1$$

and $$A_2 = (s+250)I(s)\Big|_{s=-250} = \left(\frac{250}{s}\right)\Bigg|_{s=-250} = -1$$

Substituting A_1 and A_2 and in the above equation, we get

$$I(s) = \frac{1}{s} - \frac{1}{s+250}$$

Taking inverse Laplace transform, we get

$$i(t) = 1 - 1e^{-250t}\ \text{A}$$

At $t = 1$ ms,

$$i(1) = 1 - 1e^{-250(1\times 10^{-3})} = 0.2212\ \text{A}$$

This current will be taken as the initial current for the next transition, i.e. when the switch is moved from position 1 to 2 after $t = 1$ ms.

After $t = 1$ ms, the circuit is redrawn as shown in Fig. 3.21(c).

Applying KVL to the circuit in Fig. 3.21(c), we get

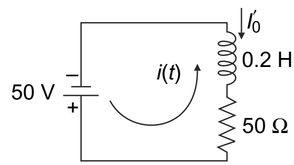


Fig. E3.21(c)

$$50i(t) + 0.2\frac{di(t)}{dt} = 50$$

Taking Laplace transform, we get

$$50I(s) + 0.2[sI(s) - i(1\text{ms})] = \frac{50}{s}$$

Substituting $i(1\text{ ms}) = -0.2212$ A [since $i(1\text{ ms})$ flows in the opposite direction compared to $i(t)$], we get

$$50I(s) + 0.2[sI(s) + 0.2212] = \frac{50}{s}$$

$$50I(s) + 0.2sI(s) + 0.04424 = \frac{50}{s}$$

$$I(s)[50 + 0.2s] = \frac{50}{s} - 0.04424$$

$$I(s)[50 + 0.2s] = \frac{50 - 0.04424s}{s}$$

$$I(s) = \frac{-0.04424s + 50}{s[0.2s + 50]} = \frac{-0.04424s + 50}{0.2s\left[s + \frac{50}{0.2}\right]} = \frac{-0.2212s + 250}{s[s + 250]}$$

By partial fraction expansion,

$$I(s) = \frac{-0.2212s + 250}{s(s+250)} = \frac{A_3}{s} + \frac{A_4}{s+250}$$

where $$A_3 = sI(s)\big|_{s=0} = \frac{-0.2212s + 250}{(s+250)}\bigg|_{s=0} = 1$$

and $$A_4 = (s+250)I(s)\big|_{s=0} = \frac{-0.2212s + 250}{s}\bigg|_{s=-250} = -1.2212$$

Substituting A_1 and A_2, we get

$$I(s) = \frac{1}{s} - \frac{1.2212}{s+250}$$

Taking inverse Laplace transform, we get

$$i(t') = (1 - 1.2212e^{-250t})\, u(t)$$

To find the time at which the current is zero

Substituting $i(t') = 0$,

$$0 = 1 - 1.2212\, e^{-250t'}$$

Hence, $1.2212\, e^{-250t'} = 1$

$$e^{-250t'} = 0.8188$$

Taking natural logarithm on both sides, we get

$$-250t' = -0.1998$$

$$t' = 7.9933 \times 10^{-4}$$

We know that $t' = t - 1 \times 10^{-3}$

Therefore, $t - 1 \times 10^{-3} = 7.9933 \times 10^{-4}$

$$t = 1.7993 \times 10^{-3}\text{ s} = 1.7993\text{ ms}$$

Example 3.22 *In the circuit shown in Fig. 3.22(a), the switch, S, is open for very long time. Suddenly the switch is closed at t = 0. Find the expression for current in the circuit as a function of time. Find the current at 0.5 second after closing the switch.*

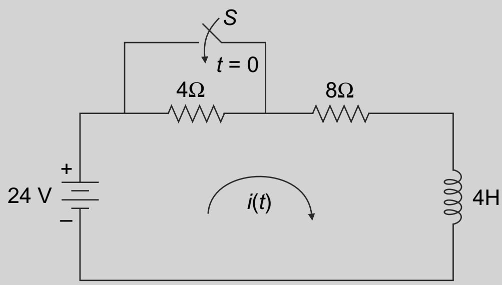


Fig. 3.22(a)

Solution At $t = 0$, the switch, S, is open. Inductor 4 H acts as a short circuit in steady state as shown in Fig. E3.22(b).

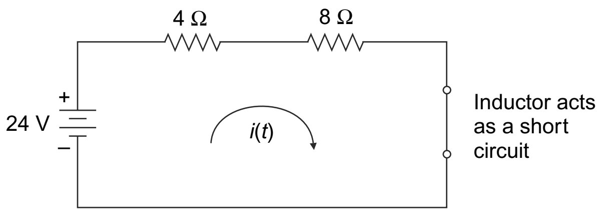


Fig. E3.22(b)

$$i(0^-) = I_0 = \frac{24}{12} = 2A = i(0^+) \qquad (1)$$

For $t \geq 0^+$, switch, S, is closed. Thus, 4 H gets directly short circuit as shown in Fig. E3.22(c). Applying KVL to the circuit, we get

i.e.

$$8i(t) + 4\frac{di(t)}{dt} = 24$$

$$i(t)+\frac{1}{2}\frac{di(t)}{dt}=3$$

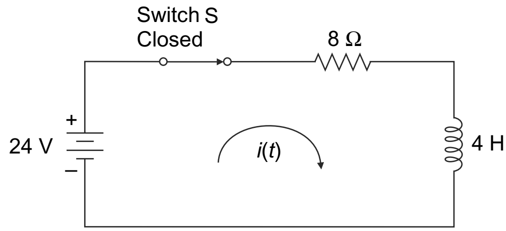


Fig. E3.22(c)

Taking Laplace transform, we get

$$I(s)+\frac{1}{2}L\left[sI(s)-i(0^+)\right]=\frac{3}{s}$$

$$I(s)\left[1+\frac{1}{2}s\right]=\frac{3}{s}$$

$$I(s)=\frac{3}{s(1+0.5s)}==\frac{A_1}{s}+\frac{A_2}{1+0.5s}$$

where $\quad A_1=sI(s)\big|_{s=0}=\frac{3}{1+0.5s}\Big|_{s=0}=3$

and $\quad A_2=(1+0.5s)I(s)\big|_{s=-2}=\frac{3}{s}\Big|_{s=-2}=-\frac{3}{2}$

Substituting A_1 and A_2, we get

$$I(s)=\frac{3}{s}-\frac{3}{2(1+0.5s)}=\frac{3}{s}-\frac{3}{s+2}$$

Taking inverse Laplace transform, we get

$$i(t)=3\left(1-e^{-2t}\right)u(t)$$

Example 3.23 *For the circuit shown in Fig. E3.23(a), determine the current through 10 Ω when the switch, S, is closed at t = 0.*

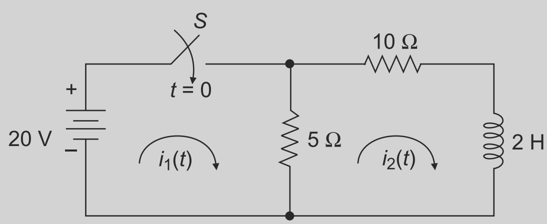


Fig. E3.23(a)

Solution At $t = 0$, the switch is open.

Therefore, $i(0^-) = I_0 = 0\text{ A} = i(0^+)$ (1)

For all $t \geq 0^+$, the switch, S, is closed. Hence the circuit will be as shown in Fig. E3.23(b). But as 5 Ω is connected in shunt with 20 V source, its presence is neglected as the voltage across it is 20 V. Hence, consider the circuit as shown in Fig. E43.23(c).

Applying KVL to the closed path in the circuit as shown in Fig. E3.23(c), we get

$$10i(t)+2\frac{di(t)}{dt}=20$$

i.e. $\quad i(t)+0.2\frac{di(t)}{dt}=2$

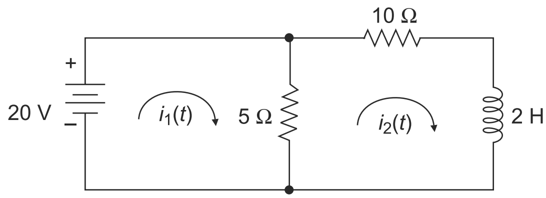


Fig. E3.23(b)

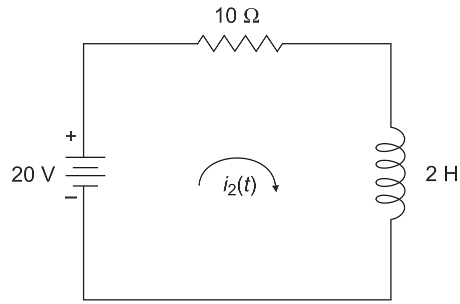


Fig. E3.23(c)

Taking Laplace transform, we get

$$I(s)+0.2\left[sI(s)-i(0^+)\right]=\frac{2}{s}$$

$$I(s)[1+0.2s]=\frac{2}{s}$$

$$I(s)=\frac{2}{s(1+0.2s)}=\frac{10}{s(s+5)}=\frac{A_1}{s}+\frac{A_2}{s+5}$$

where $A_1 = sI(s)\big|_{s=0} = \frac{10}{s+5}\Big|_{s=0} = 2$

and $A_2 = (s+5)I(s)\big|_{s=-5} = \frac{10}{s}\Big|_{s=-5} = -2$

Substituting A_1 and A_2, we get

$$I(s)=\frac{2}{s}-\frac{2}{s+5}$$

Taking inverse Laplace transform, we get

$$i(t)=2\left(1-e^{-5t}\right)u(t)$$

Example 3.24 *A coil having L = 2.4 H and R = 4 Ω is connected to a constant supply source of 100 V. How long does it take for the voltage across the resistance to reach 50 volts?*

Solution The given circuit is as shown in Fig. E3.24. Assuming zero initial conditions, applying KVL, we get

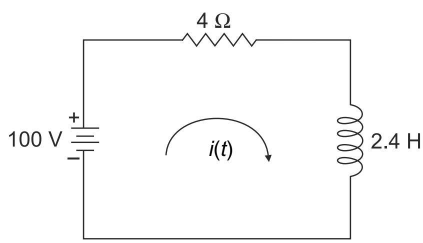


Fig. E3.24

$$4i(t)+2.4\frac{di(t)}{dt}=100$$

$$i(t)+0.6\frac{di(t)}{dt}=25$$

Taking Laplace transform, we get

$$I(s)+0.6\left[sI(s)+i(0^+)\right]=\frac{25}{s}$$

Here, $i(0^+) = 0$

Therefore, $I(s)[1+0.6s]=\frac{25}{s}$

$$I(s)=\frac{25}{s(1+0.6s)}=\frac{A_1}{s}+\frac{A_2}{(1+0.6s)}$$

where $A_1 = sI(s)\big|_{s=0} = \frac{25}{1+0.6s}\Big|_{s=0} = 25$

and $A_2 = (1+0.6s)I(s)\big|_{s=-\frac{1}{0.6}} = \frac{25}{s}\Big|_{s=-\frac{1}{0.6}} = -15$

Substituting A_1 and A_1, we get

$$I(s)=\frac{25}{s}-\frac{15}{1+0.6s}=\frac{25}{s}-\frac{25}{s+1.6667}$$

Taking inverse Laplace transform, we get

$$i(t)=\left(25-25e^{-1.6667t}\right)u(t)$$

$$i(t)=25\left[1-e^{-1.6667t}\right]u(t)$$

As $R = 4\ \Omega$ and $L = 2.4$ H are connected in series, the current indicated by the above equation flows through both the elements. Let at $t = t'$, the voltage across R becomes 50 V. The voltage across R is given by,

$$V_R = Ri(t) = 4\times 25\left(1-e^{-1.6667t}\right) = 100\left(1-e^{-1.6667t}\right)V$$

At $t = t'$, the above equation can be written as

$$V_R = 50 = 100\left(1-e^{-.6667t'}\right)V$$

Solving for t', we get

$$\frac{50}{100}=1-e^{-1.6667t'}$$

Therefore, $e^{-1.6667t'} = 0.5$

Taking natural logarithms on both the sides, we get

$$-1.6667\ t' = -0.69314$$

Therefore, $t' = 0.4158$ second

Example 3.25 *A triangular wave shown in Fig. E3.25(a) is applied as an input to a series RL circuit shown in Fig. E3.25(b). Find the current i(t).*

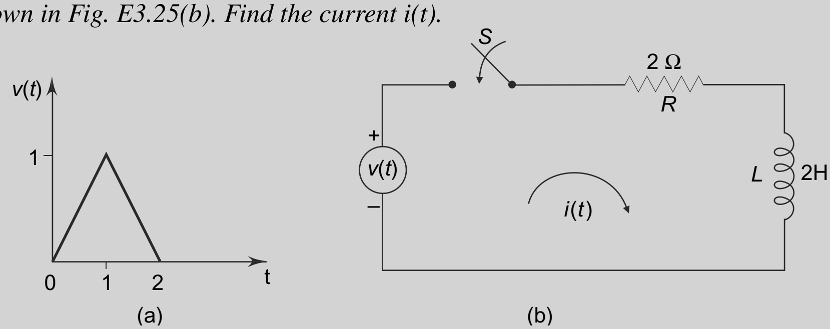


Fig. E3.25

Solution Applying Kirchhoff's voltage law, we get the time domain equation as

$$Ri(t) + L\frac{\mathrm{d}i(t)}{\mathrm{d}t} = v(t)$$

$$2i(t) + 2\frac{\mathrm{d}i(t)}{\mathrm{d}t} = v(t)$$

Taking Laplace transform, we have

$$RI(s) + LsI(s) - Li(0^+) = V(s)$$

$$[R + Ls]I(s) = V(s), \text{ [since } i(0^+) = 0]$$

Substituting $R = 2\Omega$ and $L = 2\ H$, we get

$$(2 + 2s)I(s) = V(s)$$

$$I(s) = \frac{V(s)}{2+2s} = \frac{V(s)}{2(s+1)}$$

The Laplace transform of the given triangular wave is obtained as,

$$V(s) = \frac{1-2e^{-s}+e^{-2s}}{s^2}$$

Therefore, $I(s) = \dfrac{1-2e^{-s}+e^{-2s}}{2s^2(s+1)}$

By partial fraction expansion,

$$F(s) = \frac{1}{s^2(s+1)} = \frac{A_0}{s^2} + \frac{A_1}{s} + \frac{A_2}{(s+1)}$$

$$A_0 = F(s)s^2\Big|_{s=0} = \frac{1}{s+1}\Big|_{s=0} = 1$$

$$A_1 = \frac{1}{\mathrm{d}s}\frac{1}{s+1}\Big|_{s=0} = -\frac{1}{(s+1)^2}\Big|_{s=0} = -1$$

$$A_2 = (s+1)F(s)\Big|_{s=-1} = -\frac{1}{s^2}\Big|_{s=-1} = 1$$

Therefore, $\dfrac{1}{s^2(s+1)} = \dfrac{1}{s^2} - \dfrac{1}{s} + \dfrac{1}{(s+1)}$

$$\mathcal{L}^{-1}\left\{\frac{1}{s^2(s+1)}\right\} = t - 1 + e^{-t}$$

Therefore, $i(t) = \mathcal{L}^{-1}[I(s)] = \mathcal{L}^{-1}\left\{\dfrac{1-2e^{-s}+e^{-2s}}{2s^2(s+1)}\right\}$

$$= \frac{1}{2}\left[t-1+e^{-t}\right]\left[u(t)-2u(t-1)+u(t-2)\right]$$

3.12.2 Step and Impulse Responses of Series RC Circuit

Step Response

For the d.c. response, the input excitation is $V_i u(t)$. Applying KVL to the series *RC* circuit as shown in Fig. 3.5, we get

$$\frac{1}{C}\int_{-\infty}^{t} i(t)\mathrm{d}t + Ri(t) = V_i u(t) \tag{3.28}$$

This may be written as

$$Ri(t) + \frac{1}{C}\int_{-\infty}^{0} i(t)\,\mathrm{d}t + \frac{1}{C}\int_{0}^{t} i(t)\,\mathrm{d}t = V_i u(t)$$

Taking Laplace transform, the above equation becomes

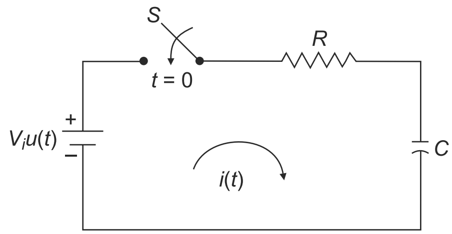


Fig. 3.5 *Serial RC Circuit*

$$RI(s) + \frac{1}{C}\left[\frac{I(s)}{s}\right] + \frac{1}{C}\mathcal{L}\left[q\left(0^+\right)\right] = \frac{V_i}{s}$$

$$RI(s) + \frac{1}{C}\left[\frac{I(s)}{s} + \frac{q\left(0^+\right)}{s}\right] = \frac{V_i}{s}$$

Since the capacitor is initially uncharged, $q(0^+) = 0$.

Therefore, $$I(s)\left[\frac{1}{sC} + R\right] = \frac{V_i}{s}$$

Hence, $$I(s) = \frac{V_i / R}{s + \frac{1}{RC}}$$

Taking inverse Laplace transform, we get

$$i(t) = \frac{V_i}{R} e^{-\frac{t}{RC}} \tag{3.29}$$

In an *RC* circuit, the time constant is *RC*, which is the time required for the voltage to reach from its initial value of zero to final value of source voltage, V_i.

The above equation shows that there is an exponential decrease in current as time *t* increases as shown in Fig. 3.6. As *t* becomes infinite, the current $i(t)$ approaches zero.

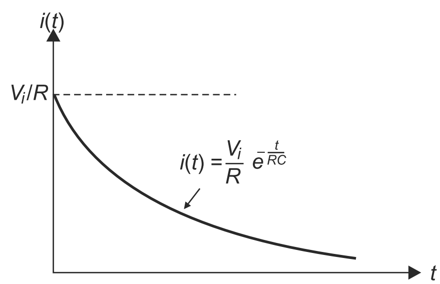


Fig. 3.6 *Exponential Decrease in Current*

Impulse Response

For the impulse response, the input excitation is $x(t) = \delta(t)$. Hence, the integral equation becomes

$$\frac{1}{C}\int_{-\infty}^{t} i(t)\mathrm{d}t + Ri(t) = \delta(t)$$

$$\frac{1}{C}\int_{-\infty}^{0} i(t)\,\mathrm{d}t + \frac{1}{C}\int_{0}^{t} i(t)\,\mathrm{d}t + Ri(t) = \delta(t)$$

Taking Laplace transform, the above equation becomes

$$\frac{1}{C}\left[\frac{I(s)}{s}\right]+\frac{1}{C}\mathcal{L}\left[q\left(0^+\right)\right]RI(s)=1$$

$$\frac{1}{C}\left[\frac{I(s)}{s}+\frac{q\left(0^+\right)}{s}\right]+RI(s)=1$$

Since $q(0^+) = 0$, $I(s)\left[\frac{1}{Cs}+R\right]=1$

Therefore,

$$I(s)=\frac{1}{R\left(1+\frac{1}{RCs}\right)}$$

$$=\frac{s}{R\left(s+\frac{1}{RC}\right)}=\frac{1}{R}\left[\frac{\left(s+\frac{1}{RC}\right)-\frac{1}{RC}}{s+\frac{1}{RC}}\right]=\frac{1}{R}\left[1-\frac{1}{RC}\frac{1}{(s+1/RC)}\right]$$

Its inverse Laplace transform is

$$i(t)=\frac{1}{R}\left[\delta(t)-\frac{1}{RC}\times e^{-t/RC}u(t)\right]$$

Example 3.26 *In the circuit of Fig. E3.26, the switch, S, is closed at t = 0 and there is no initial charge on either of the capacitors. Find the resulting current i(t). Also, find the initial and final values of the current.*

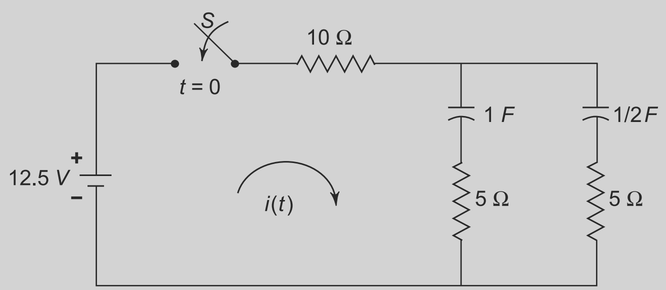


Fig. E3.26

Solution The equivalent impedance of the network in the s-domain is

$$Z(s)=10+\frac{(5+1/s)(5+2/s)}{10+1/s+2/s}=\frac{125s^2+45s+2}{s(10s+3)}$$

Therefore, $$I(s)=\frac{V(s)}{Z(s)}=\frac{12.5}{s}\times\frac{s(10s+3)}{125s^2+45s+2}=\frac{(s+0.3)}{(s+0.308)(s+0.052)}$$

Using partial fractions, $I(s)$ can be expanded as

$$I(s)=\frac{1/32}{s+0.308}+\frac{31/32}{s+0.052}$$

Taking inverse Laplace transform, we obtain

$$i(t)=\left(\frac{1}{32}e^{-0.308t}+\frac{31}{32}e^{-0.052t}\right)u(t)$$

The initial value of the current is

$$i(0^+)=\underset{t\to 0}{Lt}\ i(t)=\underset{t\to 0}{Lt}\left(\frac{1}{32}e^{-0.308t}+\frac{31}{32}e^{-0.052t}\right)=1\text{ A}$$

or
$$i(0^+)=\underset{s\to \infty}{Lt}\left[sI(s)\right]=\underset{s\to \infty}{Lt}\left[\frac{1}{32}\left(\frac{s}{s+0.308}\right)+\frac{31}{32}\left(\frac{s}{s+0.052}\right)\right]$$

$$=\underset{s\to \infty}{Lt}\left[\frac{1}{32}\left(\frac{1}{1+0.308/s}\right)+\frac{31}{32}\left(\frac{1}{1+0.052/s}\right)\right]=1A$$

The final value of the current is

$$i(\infty)=\underset{t\to \infty}{Lt}\ i(t)=\underset{t\to \infty}{Lt}\left(\frac{1}{32}e^{-0.308t}+\frac{31}{32}e^{-0.052t}\right)=0$$

or
$$i(\infty)=\underset{s\to 0}{Lt}\left[sI(s)\right]=\underset{s\to 0}{Lt}\left[\frac{1}{32}\left(\frac{s}{s+0.308}\right)+\frac{31}{32}\left(\frac{s}{s+0.052}\right)\right]=0$$

In the given network, initially the total circuit resistance is $R = 10 + 2(5)/10 = 12.5\ \Omega$ and hence $i(0^+) = 12.5/12.5 = 1$ A. Then, in the steady state, both capacitors are charged to 12.5 V and the current becomes zero.

Example 3.27 *In the circuit shown in Fig. E3.27, determine the voltage, $v_c(t)$ across the capacitor for $t \geq 0$, if $v_c(0^-) = 10$ V.*

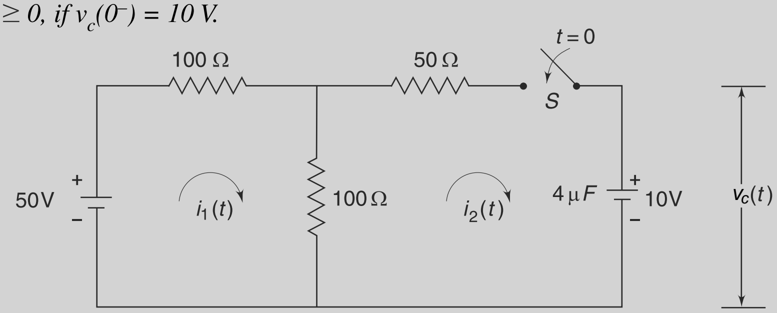


Fig. E3.27

Solution For Loop 1,

$$200i_1(t) - 100i_2(t) = 50$$

i.e. $$4i_1(t) - 2i_2(t) = 1 \quad (1)$$

For Loop 2,

$$150i_2(t)-100i_1(t)+\frac{1}{4\times 10^{-6}}\int_0^t i_2(t)\,dt=-10$$

i.e. $$-10i_1(t)+15i_2(t)+\frac{1}{4\times 10^{-5}}\int_0^t i_2(t)\,dt=-1 \qquad (2)$$

Taking Laplace transform of Eq. (1), we get

$$4I_1(s)-2I_2(s)=\frac{1}{s}$$

$$I_1(s)=\frac{1}{4}\left[\frac{1}{s}+2I_2(s)\right] \qquad (3)$$

Taking Laplace transform of Eq. (2), we get

$$-10I_1(s)+I_2(s)\left[15+\frac{25000}{s}\right]=-\frac{1}{s} \qquad (4)$$

Substituting Eq. (3) in Eq. (4), we get

$$10\left[\frac{1}{4}\left(\frac{1}{s}+2I_2(s)\right)\right]-I_2(s)\left[15+\frac{25000}{s}\right]=\frac{1}{s}$$

$$I_2(s)\left[10+\frac{25000}{s}\right]=\frac{3}{2s}$$

$$I_2(s)=\frac{3}{20s}\left[\frac{s}{s+2500}\right]=\frac{0.15}{s+2500}$$

Taking inverse Laplace transform, we get

$$i_2(t)=0.15e^{-2500t}\,\text{A}$$

$$v_c(t)=\frac{1}{4\times 10^{-6}}\int_0^t i_2(t)dt+V_c(0^-)=\frac{0.15}{4\times 10^{-6}}\left[\frac{e^{-2500t}}{-2500}\right]_0^t+10$$

$$=-15\left[e^{-2500t}-1\right]+10$$

$$v_c(t)=10+15\left[1-e^{-2500t}\right]u(t)$$

Example 3.28 *A rectangular voltage pulse of height A and duration T, seconds is applied to a series RC combination at t = 0, as shown in Fig. E3.28 (a) Determine the current i(t) in the capacitor as a function of time. Assume the capacitor to be initially uncharge (b) Determine i(t) when A = 0 V, T = 10 μs, R = 100 Ω and C = 0.1 μF.*

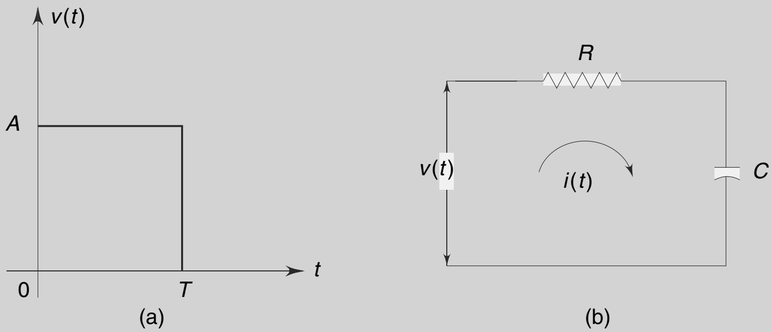


Fig. E3.28

Solution

(a) The input voltage can be written as the combination of two steps, i.e.

$$v(t) = A[u(t) - u(t - T)]$$

Applying Kirchhoff's voltage law to the circuit, we get

$$Ri(t)+\frac{1}{C}\int_{-\infty}^{t} i(t)\,\mathrm{d}t = A\left[u(t)-u(t-T)\right]$$

Taking Laplace transforms on both sides, we get

$$RI(t)+\frac{1}{C}\left[\frac{I(s)}{s}+\frac{q(0)}{s}\right]=\frac{A}{s}\left(1-e^{-sT}\right)$$

Since the initial charge on the capacitor is zero, $q(0) = 0$

Therefore, $I(s)\left[R+\frac{1}{Cs}\right]=\frac{A}{s}\left(1-e^{-sT}\right)$

or

$$I(s)=\frac{A\left(1-e^{-sT}\right)}{R\left(s+\frac{1}{RC}\right)}$$

$$=\frac{A}{R}\left[\frac{1}{(s+1/RC)}-\frac{e^{-sT}}{s+1/RC}\right]$$

Taking inverse transform on both sides, we get

$$i(t)=\frac{A}{R}\left\{u(t)e^{-t/RC}-u(t-T)e^{-(1/RC)(t-T)}\right\}$$

(b) Substituting the given values of $A = 10$ volts, $T = 10^{-5}$ seconds, $R = 100\,\Omega$ and $C = 10^{-7}F$ in the above result, we get

$$i(t)=\frac{1}{10}\left[u(t)e^{-10^{-5}t}-u\left(t-10^{-5}\right)e^{-10^{5(t-10^{-5})}}\right]$$

Example 3.29 *A periodic waveform shown in Fig. E3.29(a) is applied to the RC network of Fig. E3.29(b). Find the transient current and periodic or steady-state current.*

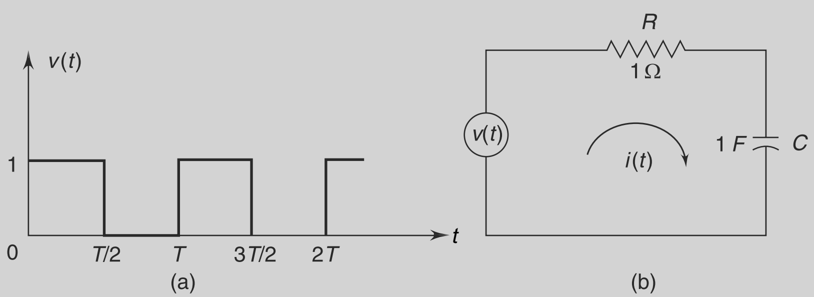


Fig. E3.29

Solution The function for the first period of the given waveform is

$$v(t)=\begin{cases}1 & \text{for } 0<t\le T/2\\ 0 & \text{for } T/2<t\le T\end{cases}$$

$$V(s)=\frac{1}{1-e^{-sT}}\int_0^T f(t)\,\mathrm{d}t$$

$$=\frac{1}{1-e^{-sT}}\left[\int_0^{T/2}1\cdot e^{-st}\,\mathrm{d}t+\int_{T/2}^{T}0\cdot e^{-st}\,\mathrm{d}t\right]=\frac{1}{1-e^{-sT}}\left[\frac{e^{-st}}{-s}\right]_0^{T/2}$$

$$=\frac{1}{1-e^{-sT}}\left[\frac{1-e^{-sT/2}}{s}\right]$$

Alternate method to find Laplace transform of the given periodic waveform

The input periodic pulse train can be represented as

$$v(t) = u(t) - u(t - T/2) + u(t - T) - u(t - 3T/2) + u(t - 2T) - u(t - 5T/2) + \ldots$$

Its Laplace transform is

$$V(s)=\frac{1}{s}\left[1-e^{-sT/2}+e^{-sT}-e^{-3sT/2}+e^{-2sT}-e^{-5sT/2}+\ldots\right]$$

$$=\frac{1}{s}\left[1-e^{-sT/2}+e^{-sT}\left(1-e^{-sT/2}\right)+e^{-2sT}\left(1-e^{-sT/2}\right)\right]$$

$$=\frac{1}{s}\left(1-e^{-sT/2}\right)\left(1+e^{-sT}+e^{-2sT}+\cdots\right)$$

$$=\frac{1}{s}\frac{\left(1-e^{-sT/2}\right)}{\left(1-e^{-sT}\right)}$$

From Fig. E3.29(b), $I(s)=\dfrac{V(s)}{R+1/Cs}=\dfrac{V(s)}{1+1/s}=\dfrac{sV(s)}{s+1}$

Substituting for $V(s)$, we get

$$I(s)=\frac{1-e^{-sT/2}}{\left(1-e^{-sT}\right)(s+1)}=\frac{A_1}{(s+1)}+\frac{A_2}{1-e^{-sT}}$$

$$=I_t(s)+\frac{I_p(s)}{1-e^{-sT}}$$

where the transient part $I_t(s)$ depends upon the network poles and the periodic or steady-state current $I_p(s)$ has a denominator $(1 - e^{-sT})$.

$$A_1=I(s)(s+1)\big|_{s=-1}=\frac{1-e^{T/2}}{1-e^{T}}$$

$$A_2=\frac{1-e^{-sT/2}}{(s+1)}-\frac{1-e^{T/2}}{1-e^{T}}\cdot\frac{1-e^{-sT}}{(s+1)}$$

Hence, the transient part is $I_t(s) = \frac{1-e^{T/2}}{1-e^T} \cdot \frac{1}{(s+1)}$

Taking inverse Laplace transform, we get the transient current

$$i_t(t) = \frac{1-e^{T/2}}{1-e^T} e^{-t} u(t)$$

The periodic or steady-state current is

$$I_p(s) = \frac{1-e^{-sT/2}}{(s+1)} - \frac{1-e^{T/2}}{1-e^T} \frac{1-e^{-sT}}{(s+1)}$$

$$= \frac{1}{(s+1)} - \frac{e^{-sT/2}}{(s+1)} - \frac{1-e^{T/2}}{1-e^T}\left[\frac{1}{(s+1)} - \frac{e^{-sT}}{s+1}\right]$$

Therefore,

$$i_p(t) = e^{-t}u(t) - e^{-(t-T/2)}u(t-T/2) - \frac{1-e^{T/2}}{1-e^T}\left[e^{-t}u(t) - e^{-(t-T)}u(t-T)\right]$$

3.12.3 Step Response of Series *RLC* Circuit

The series *RLC* circuit is shown in Fig. 3.7. Let the switch, *S*, be closed at time $t = 0$. For d.c. response, the input excitation is $V_i u(t)$. Applying Kirchhoff's voltage law to the circuit, we get

$$L\frac{\mathrm{d}i(t)}{\mathrm{d}t} + Ri(t) + \frac{1}{C}\int_{-\infty}^{t} i(t)\,\mathrm{d}t = V_i u(t) \tag{3.30}$$

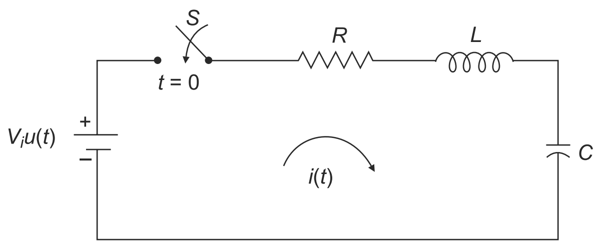


Fig. 3.7 *Series RLC Circuit*

This may be written as

$$L\frac{\mathrm{d}i(t)}{\mathrm{d}t} + Ri(t) + \frac{1}{C}\int_{-\infty}^{0} i(t)\,\mathrm{d}t + \frac{1}{C}\int_{0}^{t} i(t)\,\mathrm{d}t = V_i u(t)$$

Taking Laplace transform, the equation becomes

$$L\left[sI(s) - i(0^-)\right] + RI(s) + \frac{1}{C}\mathcal{L}\left[q(0^-)\right] + \frac{1}{C}\frac{I(s)}{s} = \frac{V_i}{s}$$

or

$$L\left[sI(s) - i(0^-)\right] + RI(s) + \frac{1}{C}\frac{q(0^-)}{s} + \frac{1}{C}\frac{I(s)}{s} = \frac{V_i}{s}$$

Because of the presence of the inductor L, $i(0^-) = 0$. Also, $q(0^-)$ is the charge on the capacitor C at $t = 0^-$. If the capacitor is initially uncharged, then $q(0^-) = 0$. Substituting these two initial conditions, we get

$$LsI(s) + RI(s) + \frac{I(s)}{sC} = \frac{V_i}{s}$$

$$I(s)\left[Ls + R + \frac{1}{sC}\right] = \frac{V_i}{s}$$

Therefore, $$I(s) = \frac{V_i}{s^2L + sR + \frac{1}{C}} = \frac{V_i}{L} \times \frac{1}{\left(s^2 + \frac{R}{L}s + \frac{1}{LC}\right)} = \frac{V_i}{L} \times \frac{1}{(s-p_1)(s-p_2)}$$

where $p_1, p_2 = -\frac{R}{2L} \pm \sqrt{\left(\frac{R}{2L}\right)^2 - \frac{1}{LC}}$

Let $\alpha = -\frac{R}{2L}$ and $\beta = \sqrt{\left(\frac{R}{2L}\right)^2 - \frac{1}{LC}}$

Therefore, $s^2 + \frac{R}{L}s + \frac{1}{LC} = [s-(\alpha+\beta)][s-(\alpha-\beta)]$

Hence, $$I(s) = \frac{V_i}{L} \times \frac{1}{[s-(\alpha+\beta)][s-(\alpha-\beta)]} \tag{3.31}$$

Using partial fractions, $I(s)$ can be expanded. Then by taking inverse Laplace transform, $i(t)$ can be evaluated.

There will be three possibilities to evaluate $i(t)$, depending upon the value of determinant.

Case (i) Overdamped Response

When $\left(\frac{R}{2L}\right)^2 > \frac{1}{LC}$, the discriminant will be positive and hence, the roots are real and unequal.

Therefore, $$I(s) = \frac{A_1}{(s-(\alpha+\beta))} + \frac{A_2}{(s-(\alpha-\beta))}$$

Hence, $$i(t) = A_1 e^{(\alpha+\beta)t} + A_2 e^{(\alpha-\beta)t}$$
$$= e^{\alpha t}\left[A_1 e^{\beta t} + A_2 e^{-\beta t}\right]$$

Case (ii) Critically Damped Response

When $\left(\frac{R}{2L}\right)^2 = \frac{1}{LC}$, the discriminant will be zero and hence, the roots are real and equal.

Therefore, $$I(s) = \frac{A_1}{(s-\alpha)^2} + \frac{A_2}{(s-\alpha)}$$

Hence, $$i(t) = A_1 t e^{\alpha t} + A_2 e^{\alpha t}$$
$$= e^{\alpha t}[A_1 t + A_2]$$

Case (iii) Underdamped Response

When $\left(\frac{R}{2L}\right)^2 < \frac{1}{LC}$, the discriminant will be imaginary and hence, the roots are complex conjugate.

We know that $\beta = \sqrt{\frac{1}{LC} - \left(\frac{R}{2L}\right)^2}$

Then the roots are $(\alpha + j\beta)$ and $(\alpha - j\beta)$

Therefore, $$I(s) = \frac{A_1}{(s-(\alpha+j\beta))} + \frac{A_2}{(s-(\alpha-j\beta))}$$

Hence, $$i(t) = A_1 e^{(\alpha+j\beta)t} + A_2 e^{(\alpha-j\beta)t}$$
$$= e^{\alpha t}\left[A_1 e^{j\beta t} + A_2 e^{-j\beta t}\right]$$

The above expression may be expressed as
$$i(t) = e^{\alpha t}\left[A\cos\beta t + B\sin\beta t\right]$$

The result shows that the current is oscillatory and it decays in a short time since α is negative.

Example 3.30 *Find the current i(t) resulting from the voltage source v(t) in the series RLC circuit shown in Fig. E3.30. Assume that there is an initial current i_L flowing in the inductor and initial voltage v_c across the capacitor when the switch is closed at t = 0.*

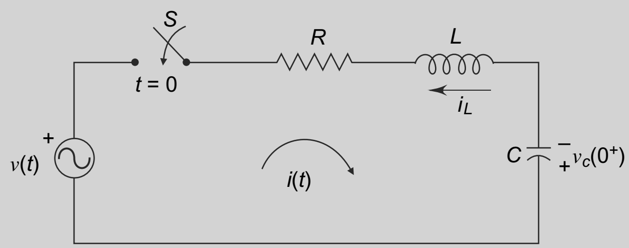


Fig. E3.30

Solution The differential equation for the series *RLC* circuit is obtained by using Kirchhoff's voltage law as

$$v(t) = Ri(t) + L\frac{\mathrm{d}i(t)}{\mathrm{d}t} + \frac{1}{C}\int_{-\infty}^{t} i(t)\,\mathrm{d}t$$

Taking the Laplace transform, we get

$$V(s) = RI(s) + L\left[sI(s) + i(0^+)\right] + \frac{1}{C}\left[\frac{I(s)}{s} - \frac{q(0^+)}{s}\right]$$
$$= RI(s) + L\left[sI(s) - i(0^+)\right] + \frac{I(s)}{sC} + \frac{v_c(0^+)}{s}$$

where $v_c(0^+) = q(0^+)/C$ is the voltage across the capacitor after the switch is closed at $t = 0^+$. The polarity of the initial conditions depends on the assumed direction for $i(t)$ with respect to the flow of the current in L and the voltage across the capacitor C.

$$I(s)\left[R+sL+\frac{1}{sC}\right]=V(s)+\frac{v_c(0^+)}{s}-Li(0^+)$$

Therefore, $$I(s)=\frac{V(s)-Li(0^+)+\dfrac{v_c(0^+)}{s}}{R+sL+\dfrac{1}{sC}}$$

Given the parameter values, we can take the inverse Laplace transform to obtain $i(t)$.

Example 3.31 *In a series RLC circuit shown in Fig. 3.31, there is no initial charge on the capacitor. If the switch is closed at t = 0, find the resulting current.*

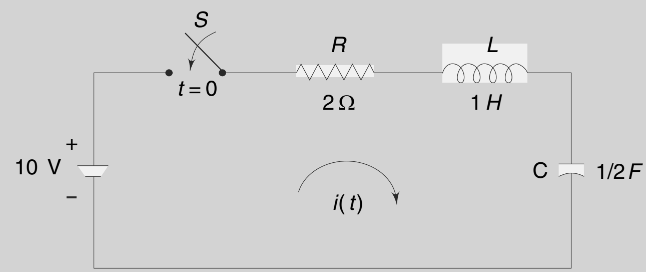


Fig. E3.31

Solution The time-domain equation of the given circuit is

$$Ri(t)+L\frac{di(t)}{dt}+\frac{1}{C}\int_0^t i(t)\,dt=10$$

Here, $i(0^+) = 0$, and $v(0^+) = 0$

Applying Laplace transform, we get

$$RI(s)+LsI(s)+\frac{1}{sC}I(s)=\frac{10}{s}$$

$$2I(s)+1sI(s)+\frac{1}{0.5s}I(s)=\frac{10}{s}$$

$$I(s)\left(2s+s^2+2\right)=10$$

Therefore, $$I(s)=\frac{10}{s^2+2s+2}=\frac{10}{(s+1)^2+1}$$

Taking inverse Laplace transform, we get

$$i(t)=10e^{-t}\sin t\ A$$

Example 3.32 *A step voltage 100 u(t) is applied to a series RLC circuit as shown in Fig. 3.32. The initial current through the inductor is zero and initial voltage across capacitor is 50 V. Determine the expression for the current in the circuit.*

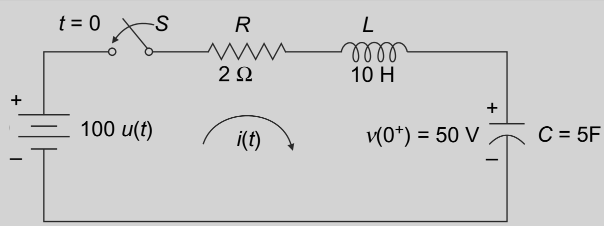


Fig. E3.32

Solution The initial current $i(0^+) = 0$

The initial voltage $v\left(0^+\right) = \frac{1}{C}\int_{-\infty}^{0} i(t)\,dt = \frac{q\left(0^+\right)}{C} = 50V$

Applying Kirchhoff's voltage law to the circuit, we get

$$2i(t) + 10\frac{di(t)}{dt} + 50 + \frac{1}{5}\int_0^t i(t)\,dt = 100$$

$$2i(t) + 10\frac{di(t)}{dt} + \frac{1}{5}\int_0^t i(t)\,dt = 50$$

Applying Laplace transform and substituting $i(0^+) = 0$, we get

$$\left[2 + 10s + \frac{1}{5s}\right]I(s) = \frac{50}{s}$$

$$\left[10s^2 + 2s + 0.2\right]I(s) = 50$$

$$I(s) = \frac{5}{s^2 + 0.2s + 0.02} = \frac{5}{(s+0.1)^2 + (0.1)^2}$$

$$= \frac{5(0.1)}{0.1\times\left[(s+0.1)^2 + \left(0.1^2\right)\right]}, \qquad \left[\text{since } \mathcal{L}\left(e^{-at}\sin\omega t\right) = \frac{\omega}{(s+a)^2 + \omega^2}\right]$$

$$= 50\left[\frac{0.1}{(s+0.1)^2 + (0.1)^2}\right]$$

Taking inverse Laplace transform, we get

$$i(t) = 50e^{-0.1t}\sin 0.1t \text{ A}$$

Example 3.33 *In a source free RLC series circuit shown in Fig. 3.33, the initial voltage across the capacitor is 10V and the initial current through inductor is zero. Determine i(t).*

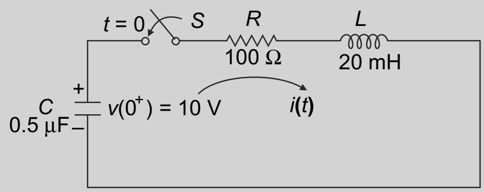


Fig. E3.33

Solution The initial current $i(0^+) = 0$

The initial voltage $V\left(0^+\right) = \frac{1}{C}\int_{-\infty}^{0} i(t)dt = \frac{q\left(0^+\right)}{C} = 10V$

Applying Kirchhoff's voltage law to the circuit, we get

$$100i(t) + 20\times10^{-3}\frac{di(t)}{dt} + v\left(0^+\right) + \frac{1}{0.5\times10^{-6}}\int_0^t i(t)\,dt = 0$$

$$100i(t)+0.02\frac{di(t)}{dt}+10+2\times10^6\int_0^t i(t)dt=0$$

$$100i(t)+0.02\frac{di}{dt}+2\times10^6\int_0^t i(t)dt=-10$$

Applying Laplace transform and substituting $i(0^+) = 0$, we get

$$100I(s)+0.02sI(s)+\frac{2\times10^6}{s}I(s)=\frac{-10}{s}$$

$$\left[100+0.02s+\frac{2\times10^6}{s}\right]I(s)=\frac{-10}{s}$$

$$\left[s^2+5000s+10^8\right]I(s)=-500$$

Therefore,
$$I(s)=\frac{-500}{s^2+5000s+10^8}=\frac{-500}{(s+2500)^2+(93.75\times10^6)}$$

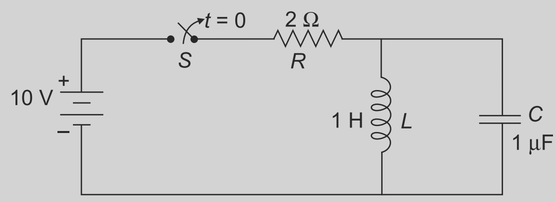
$$=\frac{-500\times9682}{9682\left[(s+2500)^2+(9682)^2\right]}$$

$$=\frac{-0.0516(9682)}{(s+2500)^2+(9682)^2},\quad\left[\text{since } \mathcal{L}\left(e^{-at}\sin\omega t\right)=\frac{\omega}{(s+a)^2+\omega^2}\right]$$

Taking inverse Laplace transform, we get

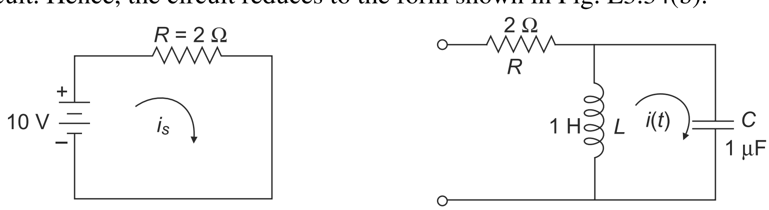
$$i(t)=-0.0516e^{-2500t}\sin 9682t\,u(t)$$

Example 3.34 *When the switch, S, is closed in the circuit shown in Fig. E3.34(a), steady-state condition is reached. Now at time t= 0, the switch, S, is opened. Obtain the expression for current $i_L(t)$ through the inductor.*



(a)

Solution Under steady-state condition, capacitor acts as an open circuit while the inductor L acts as a short circuit. Hence, the circuit reduces to the form shown in Fig. E3.34(b).



(b) (c)

Fig. E3.34

Hence, the steady-state current, $i_s = i(0^+) = \frac{10}{2} = 5$ A . The battery voltage of 10 V drops across the resistor R. The voltage across the capacitor C is zero and the current through the inductor is 5 A.

When the switch, S, is opened, the circuit reduces to the form shown in Fig. 3.34(c). The current through the inductor L continues to flow. But now the current $i(t)$ flows through the capacitor C. Applying KVL to the circuit, we obtain

$$L\frac{\mathrm{d}i(t)}{\mathrm{d}t} + \frac{1}{C}\int i(t)\,\mathrm{d}t = 0 \tag{1}$$

Taking Laplace transform, we get

$$L\left[sI(s) - i(0^+)\right] + \frac{1}{C}\left[\frac{I(s)}{s} - \frac{q(0^+)}{s}\right] = 0$$

$$1 \times sI(s) - 1 \times 5 + \frac{1}{1 \times 10^{-6}}\left(\frac{I(s)}{s} - 0\right) = 0$$

$$I(s)\left(s + \frac{10^6}{s}\right) = 5$$

$$I(s) = \frac{5s}{s^2 + (10^3)^2}$$

Taking inverse Laplace transform, we get

$$i(t) = 5\cos 10^3 t\, u(t)$$

Example 3.35 *A system has a transfer function given by* $H(s) = \frac{1}{(s+1)(s^2+s+1)}$. *Find the response of the system when the excitation is* $x(t) = (1 + e^{-3t} - e^{-t})u(t)$.

Solution The given input function is $x(t) = (1 + e^{-3t} - e^{-t})\, u(t)$

Taking Laplace transform, we get

$$X(s) = \frac{1}{s} + \frac{1}{s+3} - \frac{1}{s+1}$$

$$H(s) = \frac{1}{(s+1)(s^2+s+1)} = \frac{A_1}{s+1} + \frac{A_2 s + A_3}{s^2+s+1}$$

$$A_1(s^2+s+1) + (A_2 s + A_3)(s+1) = 1$$

Substituting $s = -1$, we get $A_1 = 1$

Comparing the coefficients of s^2, we get

$$A_1 + A_2 = 0$$
$$A_2 = -1$$

Comparing the coefficients of s, we get

$$A_1 + A_2 + A_3 = 0$$
$$1 - 1 + A_3 = 0$$
$$A_3 = 0$$

Therefore, $H(s) = \frac{1}{(s+1)(s^2+s+1)} = \frac{1}{s+1} - \frac{s}{s^2+s+1}$

We know that $Y(s) = H(s)\,X(s)$

$$Y(s) = \left(\frac{1}{s+1} - \frac{s}{s^2+s+1}\right)\left(\frac{1}{s} + \frac{1}{s+3} - \frac{1}{s+1}\right)$$

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

By using partial fraction expansions, the above functions can be expanded as

(i) $\frac{1}{s(s+1)} = \frac{1}{s} - \frac{1}{s+1}$

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

(iii) $\frac{s}{(s+3)(s^2+s+1)} = \frac{-3/7}{s+3} + \frac{3/7s+1/7}{s^2+s+1}$

(iv) $\frac{s}{(s+1)(s^2+s+1)} = -\frac{1}{s+1} + \frac{s+1}{s^2+s+1}$

Therefore, $Y(s) = \frac{1}{s} - \frac{1}{s+1} + \frac{1}{2(s+1)} - \frac{1}{2(s+3)} - \frac{1}{(s+1)^2} - \frac{1}{s^2+s+1} + \frac{3/7}{s+3} - \frac{3/7s+1/7}{s^2+s+1}$

$$- \frac{1}{s+1} + \frac{s+1}{(s^2+s+1)}$$

where $\mathcal{L}^{-1}\left\{\frac{s}{s^2+s+1}\right\} = \mathcal{L}^{-1}\left\{\frac{s+\frac{1}{2}-\frac{1}{2}}{\left(s+\frac{1}{2}\right)^2+\frac{3}{4}}\right\} = \mathcal{L}^{-1}\left\{\frac{s+\frac{1}{2}}{\left(s+\frac{1}{2}\right)^2+\frac{3}{4}}\right\} - \frac{1}{\sqrt{3}}\mathcal{L}^{-1}\left\{\frac{\sqrt{3}/2}{\left(s+\frac{1}{2}\right)^2+\frac{3}{4}}\right\}$

$$= e^{-t/2}\cos\frac{\sqrt{3}}{2}t - \frac{1}{\sqrt{3}}e^{-t/2}\sin\frac{\sqrt{3}}{2}t$$

$$\mathcal{L}^{-1}\left\{\frac{1}{s^2+s+1}\right\} = \frac{2}{\sqrt{3}}\mathcal{L}^{-1}\left\{\frac{\frac{\sqrt{3}}{2}}{\left(s+\frac{1}{2}\right)^2+\frac{3}{4}}\right\} = \frac{2}{\sqrt{3}}e^{-t/2}\sin\frac{\sqrt{3}}{2}t$$

Taking inverse Laplace transform of $Y(s)$, we get

$$y(t) = 1 - e^{-t} + \frac{1}{2}e^{-t} - \frac{1}{2}e^{-3t} - te^{-t} - \frac{2}{\sqrt{3}}e^{-t/2}\sin\frac{\sqrt{3}}{2}t + \frac{3}{7}e^{-3t} - \frac{3}{7}\left(e^{-t/2}\cos\frac{\sqrt{3}}{2}t - \frac{1}{\sqrt{3}}e^{-t/2}\sin\frac{\sqrt{3}}{2}t\right)$$

$$- \frac{1}{7}\times\frac{2}{\sqrt{3}}e^{-t/2}\sin\frac{\sqrt{3}}{2}t - e^{-t} + e^{-t/2}\cos\frac{\sqrt{3}}{2}t - \frac{1}{\sqrt{3}}e^{-t/2}\sin\frac{\sqrt{3}}{2}t + \frac{2}{\sqrt{3}}e^{-t/2}\sin\frac{\sqrt{3}}{2}t$$

Therefore,

$$y(t) = 1 + e^{-t}\left[-1 + \frac{1}{2} - 1 - t\right] + \sin\frac{\sqrt{3}}{2}t\, e^{-t/2} \times \left[-\frac{2}{\sqrt{3}} + \frac{\sqrt{3}}{7} - \frac{2}{7\sqrt{3}} - \frac{1}{\sqrt{3}} + \frac{2}{\sqrt{3}}\right]$$

$$+ \cos\frac{\sqrt{3}}{2}te^{-t/2}\left[1 - \frac{3}{7}\right] + e^{-3t}\left[\frac{3}{7} - \frac{1}{2}\right]$$

$$y(t) = 1 - \left[\frac{3}{2} + t\right]e^{-t} - \frac{6}{7\sqrt{3}}\sin\frac{\sqrt{3}}{2}t\, e^{-t/2} + \frac{4}{7}\cos\frac{\sqrt{3}}{2}t\, e^{-t/2} + e^{-3t}\left[-\frac{1}{14}\right]$$

3.12.4 Step Response of Parallel *RLC* Circuit

In the parallel *RLC* circuit shown in Fig. 3.8, let the switch *S* be opened at time $t = 0$, thus connecting the d.c. current source $I_i u(t)$ to the circuit. Applying Kirchhoff's current law to the circuit, we get the following integro-differential equation.

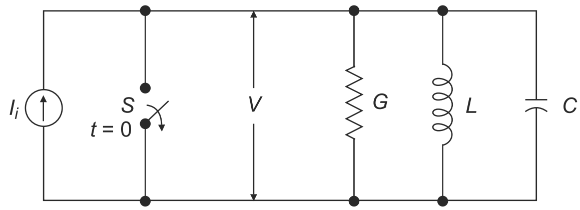


Fig. 3.8 *Parallel RLC circuit*

$$C\frac{dV}{dt} + GV + \frac{1}{L}\int_{-\infty}^{t} V\, dt = I_i \cdot u(t) \qquad (3.32)$$

This may be written as

$$C\frac{\mathrm{d}v(t)}{\mathrm{d}t} + Gv(t) + \frac{1}{L}\int_{-\infty}^{0} v(t)\,\mathrm{d}t + \frac{1}{L}\int_{0}^{t} v(t)\,\mathrm{d}t = I_i \cdot u(t)$$

Taking Laplace transform, the equation becomes

$$C\left[sV(s) - v(0^-)\right] + GV(s) + \frac{1}{L}\mathcal{L}\left[\psi\left(0^-\right)\right] + \frac{1}{L}\frac{V(s)}{s} = \frac{I_i}{s}$$

where $\psi(0^-)$ is the flux linkage and equals $Li(0^-)$.

Now, the initial conditions are inserted. Since the voltage across a capacitor, *C*, changes instantaneously, $v(0^-) = 0$. Also, the current in the inductor, *L*, during the time interval from $-\infty$ to 0 is zero. Hence, $\psi(0^-) = 0$.

$$sCV(s) + GV(s) + \frac{1}{sL}V(s) = \frac{I_i}{s}$$

or

$$V(s)\left[sC + G + \frac{1}{sL}\right] = \frac{I_i}{s}$$

Therefore,

$$V(s) = \frac{I_i}{\left(s^2C + sG + \frac{1}{L}\right)} = \frac{I_i}{C\left(s^2 + \frac{G}{C}s + \frac{1}{LC}\right)} = \frac{I_i}{C(s - p_1)(s - p_2)}$$

where

$$p_1, p_2 = \frac{-G}{2C} \pm \frac{1}{2C}\sqrt{G^2 - 4\frac{C}{L}}$$

Therefore,

$$v(t) = \frac{I_i / C}{\left(p_1 - p_2\right)}\left[e^{p_1 t} - e^{p_2 t}\right] \qquad (3.32)$$

Given the parameter values, we can take the inverse Laplace transform to obtain $v(t)$.

Example 3.36 *For the circuit shown in Fig. E3.36, find the voltage across the resistor of 0.5 Ω when the switch, S, is opened at t = 0. Assume that there is no charge on the capacitor and no current in the inductor before switching.*

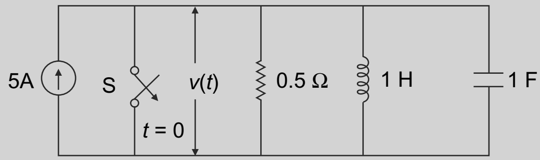


Fig. E3.36

Solution Applying KCL to the circuit, we get

$$\frac{v(t)}{0.5}+1\int_{-\infty}^{t} v(t)\,\mathrm{d}t+1\frac{\mathrm{d}v(t)}{\mathrm{d}t}=5$$

or
$$2v(t)+1\int_{-\infty}^{0} v(t)\,\mathrm{d}t+1\int_{0}^{t} v(t)\,\mathrm{d}t+\frac{\mathrm{d}v(t)}{\mathrm{d}t}=5$$

$$1\int_{-\infty}^{0} v(t)\,\mathrm{d}t = \text{initial current through inductor} = 0$$

Therefore,
$$2v(t)+1\int_{0}^{t} v(t)\,\mathrm{d}t+\frac{\mathrm{d}v(t)}{\mathrm{d}t}=5$$

Taking Laplace transform, we get

$$2V(s)+\frac{V(s)}{s}+\left[sV(s)-v(0^+)\right]=\frac{5}{s}$$

As initial voltage across C, $v(0^+) = 0$,

$$\left[2+\frac{1}{s}+s\right]V(s)=\frac{5}{s}$$

$$(s^2 + 2s + 1)\ V(s) = 5$$

Therefore,
$$V(s)=\frac{5}{(s^2+2s+1)}=\frac{5}{(s+1)^2}$$

Taking inverse Laplace transform, we get

$$v(t) = 5te^{-t}\,u(t)$$

Example 3.37 *In the parallel RLC circuit shown in Fig. E3.37, the switch, S, is opened at time t = 0. Assuming that the initial conditions are zero, obtain v(t).*

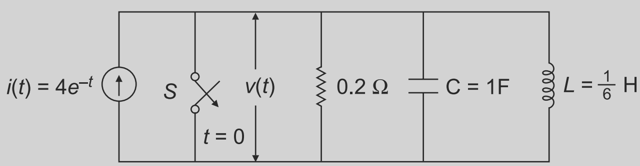


Fig. E3.37

Solution Applying KCL to the circuit, we get

$$i_R(t)+i_L(t)+i_C(t)=i(t)$$

$$\frac{v(t)}{R}+\frac{1}{L}\int_{-\infty}^{0} v(t)\,\mathrm{d}t+\frac{1}{L}\int_{-\infty}^{t} v(t)\,\mathrm{d}t+\frac{C\mathrm{d}v(t)}{\mathrm{d}t}=i(t)$$

$$\frac{v(t)}{0.2}+6\int_{-\infty}^{0} v(t)\mathrm{d}t+6\int_{-\infty}^{t} v(t)\,\mathrm{d}t+1\frac{dv(t)}{\mathrm{d}t}=4e^{-t}$$

Taking Laplace transform, we get

$$5V(s)+\frac{6\psi(0^+)}{s}+\frac{6V(s)}{s}+1\left[sV(s)-v(0^+)\right]=\frac{4}{s+1}$$

$$v(0^-)=v(0^+)=0$$

$$\psi(0^-)=\psi(0^+)=0$$

Therefore, $\left(5+\frac{6}{s}+s\right)V(s)=\frac{4}{s+1}$

Therefore, $(s^2+5s+6)V(s)=\frac{4s}{s+1}$

Therefore, $V(s)=\frac{4s}{(s+1)(s^2+5s+6)}=\frac{4s}{(s+1)(s+2)(s+3)}$

Let $\frac{4s}{(s+1)(s+2)(s+3)}=\frac{A_1}{s+1}+\frac{A_2}{s+2}+\frac{A_3}{s+3}$

Then, $A_1=\frac{4s}{(s+2)(s+3)}\Big|_{s=-1}=\frac{-4}{1(2)}=-2$

$$A_2=\frac{4s}{(s+1)(s+3)}\Big|_{s=-2}=\frac{-8}{(-1)(1)}=8$$

$$A_3=\frac{4s}{(s+1)(s+2)}\Big|_{s=-3}=\frac{-12}{(-2)(-1)}=-6$$

Therefore, $V(s)=\frac{-2}{s+1}+\frac{8}{s+2}-\frac{6}{s+3}$

Taking inverse Laplace transform, we get

$$v(t)=\left(-2e^{-t}+8e^{-2t}-6e^{-3t}\right)u(t)$$

SINUSOIDAL RESPONSE OF PASSIVE (RLC) CIRCUITS 3.13

Currents for the series *RL*, *RC* and *RLC* circuits excited by different a.c. sources, like sin ωt, cos ωt, sin $(\omega t+\theta)$ and cos $(\omega t+\theta)$ are being determined in this section.

Example 3.38 *A cosine wave* cos ωt *is applied as the input to the series RL circuit shown in Fig. E3.38(a). Find the resultant current i(t) if the switch, S, is closed at t = 0. Assume that there is no stored energy in the circuit.*

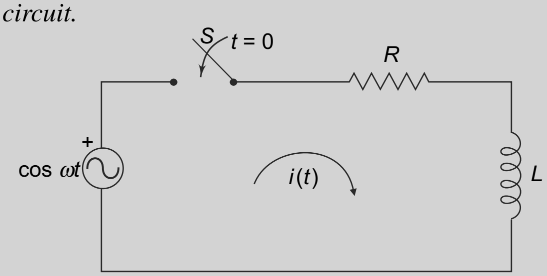


Fig. E3.38(a)

Solution Refer to the series *RL* circuit as shown in Fig. E3.38(a).
Applying Kirchhoff's voltage Law to the above circuit, we get

$$\cos \omega t = Ri(t) + L\frac{di(t)}{dt}$$

Taking Laplace transform on both sides, we get

$$\frac{s}{s^2+\omega^2} = (R+sL)I(s)$$

$$I(s) = \frac{s}{(R+sL)(s^2+\omega^2)} = \frac{s}{L(s^2+\omega^2)\left(s+\frac{R}{L}\right)} = \frac{A_1}{s+\frac{R}{L}} + \frac{A_2 s + A_3}{s^2+\omega^2} \quad (1)$$

To find A_1, A_2 and A_3

Cross multiplying Eq. (1), we get

$$\frac{s}{L} = A_1\left(s^2+\omega^2\right) + \left(A_2 s + A_3\right)\left(s+\frac{R}{L}\right)$$

$$= A_1 s^2 + A_1\omega^2 + A_2 s^2 + \frac{A_2 Rs}{L} + A_3 s + \frac{A_3 R}{L}$$

Equating the co-coefficients of s^2 terms, we get

$$A_1 + A_2 = 0 \quad (2)$$

Equating the coefficients of s term, we get

$$\frac{A_2 R}{L} + A_3 = \frac{1}{L}$$

$$A_2 R + A_3 L = 1 \quad (3)$$

Equating the constant terms, we get

$$A_1\omega^2 + \frac{A_3 R}{L} = 0$$

$$A_1 = -\frac{A_3 R}{L\omega^2} \quad (4)$$

Substituting Eq. (4) in Eq. (2), we get

$$A_2 - \frac{A_3 R}{L\omega^2} = 0 \quad (5)$$

Multiplying Eq. (5) by R, we get

$$A_2 R - \frac{R^2 A_3}{L\omega^2} = 0 \quad (6)$$

Subtracting Eq. (6) and Eq. (3), we get

$$A_3\left(\frac{R^2}{L\omega^2} + L\right) = 1$$

$$A_3 = \frac{L\omega^2}{R^2+\omega^2 L^2} = \frac{L\omega^2}{Z^2} \quad (7)$$

Substituting Eq. (6) in Eq. (3), we get

$$A_2 R + \frac{L^2\omega^2}{Z^2} = 1$$

$$A_2 = \frac{1}{R}\left(1 - \frac{\omega^2 L^2}{Z^2}\right) = \frac{1}{R}\left(\frac{R^2 + \omega^2 L^2 - \omega^2 L^2}{Z^2}\right)$$

$$A_2 = \frac{R}{Z^2}$$

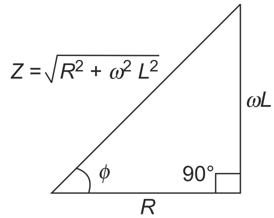


Fig. E3.38(b)

Using Eq. (2), we get

$$A_1 = -A_2 = -\frac{R}{Z^2}$$

Substituting the values of A_1, A_2 and A_3 in Eq. (1), we get

$$I(s) = -\frac{R}{Z^2\left(s + \frac{R}{L}\right)} + \frac{R}{Z^2}\left(\frac{s}{s^2 + \omega^2}\right) + \frac{\omega L}{Z^2}\left(\frac{\omega}{s^2 + \omega^2}\right)$$

Taking inverse Laplace transform on both sides, we get

$$i(t) = -\frac{R}{Z^2}e^{-\frac{R}{L}t} + \frac{R}{Z^2}\cos\omega t + \frac{\omega L}{Z^2}\sin\omega t$$

$$= -\frac{R}{Z^2}e^{-\frac{R}{L}t} + \frac{1}{Z^2}\left[R\cos\omega t + \omega L\sin\omega t\right] \tag{8}$$

Substituting $R = Z\cos\phi$ and $\omega L = Z\sin\phi$ as found from the impedance triangle shown in Fig. E3.38(b) in Eq. (8), we get the circuit current as

$$i(t) = -\frac{R}{Z^2}e^{-\frac{R}{L}t} + \frac{1}{Z^2}Z\left[\cos\phi\ \sin\ \omega t + \sin\phi\sin\omega t\right]$$

$$i(t) = \frac{1}{Z}\cos(\omega t - \phi) - \frac{R}{Z^2}e^{-\frac{R}{L}t}$$

where $\phi = \tan^{-1}\left(\frac{\omega L}{R}\right)$.

Example 3.39 *Determine the current i(t) flowing through the series RL circuit shown in Fig. E3.39 when the sine wave of 10 sin 25t is applied to it.*

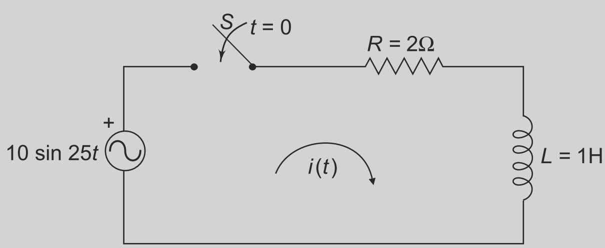


Fig. E3.39

Solution Applying Kirchhoff's voltage law to the given circuit, we get

$$2i(t)+1\frac{di(t)}{dt}=10\sin 25t$$

Taking Laplace transform, we get

$$2I(s)+\left[sI(s)-i(0)\right]=10\times\frac{25}{s^2+(25)^2}$$

where $i(0^-)$ is the initial current passing through the circuit. The inductor does not allow sudden change in current and hence, at $t = 0$, $i(0^-) = 0$.

Therefore, $$sI(s)+2I(s)=\frac{10\times 25}{s^2+(25)^2}$$

$$I(s)=\frac{250}{\left(s^2+625\right)(s+2)}$$

Using partial fractions, the above equation can be expanded as

$$I(s)=\frac{250}{(s+2)(s+j25)(s-j25)}$$

$$I(s)=\left[\frac{A_1}{s+2}+\frac{A_2}{s+j25}+\frac{A_3}{s-j25}\right]$$

where $$A_1=(s+2)I(s)\big|_{s=-2}=\frac{250}{\left[s^2+(25)^2\right]}\Bigg|_{s=-2}=\frac{250}{629}$$

$$A_2=(s+j25)I(s)\big|_{s=-j25}=\frac{250}{(s+2)(s-j25)}\Bigg|_{s=-j25}$$

$$=\frac{250}{(2-j25)(-j50)}=\frac{-5}{(25+j2)}$$

$$A_3=(s-j25)I(s)\big|_{s=j25}=\frac{250}{(s+2)(s+j25)}\Bigg|_{s=j25}$$

$$=\frac{250}{(2+j25)(j50)}=\frac{-5}{(25-j2)}$$

Substituting the values of A_1, A_2 and A_3 in $I(s)$, we get

$$I(s)=\frac{250/629}{s+2}-\frac{5}{(25+j2)(s+j25)}-\frac{5}{(25-j2)(s-j25)}$$

Taking the inverse Laplace transform, we get

$$i(t) = \frac{250}{629}e^{-2t} - \frac{5}{(25+j2)}e^{-j25t} - \frac{5}{(25-j2)}e^{j25t}$$

$$= \frac{250}{629}e^{-2t} - \frac{5}{629}\left[e^{-j25t}(25-j2) + e^{j25t}(25+j2)\right]$$

$$= \frac{250}{629}e^{-2t} - \frac{5}{629}[50\cos 25t - 4\sin 25t] = \frac{250}{629}e^{-2t} - \frac{250}{629}\cos 25t + \frac{20}{629}\sin 25t$$

$$= 0.3974e^{-2t} - 0.3974\cos 25t + 0.0318\sin 25t\ A$$

Note: Using partial fractions, the function $I(s)$ can be expanded as

$$I(s) = \frac{250}{(s^2+625)(s+2)} = \frac{A_1}{s+2} + \frac{A_2 s + A_3}{s^2+25^2}$$

$$= \frac{250/629}{s+2} + \frac{\frac{250}{629}s + \frac{500}{629}}{s^2+25^2} = \frac{250/629}{s+2} - \frac{250}{629}\times\frac{s}{s^2+25^2} + \frac{20}{629}\times\frac{25}{s^2+25^2}$$

Taking inverse Laplace transform, we get

$$i(t) = \frac{250}{629}e^{-2t} - \frac{250}{629}\cos 25t + \frac{20}{629}\sin 25t$$

$$= 0.3974e^{-2t} - 0.3974\cos 25t + 0.0318\sin 25t\ A$$

Example 3.40 *Obtain the current at $t > 0$, if a sine wave 100 sin 314 t is applied when the switch, S, is moved to the position 2 from the position 1 at $t = 0$ as shown in the RL circuit of Fig. E3.40(a). Assume a steady-state current of 1A when the switch is at position* 1.

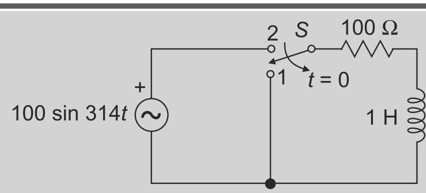


Fig. E3.40(a)

Solution When the switch, S, is at the position 1, the steady-state current, i.e. $i(0^-) = 1\ A$. When the switch, S, is moved to the position 2, i.e. at $t = 0$, the a.c. voltage 100 sin 314t appears across the series RL circuit. Since the inductance is connected, $i(0^-) = i(0^+) = 1A$

Applying Kirchhoff's voltage law to the given circuit, we get

$$Ri(t) + L\frac{di(t)}{dt} = 100\sin 314t$$

$$100i(t) + 1\frac{di(t)}{dt} = 100\sin 314t$$

Taking Laplace transform, we get

$$100I(s) + 1\left[sI(s) - i(0^+)\right] = \frac{100\times 314}{s^2+314^2}$$

$$100I(s) + [sI(s) - 1] = \frac{100\times 314}{s^2+314^2}$$

$$I(s)(s+100)=\frac{31400}{s^2+314^2}+1$$

$$I(s)=\frac{31400}{(s+100)(s^2+314^2)}+\frac{1}{(s+100)}$$

where $$\frac{31400}{(s+100)(s^2+314^2)}=\frac{A_1}{s+100}+\frac{A_2s+A_3}{s^2+314^2}$$

$$A_1=\left.\frac{31400}{s^2+314^2}\right|_{s=-100}=\frac{31400}{108596}=0.28914$$

Cross multiplying, we get

$$31400 = 0.28914s^2 + 28509 + A_2s^2 + 100\,A_2s + A_3s + 100\,A_3$$

Equating the coefficients of s^2, we get

$$A_2 = -0.28914$$

Equating the constant terms, we get

$$31400 = 28509 + 100\,A_3$$

Therefore, $A_3 = 28.91$

Hence, $$I(s)=\frac{A_1}{s+100}+\frac{A_2s+A_3}{s^2+314^2}+\frac{1}{(s+100)}=\frac{A_1+1}{s+100}+\frac{A_2s+A_3}{s^2+314^2}$$

$$=\frac{1.28914}{s+100}-\frac{0.28914s}{s^2+314^2}+\frac{28.91}{s^2+314^2}$$

$$=1.28914\left(\frac{1}{s+100}\right)-0.28914\left(\frac{s}{s^2+314^2}\right)+\frac{28.91}{314}\left(\frac{314}{s^2+314^2}\right)$$

$$=1.28914\left(\frac{1}{s+100}\right)-0.28914\left(\frac{s}{s^2+314^2}\right)+0.09207\left(\frac{314}{s^2+314^2}\right)$$

Taking inverse Laplace transform, we get

$$i(t)=1.28914e^{-100t}-0.28914\cos 314t+0.09207\sin 314t \quad A$$

A right-angled triangle is constructed with 0.09207 and 0.28914 as two sides as shown in Fig. E3.40 (b) .

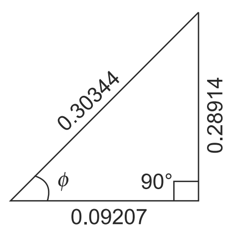


Fig. E3.40(b)

Therefore, $$\tan\phi=\frac{0.28914}{0.09207}=3.14$$

$$\phi=\tan^{-1}(3.14)=72.33°$$

Also, $\cos\phi=\dfrac{0.09207}{0.30344}$; therefore, $0.09207 = 0.30344\cos\phi$
$= 0.30344\cos(72.33°)$

Also, $\sin\phi=\dfrac{0.28914}{0.30344}$; therefore, $0.28914 = 0.30344\sin\phi = 0.30344\sin(72.33°)$

Therefore, $i(t) = 1.28914e^{-100t} + 0.30344\sin(314t - 72.33°)$ A

In the above current equation, the first term is the transient part and the second term is the steady state part.

Example 3.41 *In the circuit shown in Fig. E3.41(a), the switch, S, is closed at the position 1, for a long time. At time t = 0, the switch is moved to the position 2. Find i(t) for t ≥ 0.*

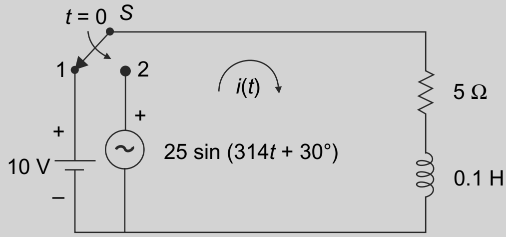


Fig. E3.41(a)

Solution Since the switch is kept closed at the position 1 for a long time, the circuit reached steady state. The steady state of *RL* circuit with the switch in the position 1 is shown in Fig. E3.41(b). The steady-state current is

$$i(0^+) = \frac{10}{5} = 2 \text{ A}$$

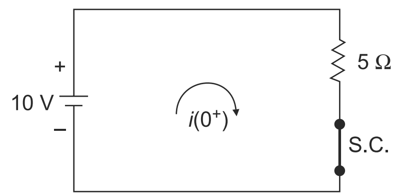


Fig. E3.41(b)

When the switch is moved from the position 1 to the position 2, a steady current $i(0^+)$ is flowing through the inductor. Since the inductor does not allow sudden change in current, this steady current $i(0^+)$ will be the initial current when the switch is closed at the position 2.

Therefore, $i(0^-) = i(0^+) = 2$ A

The *RL* circuit with the switch at the position 2 is shown in Fig. E3.41(c).

Applying KVL to the circuit, we get

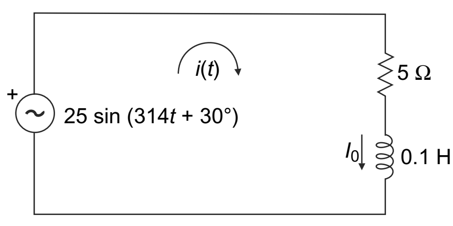


Fig. E3.41(c)

$$5i(t) + 0.1\frac{di(t)}{dt} = 25\sin(314t + 30°)$$

$$= 25(\sin 314t \cos 30° + \cos 314t \sin 30°)$$

$$= 25 \times \left(\sin 314 \frac{\sqrt{3}}{2} + \cos 314t \times \frac{1}{2}\right)$$

$$= 25 \times \frac{\sqrt{3}}{2} \times \sin 314t + 25 \times \frac{1}{2} \times \cos 314t$$

Taking Laplace transform, we get

$$5I(s) + 0.1\left[sI(s) - i(0^+)\right] = 25 \times \frac{\sqrt{3}}{2} \times \frac{314}{s^2 + 314^2} + 25 \times \frac{1}{2} \times \frac{s}{s^2 + 314^2}$$

$$I(s)(5 + 0.1s) - 0.2 = \frac{6798.5}{s^2 + 314^2} + \frac{12.5}{s^2 + 314^2}$$

$$I(s)\times 0.1(s+50)=\frac{12.5+6798.5}{s^2+314^2}+0.2$$

$$I(s)=\frac{12.5s+6798.5}{0.1(s+50)(s^2+314^2)}+\frac{0.2}{0.1(s+50)}$$

$$=\frac{125s+67985}{(s+50)(s^2+314^2)}+\frac{2}{(s+50)}$$

Let $$\frac{125s+67985}{(s+50)(s^2+314^2)}=\frac{A_1}{s+50}+\frac{A_2s+A_3}{s^2+314^2}$$

Therefore, $$\left.\frac{125s+67985}{(s^2+314^2)}\right|_{s=-50}=\frac{125\times(-50)+67985}{(-50)^2+314^2}=0.6107$$

Cross multiplying, we get

$125s + 67985 = A_1(s^2 + 314^2) + (A_2s + A_3)(s + 50)$

$125s + 67985 = A_1s^2 + A_1\, 314^2 + A_2s^2 + 50\, A_2s + A_ss + 50\, A_3$

$125s + 67985 = (A_1 + A_2)\, s^2 + (50\, A_2 + A_3)s + (A_1\, 314^2 + 50\, A_3)$

Equating coefficients of s^2 term of the equation, we get

$A_1 + A_2 = 0$

Therefore, $A_1 = A_2 = -0.6107$

Equating coefficients of s of the above equation, we get

$50\, A_2 + A_3 = 125$

Therefore, $A_3 = 125 - 50\, A_2$

$= 125 - 50 \times (-0.6107) = 155.535$

$$I(s)=\frac{0.6107}{s+50}+\frac{-0.6107s+155.535}{s^2+314^2}+\frac{2}{(s+50)}$$

$$=\frac{2.6107}{s+50}-\frac{0.6107}{s^2+314^2}+\frac{155.535}{s^2+314^2}$$

$$=2.6107\times\frac{1}{s+50}-0.6107\times\frac{s}{s^2+314^2}+\frac{155.535}{314}\times\frac{314}{s^2+314^2}$$

$$=2.6107\times\frac{1}{s+50}-0.6107\times\frac{s}{s^2+314^2}+0.4953\times\frac{314}{s^2+314^2}$$

Taking inverse Laplace transform, we get

$$\mathcal{L}^{-1}[I(s)]=\mathcal{L}^{-1}\left[2.6107\times\frac{1}{s+50}-0.6107\times\frac{s}{s^2+314^2}+0.4953\times\frac{314}{s^2+314^2}\right]$$

$$\mathcal{L}^{-1}[I(s)]=2.6107\mathcal{L}^{-1}\left[\frac{1}{s+50}\right]-0.6170\,\mathcal{L}^{-1}\left[\frac{s}{s^2+314^2}\right]+0.4963\,\mathcal{L}^{-1}\left[0.4963\times\frac{314}{s^2+314^2}\right]$$

Therefore, $i(t) = 2.6107\, e^{-50t} - 0.6107 \cos 314t + 0.4953 \sin 314t$

$= 2.6107\, e^{-50t} + [\sin 314t \times 0.4953 - \cos 314t \times 0.6107]$ A

A right-angled triangle is constructed with 0.4953 and 0.6107 as two sides as shown in Fig. E3.41(d)

$$\tan\phi = \frac{0.6170}{0.4953} = 1.233$$

Therefore, $\phi = \tan^{-1} 1.233 = 50°$

Also, $\cos\phi = \dfrac{0.4953}{0.7863}$

i.e. $0.4953 = 0.7863 \cos\phi = 0.7863 \cos 51°$

$$\sin\phi = \frac{0.6107}{0.7863}$$

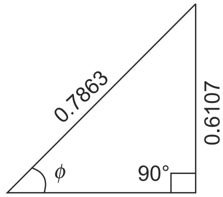


Fig. E3.41(d)

i.e. $0.6107 = 0.7863 \sin\phi = 0.7863 \cos 51°$

Therefore, $i(t) = 2.6107e^{-50t} + [\sin 314t \times 0.7863 \cos 51° - \cos 314t \times 0.7863 \sin 51°]$

$$= 2.6107e^{-50t} + 0.7863[\sin 314t \cos 51° - \cos 314t \sin 51°]$$

$$= 2.6107e^{-50t} + 0.7863 \sin(314t - 51°)\text{ A}$$

In the above current equation, the first term is the transient response and the second term is the steady-state response or forced response.

Example 3.42 *A sine wave sin ωt is applied as the input to the series RC circuit shown in Fig. E3.42(a). Find the resultant current i(t) if the switch, S, is closed at t = 0. Assume that there is no stored energy in the circuit.*

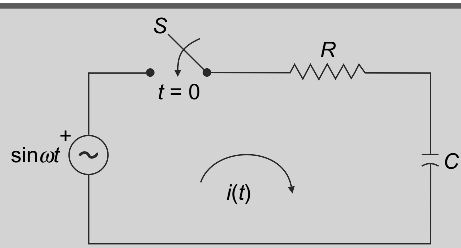


Fig. E3.42(a)

Solution Refer to the series RC circuit as shown in Fig. E3.42(a).

Applying Kirchhoff's voltage law to the above circuit, we get

$$\sin\omega t = Ri(t) + \frac{1}{C}\int i(t)\,dt$$

Taking Laplace transform on both sides, we get

$$\frac{\omega}{(s^2+\omega^2)} = RI(s) + \frac{1}{Cs} I(s)$$

$$I(s) = \frac{\dfrac{\omega}{(s^2+\omega^2)}}{R + \dfrac{1}{Cs}} = \frac{\dfrac{\omega}{(s^2+\omega^2)}}{\dfrac{R}{s}\left(s + \dfrac{1}{RC}\right)}$$

$$I(s) = \frac{s}{R\left(s + \frac{1}{RC}\right)} \frac{\omega}{s^2 + \omega^2} = \frac{A_1}{s + \frac{1}{RC}} + \frac{A_2 s + A_3}{s^2 + \omega^2} \tag{1}$$

To find the values of A_1, A_2 and A_3

Cross multiplying Eq. (1), we get

$$\frac{\omega}{R} s = A_1\left(s^2 + \omega^2\right) + \left(A_2 s + A_3\right)\left(s + \frac{1}{RC}\right)$$

$$= A_1 s^2 + A_1\omega^2 + A_2 s^2 + \frac{A_2 s}{RC} + A_3 + \frac{A_3}{RC} \tag{2}$$

Substituting the value $s = -\frac{1}{RC}$ in Eq. (2), we get

$$A_1 = \frac{\frac{-1}{\omega C}}{R^2 + \left(\frac{1}{\omega C}\right)^2} = \frac{-1}{\omega C Z^2}$$

where $Z = \sqrt{R^2 + \left(\frac{1}{\omega C}\right)^2}$

Equating the coefficients of s^2 terms, we get

$$A_1 + A_2 = 0$$

$$A_2 = -A_1 = \frac{1}{\omega C Z^2}$$

Equating the coefficients of s^2 terms, we get

$$\frac{A_2}{RC} + A_3 = \frac{\omega}{R}$$

Substituting the known values in the above equation, we get

$$A_3 = \frac{\omega}{R} - \frac{1}{\omega R C^2 Z^2} = \left[\frac{\omega^2 C^2 Z^2 - 1}{\omega R C^2 Z^2}\right]$$

$$A_3 = \left[\frac{\omega^2 C^2\left(R^2 + \frac{1}{\omega^2 C^2}\right) - 1}{\omega R C^2 Z^2}\right] = \left(\frac{\omega^2 C^2 R^2 + 1 - 1}{\omega R C^2 Z^2}\right)$$

$$A_3 = \frac{\omega R}{Z^2}$$

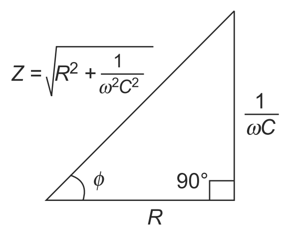


Fig. E3.42(b)

Substituting the value of A_1, A_2 and A_3 in Eq. (1), we get

$$I(s) = \frac{\left(\frac{-1}{\omega C Z^2}\right)}{\left(s + \frac{1}{RC}\right)} + \left(\frac{1}{\omega C Z^2}\right)\frac{s}{\left(s^2 + \omega^2\right)} + \left(\frac{R}{Z^2}\right)\frac{\omega}{s^2 + \omega^2}$$

Taking inverse Laplace transform on both sides, we get

$$i(t) = \frac{-1}{\omega C Z^2} e^{-\frac{t}{RC}} + \frac{1}{Z^2}\left[\frac{1}{\omega C}\cos\omega t + R\sin\omega t\right] \qquad (3)$$

From the impedance triangle shown in Fig. E3.42(b), we get

$$\tan\phi = \frac{1}{\omega RC},\ \cos\phi = \frac{R}{Z} \quad \text{and} \quad \sin\phi = \frac{1}{\omega CZ}$$

Substituting $R = Z\cos\phi$ and $\omega L = Z\sin\phi$ in Eq. (3), we get the circuit current as

$$i(t) = \frac{-1}{Z}\sin\phi e^{-\frac{t}{RC}} + \frac{1}{Z^2} Z\left[\sin\phi\cos\omega t + \sin\omega t\cos\phi\right]$$

$$i(t) = \frac{1}{Z}\sin(\omega t + \phi) - \frac{1}{Z}\sin\phi\, e^{-\frac{t}{RC}}$$

where $\phi = \tan^{-1}\left(\frac{1}{\omega CR}\right)$ and $Z = \sqrt{R^2 + \frac{1}{\omega^2 C^2}}$

Example 3.43 *A cosine wave $\cos\omega t$ is applied as the input to the series RC circuit shown in Fig. E3.43. Find the resultant current i(t) if the switch, S, is closed at t = 0. Assume that there is no stored energy in the circuit.*

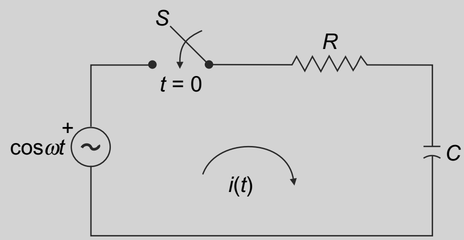


Fig. E3.43

Solution Refer to the series *RC* circuit as shown in Fig. E3.43.
Applying Kirchhoff's voltage law to the above circuit, we get

$$\cos\omega t = Ri(t) + \frac{1}{C}\int i(t)\,dt$$

Taking Laplace transform on both sides, we get

$$\frac{s}{(s^2+\omega^2)} = RI(s) + \frac{1}{sC}I(s)$$

$$I(s) = \frac{\frac{s}{(s^2+\omega^2)}}{R + \frac{1}{sC}} = \frac{\frac{s}{(s^2+\omega^2)}}{\frac{R}{s}\left(s + \frac{1}{RC}\right)}$$

$$I(s) = \frac{s^2}{R\left(s + \frac{1}{RC}\right)}\frac{1}{s^2+\omega^2} = \frac{A_1}{s + \frac{1}{RC}} + \frac{A_2 s + A_3}{s^2+\omega^2} \qquad (1)$$

To find the values of A_1, A_2 and A_3

Cross multiplying Eq. (1), we get

$$\frac{s^2}{R} = A_1 s^2 + A_1\omega^2 + A_2 s^2 + \frac{A_2 s}{RC} + A_3 s + \frac{A_3}{RC}$$

Equating the coefficients of s^2 terms, we get

$$A_1 + A_2 = \frac{1}{R} \tag{2}$$

Equating the coefficients of s terms, we get

$$A_2 + A_3 RC = 0 \tag{3}$$

Equating the constant terms, we get

$$A_1\omega^2 + \frac{A_3}{RC} = 0$$

$$A_3 = -A_1\omega^2 RC \tag{4}$$

Substituting Eq. (4) in Eq. (3), we get

$$A_2 - R^2C^2A_1\omega^2 = 0 \tag{5}$$

Multiplying Eq. (4) by $\omega^2C^2R^2$ on both sides, we get

$$\omega^2C^2R^2A_2 + \omega^2C^2R^2A_1 = \omega^2C^2R^2\frac{1}{R} \tag{6}$$

Adding Eq. (5) and Eq. (6), we get

$$A_2\left(1 + R^2\omega^2C^2\right) = \omega^2C^2R$$

$$A_2 = \frac{\omega^2C^2R}{\omega^2C^2\left(\frac{1}{\omega^2C^2} + R^2\right)} = \frac{R}{Z^2}$$

Using Eq. (2), we get

$$A_1 = \frac{1}{R} - A_2 = \frac{\left(Z^2 - R^2\right)}{Z^2R} = \frac{1}{\omega^2C^2Z^2R} \tag{7}$$

Substituting Eq. (7) in Eq. (4), we get

$$A_3 = \frac{-\omega^2RC}{\omega^2C^2Z^2R} = \frac{-1}{Z^2C}$$

Substituting the values of A_1, A_2 and A_3 in Eq. (1), we get

$$I(s) = \frac{1}{\omega^2C^2Z^2R}\left(\frac{1}{s + \frac{1}{RC}}\right) + \frac{Rs}{\left(s^2 + \omega^2\right)Z^2} - \frac{\omega}{Z^2\omega C\left(s^2 + \omega^2\right)}$$

Taking Inverse Laplace transform on both sides, we get

$$i(t) = \frac{1}{\omega^2C^2Z^2R}e^{\frac{-t}{RC}} + \frac{1}{Z^2}\left(R\cos\omega t - \frac{1}{\omega C}\sin\omega t\right)$$

Substituting $R = Z\cos\phi$ and $\frac{1}{\omega C} = Z\sin\phi$ as found in the impedance triangle shown in Fig.

E3.42(b) in the above equation, we get the circuit current as

$$i(t)=\frac{Z^2\sin^2\phi}{Z^2(Z\cos\phi)}e^{\frac{-t}{RC}}+\frac{1}{Z^2}(Z\cos\phi\cos\omega t-Z\sin\phi\sin\omega t)$$

$$=\frac{1}{Z}\left(\frac{\sin\phi}{\cos\phi}\right)\sin\phi\, e^{\frac{-t}{RC}}+\frac{1}{Z^2}Z(\cos\phi\cos\omega t-\sin\phi\sin\omega t)$$

$$i(t)=\frac{V_m}{Z}\sin\phi\tan\phi\, e^{\frac{-t}{RC}}+\frac{V_m}{Z}\cos(\omega t+\phi)$$

where $\phi=\tan^{-1}\left(\frac{1}{\omega CR}\right)$ and $Z=\sqrt{R^2+\left(\frac{1}{\omega^2C^2}\right)}$

Example 3.44 *In the RC circuit shown in Fig E3.44, the capacitor has an initial charge* $q_0(0^+)=25\times10^{-6}$ *coulombs with polarity as marked. The sinusoidal voltage* $v_i(t)=100\sin(1000t+30°)$ *is applied to the circuit. Determine the expression for the current* $i(t)$.

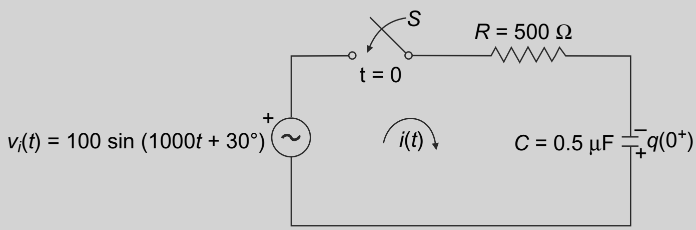


Fig. E3.44

Solution Applying KVL to the circuit, we get

$$Ri(t)+\frac{1}{C}\int_{-\infty}^{0}i(t)\,dt+\frac{1}{C}\int_0^t i(t)\,dt=v(t) \qquad (1)$$

$$500\,i(t)+\frac{-q_0(0^+)}{0.5\times10^{-6}}+\frac{1}{0.5\times10^{-6}}\int_0^t i(t)dt=100\sin(1000t+30°)$$

$$500\,i(t)-\frac{25\times10^{-6}}{0.5\times10^{-6}}+2\times10^6\int_0^t i(t)dt=100\left[0.5\cos1000t+\frac{\sqrt{3}}{2}\sin1000t\right]$$

$$500\,i(t)-50+2\times10^6\int_0^t i(t)dt=50\left[\cos1000t+\sqrt{3}\sin1000t\right]$$

Taking laplace transform on both sides, we get

$$500\,I(s)-\frac{50}{s}+\frac{2\times10^6 I(s)}{s}=50\left[\frac{s}{s^2+(1000)^2}+\frac{\sqrt{3}\times1000}{s^2+(1000)^2}\right]$$

$$10\,I(s)-\frac{1}{s}+\frac{4\times10^4 I(s)}{s}=\frac{s}{s^2+(1000)^2}+\frac{\sqrt{3}\times1000}{s^2+(1000)^2}$$

Dividing both sides by 2, we obtain

$$\left[5+\frac{2\times10^4}{s}\right]I(s)=\frac{0.5}{s}+\frac{s}{2(s^2+1000^2)}+\frac{\sqrt{3}\times1000}{2(s^2+1000^2)}$$

or $$I(s)=\frac{0.5}{5s+2\times10^4}+\frac{0.5\left(s+\sqrt{3}\times1000\right)}{\left(s^2+1000^2\right)}\times\frac{s}{\left(5s+2\times10^4\right)}=\frac{0.1}{s+4000}+\frac{0.1\left(s+\sqrt{3}\times1000\right)s}{(s+4000)\left(s^2+1000^2\right)}$$

Using partial fraction expansion, we get

$$\frac{0.1s\left(s+\sqrt{3}\times1000\right)}{(s+4000)(s^2+10^6)}=\frac{0.1s\left(s+\sqrt{3}\times1000\right)}{(s+4000)(s+j1000)(s-j1000)}$$

$$=\frac{A_1}{(s+4000)}+\frac{A_2}{(s+j1000)}+\frac{A_3}{(s-j1000)}$$

$$=\frac{0.05336}{s+4000}+\frac{0.0233-j0.00667}{s+j1000}+\frac{0.0233+j0.00667}{s-j1000}$$

Therefore, $$I(s)=\frac{0.1}{s+4000}+\frac{0.05336}{s+4000}+\frac{0.0233-j0.00667}{s+j1000}+\frac{0.0233+j0.00667}{s-j1000}$$

Taking inverse Laplace transform, we get

$$i(t)=0.15336e^{-4000t}+(0.0233-j0.00667)e^{-j1000t}+(0.0233+j0.00667)e^{j1000t}$$

$$=0.15336e^{-4000t}+0.0233\left[e^{j1000t}+e^{-j1000t}\right]+j0.00667\left[e^{j1000t}-e^{-j1000t}\right]$$

$$=0.15336e^{-4000t}+0.0466\cos1000t-0.01334\sin1000t \text{ A}$$

Example 3.45 *A series RLC circuit is excited by a sinusoidal source of voltage $v(t)$ = 115 sin 377t V. Determine the current $i(t)$. Assume $R = 5\ \Omega$, $L = 0.1$ H and $C = 500\ \mu$F.*

Solution $$V(s)=\mathcal{L}[v(t)]=\mathcal{L}[115\sin377t]=115\times\frac{377}{s^2+377^2}=\frac{43355}{s^2+377^2}$$

The time domain and s-domain of *RLC* series circuit are shown in Fig. E3.45(a) and Fig. E3.45(b) respectively.

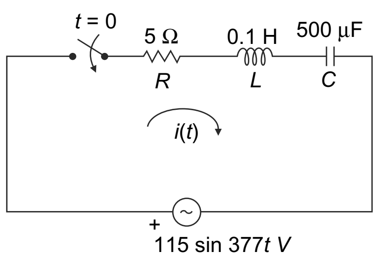


Fig. E3.45(a)

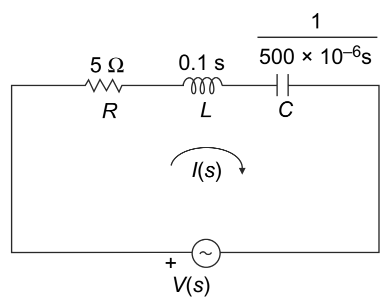


Fig. E3.45(b)

Referring to Fig. E3.45(b), we can write

$$I(s)=\frac{E(s)}{5+0.1s+\frac{1}{500\times10^{-6}s}}=\frac{43355}{s^2+377^2}\times\frac{1}{5+0.1s+\frac{1}{500\times10^{-6}s}}$$

Multiplying the numerator and denominator by ‘s’, we get

$$I(s)=\frac{43355}{s^2+377^2}\times\frac{s}{5s+0.1s^2+\frac{1}{500\times10^{-6}}}$$

$$=\frac{43355}{s^2+377^2}\times\frac{s}{0.1\left(s^2+\frac{5}{0.1}s+\frac{1}{0.1\times500\times10^{-6}}\right)}$$

The above equation can be expressed as

$$I(s)=\frac{433550s}{\left(s^2+377^2\right)\left(s^2+50s+20000\right)}=\frac{A_1s+A_2}{s^2+377^2}+\frac{A_3s+A_4}{s^2+50s+20000} \quad (1)$$

Therefore, $433550s=\left(A_1s+A_2\right)\left(s^2+50s+20000\right)+\left(A_3s+A_4\right)\left(s^2+377^2\right)$

$$=A_1s^3+50A_1s^2+20000A_1s+A_2s^2+50A_2s+20000A_2+A_3s^3+377^2A_3s+A_4s^2+377^2A_4 \quad (2)$$

Equating coefficients of s^3, we get

$$A_1+A_3=0 \quad (3)$$

Equating coefficients of s^2, we get

$$50A_1+A_2+A_4=0 \quad (4)$$

Equating coefficients of s , we get

$$20000A_1+50A_2+377^2A_3=433550 \quad (5)$$

Equating coefficients of constant, we get

$$20000A_2+377^2A_4=0 \quad (6)$$

From Eq. (3), we get $A_3=-A_1$ (7)

Substituting $A_3=-A_1$ in Eq. (5), we get

$$20000A_1+50A_2+377^2\left(-A_1\right)=433550$$

$$-122129A_1+50A_2=433550$$

Dividing by ‘–122129’, we get

$$A_1+\frac{50}{-122129}A_2=\frac{433550}{-122129}$$

$$A_1-4.094\times10^{-4}A_2=-3.5499 \quad (8)$$

From Eq. (6), we get

$$A_4=\frac{-20000}{377^2}A_2=-0.1407A_2 \quad (9)$$

Substituting, $A_4=-0.1407A_2$ in Eq. (4), we get

$$50A_1+A_2-0.1407A_2=0$$

$$50A_1 + 0.8593A_2 = 0 \tag{10}$$

Multiplying Eq. (8) by '–50' and adding with Eq. (10), we get

$$0.87977A_2 = 177.495$$

Therefore, $A_2 = \dfrac{177.495}{0.87977} = 201.7516$

From Eq. (9), $A_4 = -0.1407\,A_2 = -0.1407 \times 201.7516 = -28.3865$

From Eq. (10), $A_1 = \dfrac{-0.8593}{50}A_2 = \dfrac{-0.8593 \times 201.7516}{50} = -3.4673$

From Eq. (7), $A_3 = -A_1 = 3.4673$

Therefore, $I(s) = \dfrac{-3.4673s + 201.7516}{s^2 + 377^2} + \dfrac{3.4673s - 28.3865}{s^2 + 50s + 20000}$

$$= \frac{-3.4673s + 201.7516}{s^2 + 377^2} + \frac{3.4673s - 28.3865}{\left(s^2 + 2 \times 25s + 25^2\right) + \left(20000 - 25^2\right)}$$

$$= \frac{-3.4673s + 201.7516}{s^2 + 377^2} + \frac{3.4673s - 28.865}{(s+25)^2 + 19375}$$

$$= \frac{-3.4673s + 201.7516}{s^2 + 377^2} + \frac{3.4673\left(s - \dfrac{28.3865}{3.4673}\right)}{(s+25)^2 + \left(\sqrt{19375}\right)^2}$$

$$= \frac{-3.4673s + 201.7516}{s^2 + 377^2} + \frac{3.4673(s - 8.1869)}{(s+25)^2 + 139.2^2}$$

$$= \frac{-3.4673s}{s^2 + 377^2} + \frac{201.7516}{s^2 + 377^2} + \frac{3.4673(s + 25 - 33.1869)}{(s+25)^2 + 139.2^2}$$

$$= -3.4673 \times \frac{s}{s^2 + 377^2} + \frac{201.7516}{377} \times \frac{377}{s^2 + 377^2}$$
$$+ 3.4673 \times \frac{s+25}{(s+25)^2 + 139.2^2} - \frac{3.4673 \times 33.1869}{139.2} \times \frac{139.2}{(s+25)^2 + 139.2^2}$$

$$= -3.4673 \times \frac{s}{s^2 + 377^2} + 0.5352 \times \frac{377}{s^2 + 377^2}$$
$$+ 3.4673 \times \frac{s+25}{(s+25)^2 + 139.2^2} - 0.8266 \times \frac{139.2}{(s+25)^2 + 139.2^2}$$

Taking inverse Laplace transform of $I(s)$, we get

$$i(t) = -3.4673\cos 377t + 0.5352\sin 3775t + 3.4673e^{-25t}\cos 139.2t - 0.8266e^{-25t}\sin 139.2t$$
$$= \left[\sin 377t \times 0.5352 - \cos 377t \times 3.4673\right] - e^{-25t}\left[\sin 139.2t \times 0.8266 - \cos 139.2t \times 3.4673\right] \text{ A}$$

Let us construct a right-angled triangle with 0.5352 and 3.4673 as two sides as shown in Fig. E3.45(c).

Referring to Fig. E3.45(c), we can write

$$\tan\phi_1 = \frac{3.4673}{0.5352} = 6.4785$$

Therefore, $\phi_1 = \tan^{-1} 6.4785 = 81.2°$

Also, $$\cos\phi_1 = \frac{0.5352}{3.5084}$$

Therefore, $0.5352 = 3.5084\cos\phi_1 = 3.5084\cos 81.2°$

Also, $$\sin\phi_1 = \frac{3.4673}{3.5084}$$

Therefore, $3.4673 = 3.5084\sin\phi_1 = 3.5084\sin 81.2°$

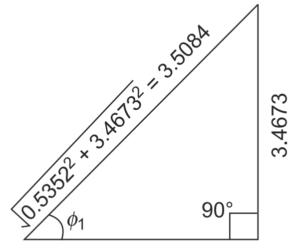


Fig. E3.45(c)

Let us construct another right-angled triangle with 0.8266 and 3.4673 as two sides as shown in Fig. E3.45(d).

Referring to Fig. E3.45(d), we can write

$$\tan\phi_2 = \frac{3.4673}{0.8266} = 4.1947$$

Therefore, $\phi_2 = \tan^{-1} 4.1947 = 76.6°$

Also, $$\cos\phi_2 = \frac{0.8266}{3.5645}$$

Therefore, $0.8266 = 3.5645\cos\phi_2 = 3.5645\ \cos 76.6°$

Also, $\sin\phi_2 = \dfrac{3.4673}{3.5645}$

Therefore, $3.4673 = 3.5645\ \sin\phi_2 = 3.5645\ \sin 76.6°$

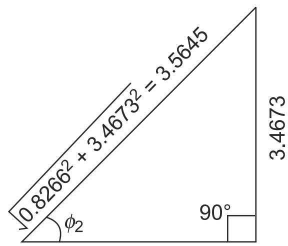


Fig. E3.45(d)

Now $i(t)$ can be written as

$$\begin{aligned} i(t) &= [\sin 377t \times 3.5084\ \cos 81.2° - \cos 377t \times 3.5084 \sin 81.2°] \\ &\quad - e^{-25t}[\sin 139.2t \times 3.5645\ \cos 76.6° - \cos 139.2t \times 3.5645\ \sin 76.6°] \\ &= 3.5084[\sin 377t \cos 81.2° - \cos 377t \sin 81.2°] \\ &\quad - 3.5645e^{-25t}[\sin 139.2t \cos 76.6° - \cos 139.2t \sin 76.6°] \\ &= 3.5084 \sin(377t - 81.2°) - 3.5645e^{-25t}\ \sin(139.2t - 76.6°)\,\text{A} \end{aligned}$$

REVIEW QUESTIONS

3.1 State and explain Laplace transform and its inverse transform.

3.2 What is region of convergence?

3.3 Find the Laplace transforms of

(a) e^{-t}, (b) e^{10t},

(c) $\delta(t)$, (d) $2 - 2e^{t} + 0.5 \sin 4t$,

(e) $e^{-t}\sin 4t$ and (f) $e^{2t} + 2e^{-2}t - t^2$

Ans: (a) $\dfrac{1}{(s+1)}$, (b) $\dfrac{1}{(s-10)}$, (c) 1,

(d) $\dfrac{-2(s+16)}{s(s-1)(s^2+16)}$, (e) $\dfrac{4}{(s+1)^2+16}$, (f) $\dfrac{3s^4-2s^3-2s^2+8}{s^3(s^2-4)}$

3.4 Find the Laplace transform of sin *at* sin *bt*.

3.5 Find the two-sided Laplace transform of $f(t)=\begin{cases} e^{-3t} & \text{for } t\ge 0 \\ 0, & \text{for } t<0 \end{cases}$

What is its region of convergence?

3.6 If $\mathcal{L}\{f_1(t)\}=F_1(s)$ and $\mathcal{L}\{f_2(t)\}=F_2(s)$, show that $\mathcal{L}\{f_1(t)*f_2(t)\}=F_1(s)\cdot F_2(s)$

3.7 If the Laplace transform of $f(t)$ is known to be $F(s)$, what is the Laplace transform of $f(t-T)$?

3.8 Find the Laplace transform of $f(t)=\sin t \cosh(t)$.

3.9 Obtain the Laplace transform of $e^{-at}\cos(\omega_c t+\theta)$

Ans: $\dfrac{\cos\theta(s+a)}{(s+a)^2+\omega_c^2}-\dfrac{\omega_c \sin\theta}{(s+a)^2+\omega_c^2}$

3.10 Find the Laplace transform of the following waveforms shown in Fig. Q3.10(a) and (b).

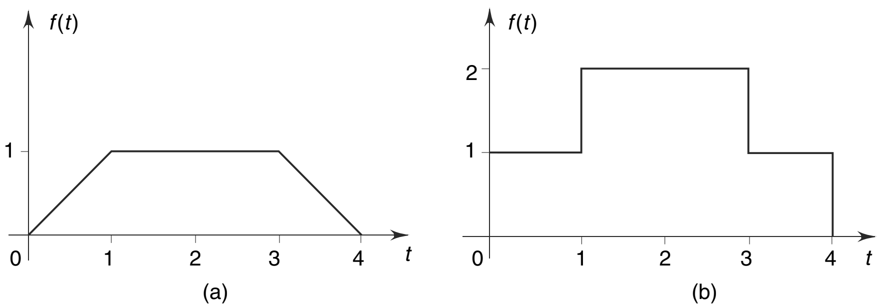


Fig. Q3.10

3.11 Find the inverse Laplace transform of

(a) $\dfrac{s-1}{s(s+1)}$, (b) $\dfrac{s^3+1}{s(s+1)(s+2)}$, (c) $\dfrac{s-1}{(s+1)(s^2+2s+5)}$

Ans: (a) $-1+e^{-t}$, (b) $\delta(t)+0.5-3.5e^{-2t}$

(c) $-0.5e^{-t}+0.5e^{-t}\cos 2t+0.5e^{-t}\sin 2t$

3.12 Find the Laplace transform of the waveform $f(t)$ shown in Fig. Q3.12.

Ans: $F(s)=\dfrac{1}{s}\left(1-3e^{-s}+4e^{-2s}-4e^{-4s}+2e^{-5s}\right)$

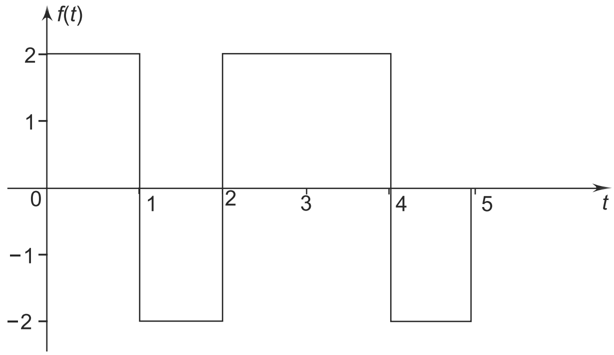


Fig. Q3.12

3.13 Discuss initial value and final value theorems in Laplace transform domain.

3.14 Find $f(\infty)$ if $F(s)$ is given by

(a) $\dfrac{1}{(s-1)}$ (b) $\dfrac{s-2}{s(s+1)}$ (c) $\dfrac{(s-1)}{s(s^2-1)}$ (d) $\dfrac{1}{(s+j)(s-j)}$

Ans: (a) ∞ (b) -2 (c) 1 (d) undefined

3.15 Find $f(0^+)$ if $F(s)=\dfrac{2s+1}{s^2-1}$.

Ans: 2

3.16 Find the initial and final values of the function $F(s)=\dfrac{17s^3+7s^2+s+6}{s^5+3s^4+5s^3+4s^2+2s}$

Ans: 0, 3

3.17 Explain the following terms in relation to Laplace transform

(a) Linearity (b) Scaling (c) Time-shift

(d) Frequency differentiation (e) Time convolution

3.18 State and explain any two properties of Laplace transform.

3.19 Explain the methods of determining the inverse Laplace transform.

3.20 Discuss the concept of transfer function and its applications.

3.21 Determine the inverse Laplace transform of $F(s)=\dfrac{2s^2+3s+3}{(s+1)(s+3)^3}$

3.22 Obtain the inverse Laplace transform of $\dfrac{s}{(s^2+1)^2}$.

Ans: $\dfrac{1}{2}t\sin t$

3.23 Obtain the inverse Laplace transform of the function $F(s)=\dfrac{\beta}{s^2+\beta^2}\left(1-e^{-sT}\right)$

Ans: $f(t)=2\cos\left(\beta t-\dfrac{\beta T}{2}\right)\sin\left(\dfrac{\beta T}{2}\right)$

3.24 Find the inverse Laplace transform of

(i) $F_1(s)=\dfrac{s^2+5}{s^3+2s^2+4s}$ (ii) $F_2(s)=\dfrac{3e^{-s/3}}{s^2(s^2+2)}$

Ans: (i) $f_1(t)=\frac{5}{4}u(t)e^{-t}\left[\frac{1}{4}\cos\sqrt{3}+\frac{9}{4\sqrt{3}}\sin\sqrt{3}t\right]$

(ii) $f_2(t)=\frac{3}{2}f\left(t-\frac{1}{3}\right)-\frac{3}{2\sqrt{2}}\sin\sqrt{2}\left(t-\frac{1}{3}\right)u\left(t-\frac{1}{3}\right)$

3.25 The transfer function of a network is given by $H(s)=\frac{3s}{(s+2)(s^2+s+2)}$
Plot the pole-zero diagram and hence obtain $h(t)$.

Ans: $h(t)=j3e^{-t}t\left(\frac{e^{jt}-e^{-jt}}{j2}\right)-\frac{3}{2}e^{-j2t}$

3.26 Draw the poles and zero for $V(s)=\frac{(s+1)(s+3)}{(s+2)(s+4)}$ and evaluate $v(t)$ by making use of the pole-zero diagram. Confirm the result analytically.

Ans: $v(t)=\delta(t)-\frac{1}{2}e^{-2t}-\frac{3}{2}e^{-4t}$

3.27 The transform current is given by $I(s)=\frac{15s}{(s+1)(s+3)}$. Draw the pole-zero diagram and hence determine $i(t)$.

3.28 Explain the significance of pole-zero diagram in circuit analysis. How will you determine the time domain response from the pole-zero plot?

3.29 The current flowing in a network is given by $I(s)=\frac{s^2+4s+3}{s^2+2s}$. Draw the pole-zero diagram and hence obtain $i(t)$.

3.30 Discuss the criterion for stability in terms of denominator polynomials of the transfer function.

3.31 Determine the poles and zeros of the impedance $Z(s)$ of the network shown in Fig. Q3.31.

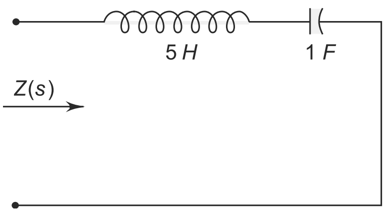


Fig. Q3.31

3.32 From the pole-zero diagram shown in Fig. Q3.32, write an expression for the voltage transform $V(s)$. Determine $v(t)$.

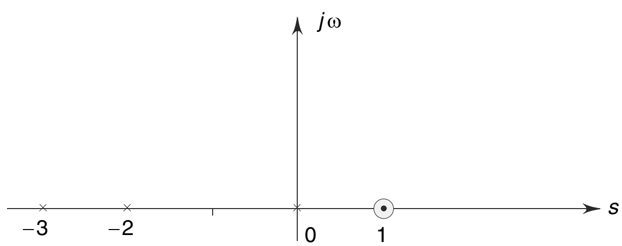


Fig. Q3.32

3.33 Draw the poles and zero diagram for the function $I(s)=\dfrac{4s}{s^2+2s+2}$ and hence determine the amplitude and phase response for $\omega = 0$, 1, 2, 3 and 5.

Ans: $M(j0) = 0$, $M(j1) = 1.79$, $M(j2) = 1.78$
$M(j3) = 1.3$, $M(j5) = 0.8$
$\Phi(j0) = 90°$, $\Phi(j1) = 26.5°$, $\Phi(j2) = -26°$
$\Phi(j3) = -50°$, $\Phi(j5) = -66°$

3.34 Write the expression for the periodic function shown in Fig Q3.34 and evaluate its Laplace transform.

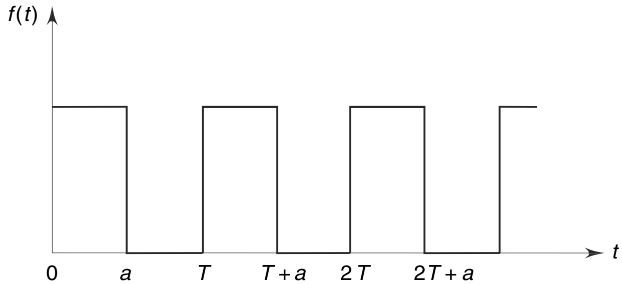


Fig. Q3.34

Ans: $F(s)=\dfrac{1}{s}\cdot\dfrac{1-e^{-\alpha s}}{1-e^{-sT}}$

3.35 Determine the Laplace transform of the periodic rectified half-sine wave as shown in Fig. Q3.35.

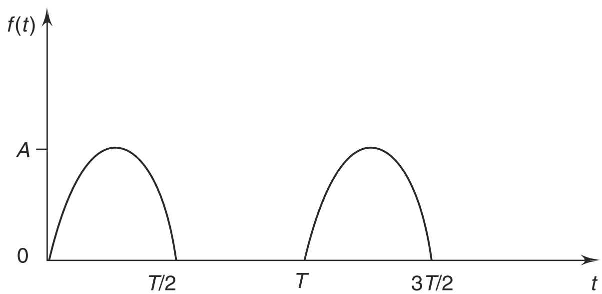


Fig. Q3.35

Ans: $F(s)=\dfrac{1}{\left(1-e^{-sT/2}\right)}\cdot\dfrac{A\omega_0}{s^2+\omega_0^2}$ where $\omega_0=\dfrac{2\pi}{T}$.

3.36 Determine the Laplace transform of the periodic sawtooth waveform, as shown in Fig. Q3.36.

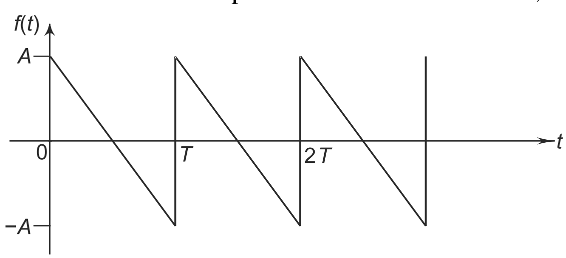


Fig. Q3.36

Ans: $F(s) = \dfrac{2A}{Ts}\left[\dfrac{T}{2}\coth\dfrac{sT}{2} - \dfrac{1}{s}\right]$

3.37 In the network of Fig. Q3.37, determine the current in the inductor L_2 after the switch is closed at $t = 0$. Assume that the voltage source $v(t)$ is applied at $t = -\infty$.

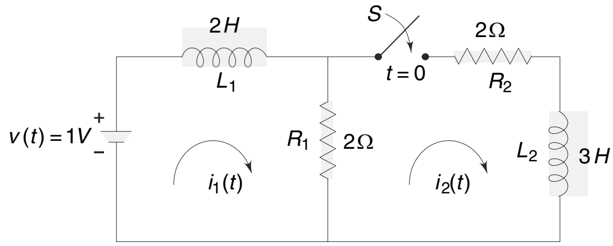


Fig. Q3.37

Ans: $i_2(t) = \left(\dfrac{1}{2} - \dfrac{1}{10}e^{-2t} - \dfrac{2}{5}e^{-t/3}\right)u(t)$

3.38 Derive from first principles, the Laplace transform of a unit-step function. Hence or otherwise determine the Laplace transform of a unit ramp function and a unit impulse function.

3.39 What do you understand by the impulse response of a network? Explain its significance in circuit analysis.

3.40 If impulse response of a network is e^{-at}, what will be its step response?

3.41 The unit step of a network is $(1 - e^{-at})$. Determine the impulse response $h(t)$ of the network.

3.42 The unit step response of a linear system $r(t) = (2e^{-5t} - 1).\ u(t)$. Find (a) the impulse response and (b) the response due to an input $x(t)$ shown in Fig. Q3.42.

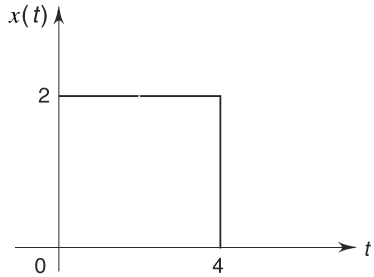


Fig. Q3.42

3.43 Determine the Laplace transform of

$$v(t) = e^{-5t}u(t) - e^{-5(t-1)}u(t - 1)$$

If this voltage is applied to a network whose impedance is $Z(s) = \dfrac{s^2 + 4s + 3}{s(s^2 + 6s + 8)}$, then find the current $I(s)$ and also $i(t)$.

3.44 A sinusoidal voltage 25 sin t is applied at the instant $t = 0$ to an RL circuit with $R = 5\ \Omega$ and $L = 1$ H. Determine $i(t)$ by using Laplace transform method.

3.45 In the circuit shown in Fig. Q3.45, the steady state condition exists with the switch in position 1. The switch is moved to position 2 at $t = 0$. Calculate the current through the coil at the switching instant and current for all values $t > 0$.

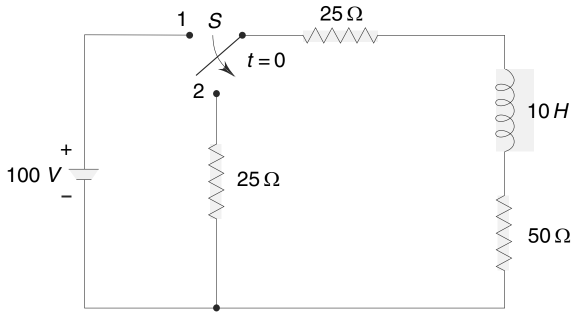


Fig. Q3.45

3.46 In the circuit of Fig. Q3.46, the switch *S* is closed and steady-state conditions have been reached. At $t = 0$, the switch *S* is opened. Obtain the expression for the current through the inductor.

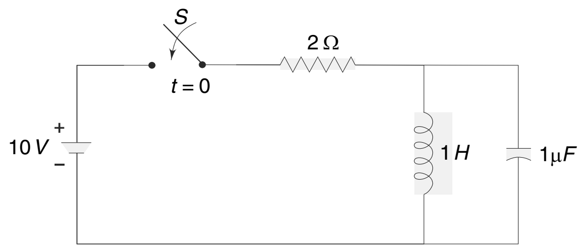


Fig. Q3.46

Ans: $5 \cos 1000t$.

3.47 In the circuit of Fig. Q3.47, the switch *S* is closed at $t = 0$ after the switch is kept open for a long time. Determine the voltage across the capacitor.

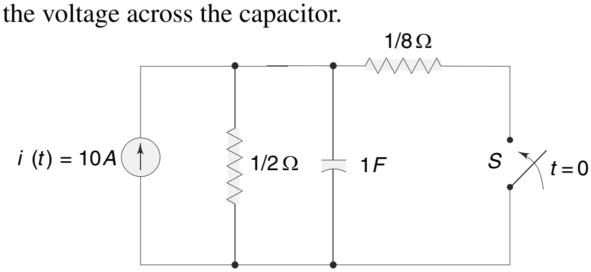


Fig. Q3.47

Ans: $1 + 4e^{-10t}$

3.48 In the circuit shown in Fig. Q3.48, the initial current through *L* is 2 A and initial voltage across *C* is 1 V and the input excitation is $x(t) = \cos 2t$. Obtain the resultant current $i(t)$ and hence $v(t)$ across *C*.

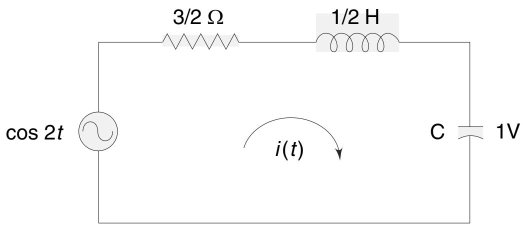


Fig. Q3.48

Ans: $i(t) = -\frac{18}{5}e^{-t} + 5e^{-2t} + \frac{3}{5}\cos 2t + \frac{1}{5}\sin 2t$

$v(t) = -\frac{18}{5}e^{-t} - \frac{5}{2}e^{-2t} + \frac{3}{10}\sin 2t - \frac{1}{10}\cos 2t$

3.49 In the circuit shown in Fig. Q3.47, the initial current is $i_L(0) = 5$ A, initial voltage is $v_c(0) = 10$ V and $x(t) = 10\,u(t)$. Find the voltage $v_c(t)$ across the capacitor for $t > 0$.

Ans: $v_c(t) = 20 - 10e^{-t} - 5e^{-t}$

3.50 Determine the resultant current $i(t)$ when the pulse shown in Fig. Q3.50(a) is applied to the *RL* circuit shown in Fig. Q3.50(b).

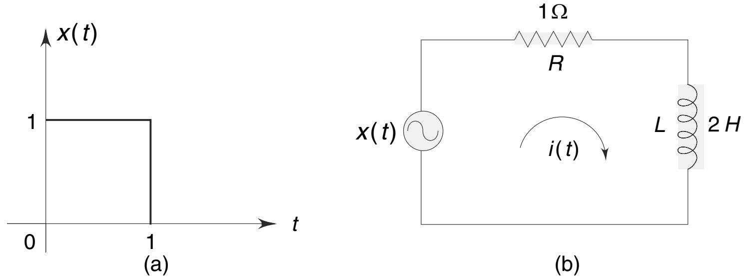


Fig. Q3.50

Ans: $i(t) = (1 - e^{-t/2})u(t) - (1 - e^{-1/2(t-1)}\,u(t-1)$

3.51 An exponential current $2e^{-3t}$ is applied at time $t = 0$ to a parallel *RC* circuit comprising resistor $R = \frac{1}{2}\Omega$ and capacitor $C = 1$ F. Using Laplace transformation, obtain the voltage $v(t)$ across the network. Assume zero charge across the capacitor before the application of current.

Ans: $v(t) = 2e^{-2t} - 2e^{-3t}$

3.52 In the parallel RLC circuit, $I_0 = 5$ amp, $L = 0.2$ H, $C = 2$F, and $R = 0.5$. Switch *S* is opened at time $t = 0$. Obtain the voltage $v(t)$ across the parallel network. Assume zero current through inductor *L* and zero voltage across capacitor *C* before switching.

Ans: $v(t) = \frac{5}{3}e^{-t/2}\sin\frac{3t}{2}$

3.53 A rectangular waveform shown in Fig. Q3.53 (a) is applied to an *RLC* circuit of Fig. Q3.53(b). Obtain the voltage $v(t)$ across the capacitor *C*.

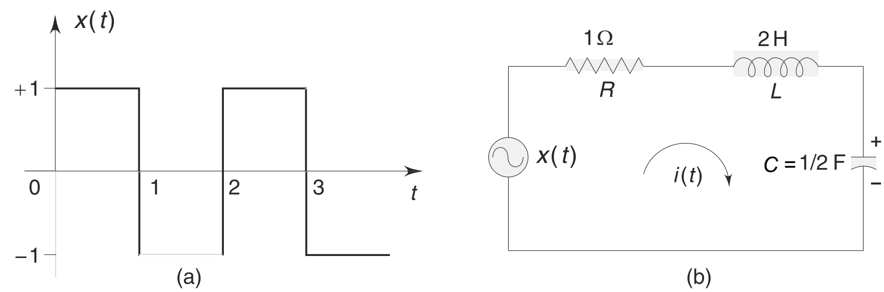


Fig. Q3.53

Chapter 4 z-Transforms

INTRODUCTION 4.1

The Laplace transform plays a very important role in the analysis of analog signals or systems and in solving linear constant coefficient differential equations. It transforms the differential equations into the complex s-plane where algebraic operations and inverse transform can be performed to obtain the solution.

Like the Laplace transform, the z-transform provides the solution for linear constant coefficient difference equations, relating the input and output digital signals in the time domain. It gives a method for the analysis of discrete time systems in the frequency domain.

An analog filter can be described by a frequency domain transfer function of the general form

$$H(s) = \frac{K(s-z_1)(s-z_2)(s-z_3)...}{(s-p_1)(s-p_2)(s-p_3)...}$$

where s is the Laplace variable and K is a constant. The poles p_1, p_2, p_3... and zeros z_1, z_2, z_3... can be plotted in the complex s-plane.

The transfer function $H(z)$ of a digital filter may be described as

$$H(z) = \frac{K(z-z_1)(z-z_2)(z-z_3)...}{(z-p_1)(z-p_2)(z-p_3)...}$$

Here the variable z is not the same as the variable s. For example, the frequency response of a digital filter is determined by substituting $z = e^{j\omega}$; but the equivalent substitution in the analog case is $s = j\omega$, where ω is the angular frequency in radians per second. Another essential difference is that the frequency response of an analog filter is not a periodic function. The transfer function $H(s)$ is converted into a transfer function $H(z)$, so that the frequency response of the digital filter over the range $0 \le \omega \le \pi$ approximates that of the analog filter over the range $0 \le \omega \le \infty$.

The analysis of any sampled signal or sampled data system in the frequency domain is extremely difficult using s-plane representation because the signal or system equations will contain infinite long polynomials due to the characteristic infinite number of poles and zeros. Fortunately, this problem may be overcome by using the z-transform, which reduces the poles and the zeros to a finite number in the z-plane.

The purpose of the z-transform is to map (transform) any point $s = \pm\, \sigma \pm j\omega$ in the s-plane to a corresponding point $z(r\underline{|\theta})$ in the z-plane by the relationship

$$z = e^{sT}, \text{ where } T \text{ is the sampling period (seconds)}$$

Table 4.1 *Values of Z in Polar Form*

$\sigma = 0,\ \omega_s = \frac{2\pi}{T}$										
$j\omega$	0	$\omega_s/8$	$\omega_s/4$	$3\omega_s/8$	$\omega_s/2$	$5\omega_s/8$	$3\omega_s/4$	$7\omega_s/8$	ω_s
$Z = 1\angle\omega T$	$1\angle 0°$	$1\angle 45°$	$1\angle 90°$	$1\angle 135°$	$1\angle 180°$	$1\angle 225°$	$1\angle 270°$	$1\angle 315°$	$1\angle 360°$

Under this mapping, the imaginary axis, $\sigma = 0$ maps on to the unit circle $|z| = 1$ in the z-plane. Also, the left hand half-plane $\sigma < 0$ corresponds to the interior of the unit circle $|z| = 1$ in the z-plane. This correspondence is shown in Fig. 4.1.

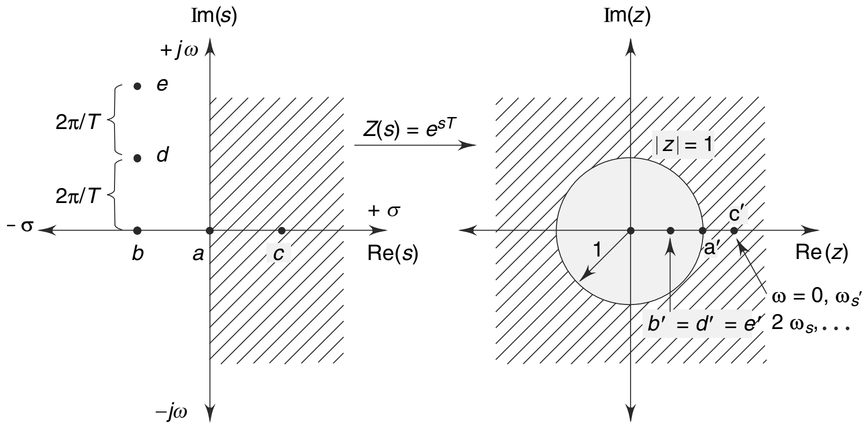


Fig. 4.1 *Mapping of s-plane to z-plane for $z = e^{j\omega T}$*

Considering that the real part of x is zero, i.e. $\sigma = 0$, we have $z = e^{j\omega T} = 1\angle \pm j\omega T$, which gives the values of z (in polar form) shown as in Table 4.1.

We know that the Laplace transform gives

$$\mathcal{L}[x^*(t)] = X(s) = \sum_{n=0}^{\infty} x(nT)e^{-nsT}$$

But we have $z = e^{sT}$ for the z-transform and hence $z^{-n} = e^{-nsT}$, which corresponds to a delay of n sampling periods. Therefore, the z-transform of $x^*(t)$ is

$$Z[x^*(t)] = X(z) = \sum_{n=0}^{\infty} x(nT)z^{-n}$$

In order to simplify the notation, the nth sample may be written as $x(n)$ and hence the above equation becomes

$$X(z) = \sum_{n=0}^{\infty} x(n)z^{-n} \tag{4.1}$$

Evaluating $X(z)$ at the complex number $z = re^{j\omega}$ gives

$$X(z)\big|_{z=re^{j\omega}} = \sum_{n=-\infty}^{\infty} x(n)(re^{j\omega})^{-n} = \sum_{n=-\infty}^{\infty} [x(n)r^{-n}]e^{-j\omega n}$$

Therefore, if $r = 1$, the z-transform evaluated on the unit circle gives the Fourier transform of the sequence $x(n)$.

The z-transform method is used to analyse discrete-time systems for finding the transfer function, stability and digital network realisations of the system.

Example 4.1 *Determine the z-transform for the analog input signal $x(t) = e^{-at}$ applied to a digital filter.*

Solution We know that

$$x^*(t) = \sum_{n=-\infty}^{\infty} x(t)\delta(t-nT)$$

For $t = nT$, the sampled signal sequence is

$$x^*(nT) = [e^0, e^{-aT}, e^{-2aT}, e^{-3aT}, \ldots]$$

By applying Eq. (4.1) of $X(z) = \sum_{n=0}^{\infty} x(n)z^{-n}$, we get

$$X(z) = 1 + e^{-aT}z^{-1} + e^{-2aT}z^{-2} + e^{-3aT}z^{-3} + \ldots$$

$$= \sum_{n=0}^{\infty} e^{-anT}z^{-n} = \sum_{n=0}^{\infty} (e^{-anT}z^{-1})^n$$

Using infinite summation formula of $\sum_{k=0}^{\infty} a^k = \frac{1}{1-a}, |a| < 1$, we get

$$X(z) = \frac{1}{1-e^{-aT}z^{-1}}$$

Multiplying the numerator and denominator by z, we have

$$X(z) = \frac{z}{z-e^{-aT}}$$

DEFINITION OF THE z-TRANSFORM 4.2

The z-transform of a discrete-time signal $x(n)$ is defined as the power series

$$Z[x(n)] = X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \quad (4.2)$$

where z is a complex variable. This expression is generally referred to as the *two-sided z-transform*.

If $x(n)$ is a causal sequence, $x(n) = 0$ for $n < 0$, then its z-transform is

$$X(z) = \sum_{n=0}^{\infty} x(n)z^{-n}$$

This expression is called a *one-sided z-transform*.

This causal sequence produces negative powers of z in $X(z)$. Generally we assume that $x(n)$ is a causal sequence, unless it is otherwise stated.

If $x(n)$ is a non-causal sequence, $x(n) = 0$ for $n \geq 0$, then its z-transform is

$$X(z) = \sum_{n=-\infty}^{-1} x(n)z^{-n} \quad (4.3)$$

This expression is also called a one-sided z-transform. This non-causal sequence produces positive powers of z in $X(z)$.

4.2.1 Definition of the Inverse z-Transform

The inverse z-transform is computed or derived to recover the original time domain discrete signal sequence $x(n)$ from its frequency domain signal $X(z)$. The operation can be expressed mathematically as

$$x(n) = Z^{-1}[X(z)] \tag{4.4}$$

4.2.2 Region of Convergence (ROC)

Equation (4.1) gives

$$X(z)\big|_{z=re^{j\omega}} = X(re^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n) r^{-n} e^{-j\omega n}$$

which is the Fourier transform of the modified sequence $[x(n)r^{-n}]$. If $r = 1$, i.e. $|z| = 1$, $X(z)$ reduces to its Fourier transform. The series of the above equation converges if $x(n)r^{-n}$ is absolutely summable, i.e.

$$\sum_{n=-\infty}^{\infty} \left| x(n) r^{-n} \right| < \infty$$

If the output signal magnitude of the digital signal system, $x(n)$, is to be finite, then the magnitude of its z-transform, $X(z)$, must be finite. The set of z values in the z-plane, for which the magnitude of $X(z)$ is finite, is called the *Region of Convergence* (ROC). That is, convergence of $\sum_{n=0}^{\infty} \left| x(n) r^{-n} \right|$ guarantees convergence of Eq. (4.1), where $X(z)$ is a function of z^{-n}. Therefore, the condition for $X(z)$ to be finite is $|z| > 1$. In other words, the ROC for $X(z)$ is the area outside the unit circle in the z-plane.

The ROC of a rational z-transform is bounded by the location of its poles. For example, the z-transform of the unit step response $u(n)$ is $X(z) = \dfrac{z}{z-1}$ which has a zero at $z = 0$ and a pole at $z = 1$ and the ROC is $|z| > 1$ and extending all the way to ∞, as shown in Fig. 4.2.

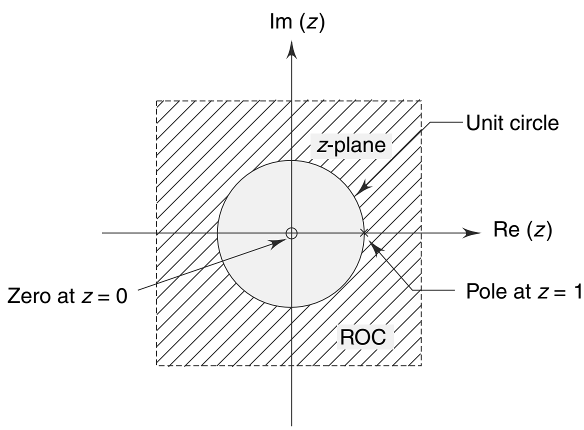


Fig. 4.2 *Pole-Zero Plot and ROC of the Unit-Step Response u(n)*

Important Properties of the ROC for the z-Transform

(i) $X(z)$ converges uniformly if and only if the ROC of the z-transform $X(z)$ of the sequence includes the unit circle. The ROC of $X(z)$ consists of a ring in the z-plane centered about the origin. That is, the ROC of the z-transform of $x(n)$ has values of z for which $x(n)\, r^{-n}$ is absolutely summable.

$$\sum_{n=-\infty}^{\infty} \left|x(n)r^{-n}\right| < \infty$$

(ii) The ROC does not contain any poles.

(iii) When $x(n)$ is of finite duration, then the ROC is the entire z-plane, except possibly $z = 0$ and / or $z = \infty$.

(iv) If $x(n)$ is a right-sided sequence, the ROC will not include infinity.

(v) If $x(n)$ is a left-sided sequence, the ROC will not include $z = 0$. However, if $x(n) = 0$ for all $n > 0$, the ROC will include $z = 0$.

(vi) If $x(n)$ is two-sided, and if the circle $|z| = r_0$ is in the ROC, then the ROC will consist of a ring in the z-plane that includes the circle $|z| = r_0$. That is, the ROC includes the intersection of the ROC's of the components.

(vii) If $X(z)$ is rational, then the ROC extends to infinity, i.e. the ROC is bounded by poles.

(viii) If $x(n)$ is causal, then the ROC includes $z = \infty$.

(ix) If $x(n)$ is anti-causal, then the ROC includes $z = 0$.

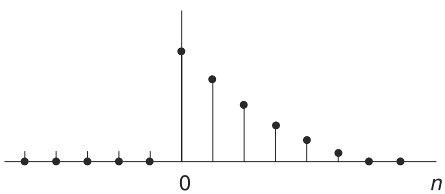
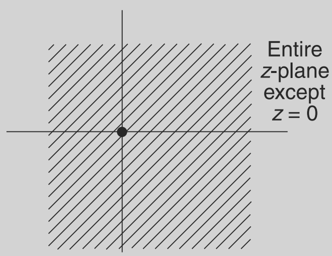
To determine the ROC for the series expressed by the Eq. (4.2), which is referred to as the two-sided z-transform, this equation can be written as

$$\sum_{n=-\infty}^{\infty} x(n)r^{-n} = \sum_{n=-\infty}^{-1} x(n)z^{-n} + \sum_{n=0}^{\infty} x(n)z^{-n}$$

$$= \sum_{n=1}^{\infty} x(-n)z^{+n} + \sum_{n=0}^{\infty} x(n)z^{-n}$$

The first series, a non-causal sequence, converges for $|z| < r_2$, and the second series, a causal sequence, converges for $|z| > r_1$, resulting in an annular region of convergence. Then Eq. (4.2) converges for $r_1 < |z| < r_2$. provided $r_1 < r_2$. The causal, anti-causal and two-sided signals with their corresponding ROCs are shown in Table 4.2. Some important commonly used z-transform pairs are given in Table 4.3.

Table 4.2 *The Causal, anti-causal and two-sided signals and their ROCs*

Signals	*ROCs*
(a) Finite duration signals Causal  	

(*Contd...*)

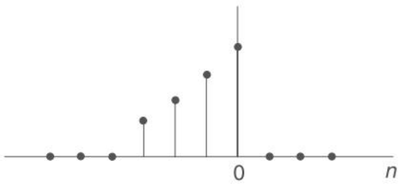
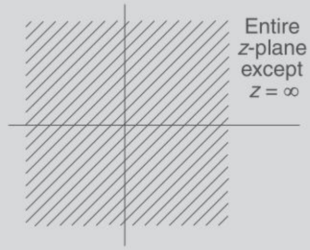
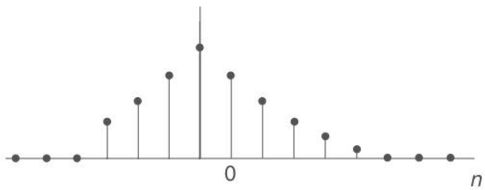
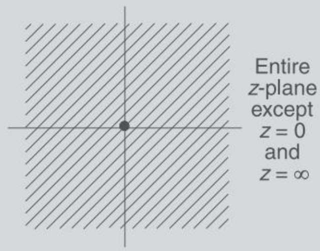
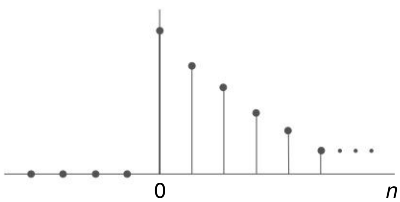
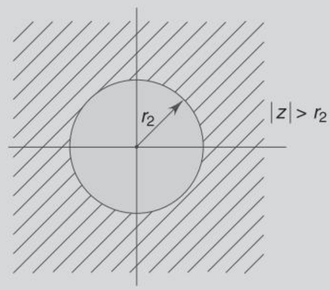
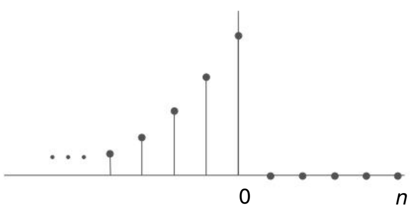
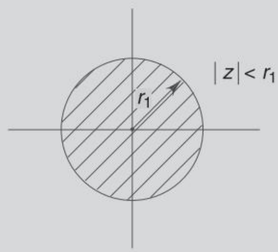
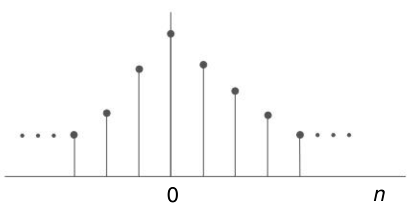
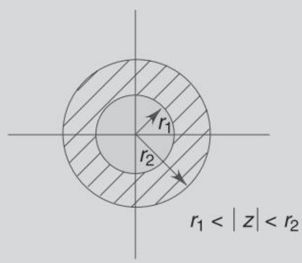
Signals	*ROCs*
Anti-causal 	
Two-sided 	
(b) Infinite duration signals Causal 	
Anti-causal 	
Two-sided 	

Table 4.3 *Some important z-transform pairs*

S. No.	Signal $x(t)$	Sequence $x(n)$	Laplace transform $X(s)$	z-transform $X(z)$	ROC
1.	$\delta(t)$	$\delta(n)$		1	All z-plane
2.	$\delta(t-k)$	$\delta(n-k)$	e^{-ks}	z^{-k}	$\|z\| > 0, k > 0$ $\|z\| < \infty, k < 0$
3.	$u(t)$	$u(n)$	$\frac{1}{s}$	$\frac{1}{1-z^{-1}} = \frac{z}{z-1}$	$\|z\|>1$
4.		$-u(-n-1)$	$\frac{1}{s}$	$\frac{1}{1-z^{-1}} = \frac{z}{z-1}$	$\|z\|<1$
5.	$tu(t)$	$nu(n)$	$\frac{1}{s^2}$	$\frac{z^{-1}}{(1-z^{-1})} = \frac{z}{(z-1)}$	$\|z\|>1$
6.		$a^n u(n)$		$\frac{1}{1-az^{-1}} = \frac{z}{(z-a)}$	$\|z\|>\|a\|$
7.		$-a^n u(-n-1)$		$\frac{1}{1-az^{-1}} = \frac{z}{(z-a)}$	$\|z\|<\|a\|$
8.		$na^n u(n)$		$\frac{az}{(z-a)^2}$	$\|z\|>\|a\|$
9.		$-na^n u(-n-1)$		$\frac{az}{(z-a)^2}$	$\|z\|<\|a\|$
10.	e^{-at}	e^{-an}	$\frac{1}{(s+a)}$	$\frac{1}{1-e^{-a}z^{-1}} = \frac{z}{z-e^{-a}}$	$\|z\|>\|e^{-a}\|$
11.	t^2	$n^2u(n)$	$\frac{2}{s^3}$	$z^{-1}\frac{(1+z^{-1})}{(1-z^{-1})^3} = \frac{z(z+1)}{(z-1)^3}$	$\|z\|>1$
12.	te^{-at}	ne^{-an}	$\frac{1}{(s+a)^2}$	$\frac{z^{-1}e^{-a}}{(1-e^{-a}z^{-1})^2} = \frac{ze^{-a}}{(z-e^{-a})^2}$	$\|z\|>\|e^{-a}\|$
13.	$\sin\omega_0 t$	$\sin\omega_0 n$	$\frac{\omega_0}{s^2+\omega_0^2}$	$\frac{z\sin\omega_0}{z^2-2z\cos\omega_0+1}$	$\|z\|>1$
14.	$\cos\omega_0 t$	$\cos\omega_0 n$	$\frac{s}{s^2+\omega_0^2}$	$\frac{z(z-\cos\omega_0)}{z^2-2z\cos\omega_0+1}$	$\|z\|>1$
15.	$\sinh\omega_0$	$\sinh\omega_0 n$	$\frac{\omega_0}{s^2+\omega_0^2}$	$\frac{z\sinh\omega_0}{z^2-2z\cosh\omega_0+1}$	$\|z\|>$

(Contd...)

(Contd...)

16.	$\cosh\omega_0 t$	$\cosh\omega_0 n$	$\dfrac{s}{s^2-\omega_0^2}$	$\dfrac{z(z-\cosh\omega_0)}{z^2-2z\cosh\omega_0+1}$	$\lvert z\rvert>1$
17.	$e^{-at}\sin\omega_0 t$	$e^{-at}\sin\omega_0 n$	$\dfrac{\omega_0}{(s+a)^2+\omega_0^2}$	$\dfrac{ze^{-a}\sin\omega_0}{z^2-2ze^{-a}\cos\omega_0+e^{-2a}}$	$\lvert z\rvert>\lvert e^{-a}\rvert$
18.	$e^{-at}\cos\omega_0 t$	$e^{-at}\cos\omega_0 n$	$\dfrac{s+a}{(s+a)^2+\omega_0^2}$	$\dfrac{z(z-e^{-a}\cos\omega_0)}{z^2-2ze^{-a}\cos\omega_0+e^{-2a}}$	$\lvert z\rvert>\lvert e^{-a}\rvert$
19.		$a^n\sin\omega_0 n$		$\dfrac{za\sin\omega_0}{z^2-2za\cos\omega_0+a^2}$	$\lvert z\rvert>\lvert a\rvert$
20.		$a^n\cos\omega_0 n$		$\dfrac{z(z-a\sin\omega_0)}{z^2-2za\cos\omega_0+a^2}$	$\lvert z\rvert>\lvert a\rvert$

Sequence Representation

A signal or sequence at time origin ($n = 0$) is indicated by the symbol ↑. If the sequence is not indicated by ↑, then it is understood that the first (leftmost) point in the sequence is at the time origin.

Example 4.2 *Determine the z-transform of the following finite duration signals.*

(a) $x(n)=\left\{3,\ 1,\ 2,\ \underset{\uparrow}{5},\ 7,\ 0,\ 1\right\}$ (b) $x(n)=\left\{2,\ 4,\ \underset{\uparrow}{5},\ 7,\ 0,\ 1,\ 2\right\}$

(c) $x(n) = \{1, 2, 5, 4, 0, 1\}$ (d) $x(n) = \{0, 0, 1, 2, 5, 4, 0, 1\}$

(e) $x(n) = \delta(n)$ (f) $x(n) = \delta(n - k)$

(g) $x(n) = \delta(n + k)$

Solution

(a) $x(n)=\left\{3,\ 1,\ 2,\ \underset{\uparrow}{5},\ 7,\ 0,\ 1\right\}$

Taking z-transform, we get

$$X(z) = 3z^3 + z^2 + 2z + 5 + 7z^{-1} + z^{-3}.$$

ROC: Entire z-plane except $z = 0$ and $z = \infty$.

(b) $x(n)=\left\{2,\ 4,\ \underset{\uparrow}{5},\ 7,\ 0,\ 1,\ 2\right\}$

Taking z-transform, we get

$$X(z) = 2z^2 + 4z + 5 + 7z^{-1} + z^{-3} + 2z^{-4}.$$

ROC: Entire z-plane except $z = 0$ and $z = \infty$

(c) $x(n) = \{1, 2, 5, 4, 0, 1\}$

Taking z-transform, we get

$$X(z) = 1 + 2z^{-1} + 5z^{-2} + 4z^{-3} + z^{-5}$$

ROC: Entire z-plane except $z = 0$.

(d) $x(n) = \{0, 0, 1, 2, 5, 4, 0, 1\}$

Taking z-transform, we get

$$X(z) = z^{-2} + 2z^{-3} + 5z^{-4} + 4z^{-5} + z^{-7}.$$

ROC: Entire z-plane except $z = 0$.

(e) $x(n) = \delta(n)$, hence $X(z) = 1$, ROC: Entire z-plane.

(f) $x(n) = \delta(n - k)$, $k > 0$, hence $X(z) = z^{-k}$, ROC: Entire z-plane except $z = 0$

(g) $x(n) = \delta(n + k)$, $k > 0$, hence $X(z) = z^{k}$, ROC: Entire z-plane except $z = \infty$.

Example 4.3 *Determine the z-transform including the region of convergence of*

$$x(n) = \begin{cases} a^n, & n \ge 0 \\ 0, & n < 0 \end{cases}$$

Solution The z-transform for the given $x(n)$ is

$$X(z) = Z[a^n] = \sum_{n=-\infty}^{\infty} a^n z^{-n} = \sum_{n=0}^{\infty} (az^{-1})^n$$

We know that $\sum_{0}^{\infty} a^n = \frac{1}{1-a}, \; |a| < 1$

Hence, $X(z) = \frac{1}{1 - az^{-1}} = \frac{z}{z - a}$

This converges when $|az^{-1}| < 1$ or $|z| > |a|$. Values of z for which $X(z) = 0$ are called zeros of $X(z)$, and values of z for which $X(z) \to \infty$ are called poles of $X(z)$.

Here the poles are at $z = a$ and zeros at $z = 0$. The region of convergence is shown in Fig. E4.3.

Pole-zero Representation

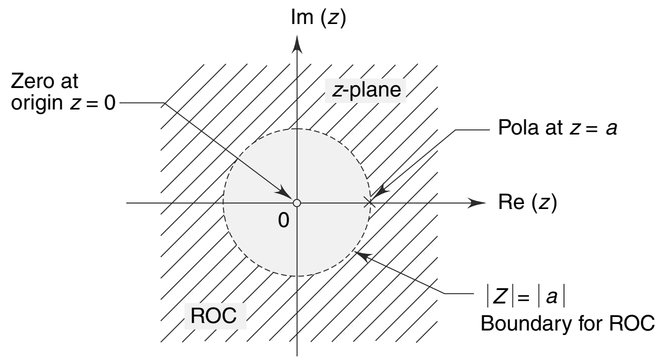


Fig. E4.3 *ROC for the z-transform of x(n) = a^n*

The transfer function $H(z)$ of a discrete-time system can be expressed in terms of its poles and zeros.

$$H(z) = \frac{N(z)}{D(z)}$$

where

$$N(z) = a_0 z^N + a_1 z^{N-1} + a_2 z^{N-2} + \dots + a_N$$

$$D(z) = b_0 z^N + b_1 z^{N-1} + b_2 z^{N-2} + \dots + b_N$$

Here a_k and b_k are the coefficients of the filter.

If $H(z)$ has poles at $z = p_1, p_2,\ldots, p_N$ and zeros at $z = z_1, z_2,\ldots, z_N$, then $H(z)$ can be factored and represented as

$$H(z) = \frac{K(z-z_1)(z-z_2)...(z-z_N)}{(z-p_1)(z-p_2)...(z-z_N)}$$

where z_i is the ith zero, p_i is the ith pole and K is the gain factor. The poles of $H(z)$ are the values of z for which $H(z)$ becomes infinity. The values of z for which $H(z)$ becomes zero are referred to as zeros. The poles and zeros of $H(z)$ may be real or complex. When they are complex, they occur in conjugate pairs, so that the coefficients a_k and b_k are real. If the locations of the poles and zeros of $H(z)$ are known, then $H(z)$ can be readily reconstructed. The stability of the system can be determined using the locations of the poles of $H(z)$.

Example 4.4 *Sketch the pole-zero diagram for the transfer function*

$$H(z) = \frac{1-z^{-1}-2z^{-2}}{1-1.75z^{-1}+1.25z^{-2}-0.375z^{-3}}$$

Solution First, $H(z)$ has to be expressed in positive powers of z and then factorised to determine its poles and zeros. Multiplying the denominator and numerator by z^3, the highest power of z, we obtain

$$H(z) = \frac{z(z^2-z-2)}{z^3-1.75z^2+1.25z-0.375}$$

Factorising, we have

$$H(z) = \frac{z(z-2)(z+1)}{(z-0.5+j0.5)(z-0.5-j0.5)(z-0.75)}$$

Hence, the pole locations are at $z = 0.5 \pm j0.5$ and at $z = 0.75$. The zeros are at $z = 0$, $z = 2$ and $z = -1$. The pole-zero diagram is shown in the Fig. E4.4.

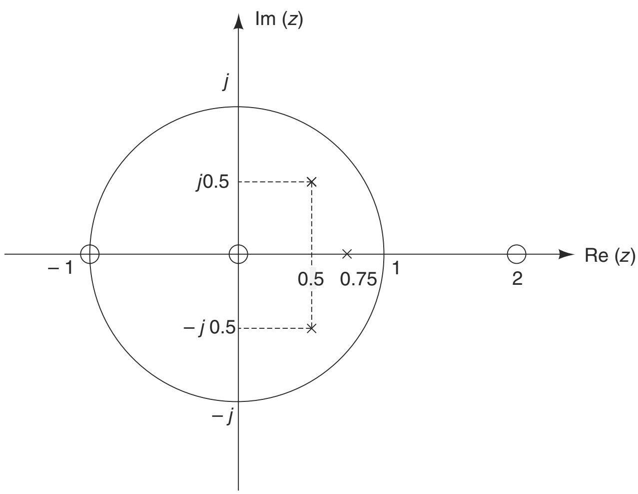


Fig. E4.4 *Pole-zero Diagram*

Example 4.5 *Determine the transfer function H(z) of a discrete-time filter with the pole-zero diagram shown in Fig. E4.5.*

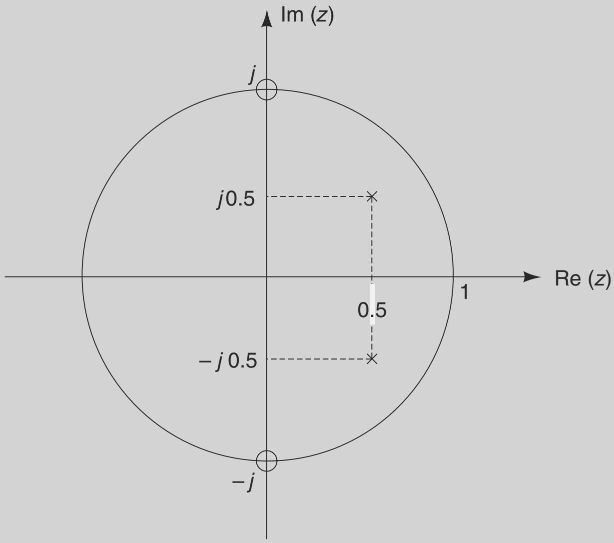


Fig. E4.5 *Pole-zero Diagram*

Solution From the pole-zero diagram, the zeros of the transfer function are at $z = \pm j$ and the poles are at $z = 0.5 \pm j0.5$. The transfer function $H(z)$ can be written down directly as

$$H(z) = \frac{K(z-j)(z+j)}{(z-0.5-j0.5)(z-0.5-j0.5)}$$

$$= \frac{K(z^2+1)}{z^2-z+0.5}$$

$$= \frac{K(1+z^{-2})}{1-z^{-1}-0.5z^{-2}}$$

PROPERTIES OF *z*-TRANSFORM 4.3

A number of useful theorems for z-transforms are presented and discussed in this section. These are summarised in Table 4.4.

4.3.1 Linearity

If $x_1(n) \overset{z}{\longleftrightarrow} X_1(z)$ and $x_2(n) \overset{z}{\longleftrightarrow} X_2(z)$, then

$$x(n) = a_1 x_1(n) + a_2 x_2(n) \overset{z}{\longleftrightarrow} X(z) = a_1 X_1(z) + a_2 X_2(z) \tag{4.5}$$

where a_1 and a_2 are arbitrary constants. It implies that the z-transform of a linear combination of signals is the same as the linear combination of their z-transforms.

Table 4.4 *Properties of z-Transform*

S. No.	*Property or operation*	*Signal*	*z-transform*
1.	Transformation	$x(n)$	$X(z)=\sum_{n=0}^{\infty}x(n)z^{-n}$
2.	Inverse transformation	$\frac{1}{2\pi j}\int X(z)z^{n-1}dz$	$X(z)$
3.	Linearity	$a_1\,x_1\,(n)+a_2\,x_2\,(n)$	$a_1\,X_1\,(z)+a_2\,X_2\,(z)$
4.	Time reversal	$x\,(-\,n)$	$X\,(z^{-1})$
5.	Time shifting	(i) $x\,(n-k)$ (ii) $x\,(n+k)$	(i) $z^{-k}\,X(z)$ (ii) $z^{k}\,X(z)$
6.	Convolution	$x_1(n)*x_2(n)$	$X_1(z)\,X_2(z)$
7.	Correlation	$r_{x_1x_2}(l)=\sum_{n=-\infty}^{\infty}x_1(n)x_2(n-l)$	$R_{x1\,x2}\,(z)=X_1(z)\,X_2(z^{-1})$
8.	Scaling	$a^{\mathrm{n}}x(n)$	$X\,(a^{-1}z)$
9.	Differentiation	$nx\,(n)$	$z^{-1}\frac{\mathrm{d}X(z)}{\mathrm{d}z^{-1}}$ or $-z\frac{\mathrm{d}X(z)}{\mathrm{d}z}$
10.	Time differentiation	$x(n)-x\,(n-1)$	$X\,(z)\,(1-z^{-1})$
11.	Time integration	$\sum_{k=0}^{\infty}X(k)$	$X(z)=\left(\frac{z}{z-1}\right)$
12.	Initial value theorem	$\lim_{n\to 0}x(n)$	$\lim_{\lvert z\rvert\to\infty}X(z)$
13.	Final value theorem	$\lim_{n\to\infty}x(n)$	$\lim_{\lvert z\rvert\to 1}\left(\frac{z-1}{z}\right)X(z)$

Example 4.6 *Determine the z-transform of the signal*

$$x(n) = \delta(n+1) + 3\delta(n) + 6\delta(n-3) - \delta(n-4).$$

Solution From the linearity property, we have

$$X(z) = Z\{\delta(n+1)\} + 3Z\{\delta(n)\} + 6Z\{\delta(n-3)\} - Z\{\delta(n-4)\}$$

Using the z-transform pairs, we obtain

$$X(z) = z + 3 + 6z^{-3} - z^{-4}$$

Therefore, $x(n)=\left\{\begin{matrix}1, 3, 0, 0, 6, -1\\ \uparrow\end{matrix}\right\}$

The ROC is the entire z-plane except $z = 0$ and $z = \infty$.

The same result can be obtained by using the definition of the transform.

Example 4.7 *Find the z-transform of x(n) =* $\cos \omega_0 n$ *for* $n \geq 0$

Solution $x(n) = \cos\omega_0 n = \frac{1}{2}[e^{j\omega_0 n} + e^{-j\omega_0 n}]$

Using the transform, for $n \geq 0$,

$$Z[a^n] = \frac{1}{1-az^{-1}}, |z| > a$$

Therefore, for $n \geq 0$, $Z[(e^{j\omega_0})^n] = \frac{1}{1-e^{j\omega_0}z^{-1}}, \ |z| > 1$

Similarly for $n \geq 0$, $Z[(e^{-j\omega_0})^n] = \frac{1}{1-e^{j\omega_0}z^{-1}}, \ |z| > 1$

Therefore, $X(z) = Z[\cos\omega_0 n] = Z\left[\frac{1}{2}(e^{j\omega_0 n} + e^{-j\omega_0 n})\right]$

$$= \frac{\frac{1}{2}}{1-e^{j\omega_0}z^{-1}} + \frac{\frac{1}{2}}{1-e^{-j\omega_0}z^{-1}} = \frac{1-\frac{1}{2}[e^{j\omega_0}+e^{-j\omega_0}]z^{-1}}{(1-e^{j\omega_0}z^{-1})(1-e^{-j\omega_0}z^{-1})}$$

$$= \frac{1-z^{-1}\cos\omega_0}{1-2z^{-1}\cos\omega_0 + z^{-2}} = \frac{z(z-\cos\omega_0)}{z^2-2z\cos\omega_0+1}, \ |z| > 1$$

Similarly, we can find $Z[\sin \omega_0 n]$ using the property of linearity, i.e.

$$Z[\sin\omega_0 n] = Z\left[\frac{1}{2j}(e^{j\omega_0 n} - e^{-j\omega_0 n})\right]$$

$$= \frac{z^{-1}\sin\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}} = \frac{z\sin\omega_0}{z^2-2z\cos\omega_0+1}, \ |z| > 1$$

4.3.2 Time Reversal

If $x(n) \overset{z}{\longleftrightarrow} X(z)$; ROC: $r_1 < |z| < r_2$,

then, $x(-n) \overset{z}{\longleftrightarrow} X(z^{-1})$; ROC: $\frac{1}{r_2} < |z| < \frac{1}{r_1}$ (4.6)

Proof From the definition of z-transform, we have

$$Z[x(-n)] = \sum_{n=-\infty}^{\infty} x(-n)z^{-n} = \sum_{l=-\infty}^{\infty} x(l)(z^{-1})^{-l} = X(z^{-1})$$

where the change of variable $l = -n$ is made. The ROC of $X(z^{-1})$ is

$$r_1 < |z^{-1}| < r_2 \quad \text{or} \quad \frac{1}{r_2} < |z| < \frac{1}{r_1}$$

Here the ROC for $x(n)$ is the inverse of that for $x(-n)$. This means that if z belongs to the ROC of $x(n)$, then $1/z$ is in the ROC for $x(-n)$.

When a signal is folded, the coefficient of z^{-n} becomes the coefficient z^n. Thus, folding a signal is equivalent to replacing z by z^{-1} in the z-transform formula. In other words, reflection in the time domain corresponds to inversion in the z-domain.

Example 4.8 *Find the z-transform of the signal $x(n) = u(-n)$*

Solution We know that $u(n) \xleftrightarrow{z} \frac{z}{z-1}$; ROC: $|z| > 1$

By using the time reversal property, we obtain

$$u(-n) \xleftrightarrow{z} \frac{z^{-1}}{z^{-1}-1} = \frac{1}{1-z}; \text{ ROC: } |z| < 1$$

Example 4.9 *Determine the z-transform of the sequence given by*

$$x(n) = \begin{cases} 2^n, & n < 0 \\ \left(\frac{1}{2}\right)^n, & n = 0, 2, 4 \\ \left(\frac{1}{3}\right)^n, & n = 1, 3, 5 \end{cases}$$

Solution

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n}$$

$$= \sum_{n=-\infty}^{-1} 2^n z^{-n} + \sum_{n=0(n-even)}^{\infty} \left(\frac{1}{2}\right)^n z^{-n} + \sum_{n=0(n-odd)}^{\infty} \left(\frac{1}{3}\right)^n z^{-n}$$

$$= \sum_{m=1}^{\infty} 2^{-m} z^m + \sum_{p=0}^{\infty} \left(\frac{1}{2}\right)^{2p} z^{-2p} + \sum_{q=0}^{\infty} \left(\frac{1}{3}\right)^{2q+1} z^{-(2q+1)}$$

where $m = -n$, $p = \frac{n}{2}$ and $q = \frac{(n-1)}{2}$

Therefore, $X(z) = X_1(z) + X_2(z) + X_3(z)$

$$= \frac{\frac{z}{2}}{1-\frac{z}{2}} + \frac{z^2}{z^2 - \frac{1}{4}} + \frac{\frac{z}{3}}{z^2 - \frac{1}{9}}$$

ROC for $X_1(z)$: $|z| < 2$

ROC for $X_2(z)$: $|z| > \frac{1}{2}$

ROC for $X_3(z)$: $|z| > \frac{1}{3}$

Hence, ROC for $X(z)$: $\frac{1}{2} < |z| < 2$

4.3.3 Time Shifting

Time Delay (Right-Shifting)

$$Z\{x(n-k)\} = z^{-k} Z\{x(n)\} = z^{-k} X(z) \quad (4.7a)$$

Proof From the definition of the z-transform, we have

$$X(z) = \sum_{n=0}^{\infty} x(n) z^{-n}$$

Multiplying throughout by z^{-k}, we get

$$z^{-k} X(z) = \sum_{n=0}^{\infty} x(n) z^{-n-k}$$

Substituting $-m$ for $-n-k$, we get

$$z^{-k} X(z) = \sum_{m=k}^{\infty} x(m-k) z^{-m} = \sum_{m=0}^{\infty} x(m-k) = \sum_{n=0}^{\infty} x(n-k) z^{-n}$$

where, m is changed to letter n, since m is a dummy index. The third term in this expression was obtained by invoking the one-sided character of $x(k)$ with x (negative number) = 0. For the case when $x(-n)$ has values, the quantity $x(-k) + x(1-k) z^{-1} + \ldots + x(-1) z^{-(k-1)}$ is added to the right-hand side of the above equation and hence m runs from 0 to $k - 1$. The multiplication by z^{-k} $(k > 0)$ creates a pole at $z = 0$ and deletes a pole at infinity. Therefore, the ROC of $z^{-k}(z)$ is the same as that of $X(z)$ except for $z = 0$ if $k > 0$ and $z = \infty$ if $k < 0$.

Time Advance (Left-Shifting)

$$Z\{x(n+k)\} = z^k Z\{x(n)\} = z^k X(z) \tag{4.7b}$$

Proof From the basic definition

$$Z\{x(n+1)\} = \sum_{n=0}^{\infty} x(n+1) z^{-n}$$

Now setting $n + 1 = m$, we find that

$$Z\{x(m)\} = \sum_{m=1}^{\infty} x(m) z^{-m+1} = z \sum_{m=1}^{\infty} x(m)^{-m} - zx(0) = zX(z) - zx(0)$$

By a similar procedure, we can show that

$$Z\{x(n+k)\} = z^k X(z) - z^k x(0) - z^{k-1} x(1) - \ldots - zx(k-1)$$
$$= z^n X(z) - \sum_{n=0}^{k-1} x(n) z^{k-n}$$

Because of the factor z^k, zeros are introduced at $z = 0$ and at infinity. Observe that if the function is defined by

$$x(n+k) = \begin{cases} 0, & \text{for } n = -k, -k-1, -k-2, \ldots \\ f(n+k), & \text{for } n = 0, 1, 2, \ldots \end{cases}$$

then the shifting property gives

$$Z\{x(n+k)\} = z^k X(z)$$

Example 4.10 *By applying the time shifting property, determine the inverse z-transform of the signal*

$$X(z) = \frac{z^{-1}}{1 - 3z^{-1}}$$

Solution $X(z) = \frac{z^{-1}}{1-3z^{-1}} = z^{-1}X_1(z)$

where $X_1(z) = \frac{1}{1-3z^{-1}}$

Here, from the time shifting property, we have $k = 1$ and $x(n) = (3)^n u(n)$

Hence $x(n) = (3)^{n-1} u(n-1)$

Example 4.11 *Find x(n)*

if $X(z) = \frac{1+\frac{1}{2}z^{-1}}{1-\frac{1}{2}z^{-1}}$

Solution Given $X(z) = \frac{1+\frac{1}{2}z^{-1}}{1-\frac{1}{2}z^{-1}} = \frac{1}{1-\frac{1}{2}z^{-1}} + \frac{\frac{1}{2}z^{-1}}{1-\frac{1}{2}z^{-1}}$

Therefore,

$$x(n) = Z^{-1}\left[\frac{1}{1-\frac{1}{2}z^{-1}} + \frac{1}{2}\frac{z^{-1}}{1-\frac{1}{2}z^{-1}}\right]$$

$$= \left(\frac{1}{2}\right)^n u(n) + \frac{1}{2}\left(\frac{1}{2}\right)^{n-1} u(n-1) = \left(\frac{1}{2}\right)^n [u(n)+u(n-1)]$$

$$= \left(\frac{1}{2}\right)^n [u(n)-u(n-1)+2u(n-1)] = \left(\frac{1}{2}\right)^n [\delta(n)+2u(n-1)]$$

4.3.4 Time Scaling

If $x(n) \stackrel{z}{\longleftrightarrow} X(z)$, ROC: $r_1 < |z| < r_2$, the

$$a^n x(n) \stackrel{z}{\longleftrightarrow} X(a^{-1}z), \text{ ROC: } |a|r_1 < |z| < |a|r_2 \tag{4.8}$$

where a is an arbitrary constant, which can be real or complex.

Proof From the definition of the z-transform, we have

$$Z[a^n x(n)] = \sum_{n=-\infty}^{\infty} a^n x(n) z^{-n} = \sum_{n=-\infty}^{\infty} x(n)(a^{-1}z)^{-n}$$

$$= X(a^{-1}z)$$

As the ROC of $X(z)$ is $r_1 < |z| < r_2$, the ROC of $X(n)(a^{-1}z)$ is

$$r_1 < |a^{-1}z| < r_2$$

or

$$|a|r_1 < |z| < |a| < r_2$$

Example 4.12 *Using scaling property, determine the z-transform of* (a) $a^n \cos \omega_0 n$, *and* (b) $a^n \sin \omega_0 n$.

Solution

(*i*) We know that $Z[\cos\omega_0 n] = \dfrac{1-z^{-1}\cos\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}} = \dfrac{z(z-\cos\omega_0)}{z^2-2z\cos\omega_0+1}$

Using the scaling property, we obtain

$$Z[a^n \cos\omega_0 n] = \frac{1-a z^{-1}\cos\omega_0}{1-2az^{-1}\cos\omega_0+a^2 z^{-2}} = \frac{z(z-a\cos\omega_0)}{z^2-2za\cos\omega_0+a^2}$$

Similarly, we obtain

$$Z[a^n \sin\omega_0 n] = \frac{a z^{-1}\sin\omega_0}{1-2az^{-1}\cos\omega_0+a^2 z^{-2}} = \frac{z a\sin\omega_0}{z^2-2za\cos\omega_0+a^2}$$

Example 4.13 *Find the z-transform of* $x(n) = 2^n u(n-2)$.

Solution Given $x(n) = 2^n u(n-2)$

$$Z[u(n)] = \frac{1}{1-z^{-1}}$$

Hence, $$Z[u(n-2)] = \frac{z^{-2}}{1-z^{-1}}$$

Therefore, $$Z[2^n u(n-2)] = \left.\frac{z^{-2}}{1-z^{-1}}\right|_{z^{-1}\to 2z^{-1}} = \frac{(2z^{-1})^2}{1-2z^{-1}} = \frac{4z^{-2}}{1-2z^{-1}}$$

4.3.5 Differentiation

If $x(n) \overset{z}{\longleftrightarrow} X(z)$ then

$$nx(n) \overset{z}{\longleftrightarrow} -z\frac{dX(z)}{dz} \quad \text{or} \quad z^{-1}\frac{dX(z)}{dz^{-1}} \tag{4.9}$$

Proof By differentiating both sides of the definition of z-transform, we have

$$\frac{dX(z)}{dz} = \sum_{n=-\infty}^{\infty} x(n)(-n)z^{-n-1} = -z^{-1}\sum_{n=-\infty}^{\infty}[nx(n)]z^{-n}$$
$$= -z^{-1}Z\{nx(n)\}$$

Example 4.14 *Find the z-transform of* $x(n) = n^2 u(n)$.

Solution $x(n) = n^2 u(n)$

$$X(z) = Z[n^2 u(n)] = Z[n(n\,u(n))]$$
$$= z^{-1}\frac{d}{dz^{-1}}[Z(nu(n))]$$
$$= z^{-1}\frac{\mathrm{d}}{\mathrm{d}z^{-1}}\frac{z^{-1}}{(1-z^{-1})^2}$$
$$= z^{-1}\left[\frac{(1-z^{-1})^2 - z^{-1}[2(1-z^{-1})(-1)]}{(1-z^{-1})^4}\right]$$
$$= z^{-1}\frac{(1-z^{-1})(1-z^{-1}+2z^{-1})}{(1-z^{-1})^4} = z^{-1}\frac{1+z^{-1}}{(1-z^{-1})^3}$$

4.3.6 Convolution

It is expressed as $x(n) = x_1(n) * x_2(n) = \sum_{k=0}^{n} x_1(n-k)x_2(k)$

If $x_1(n) \xleftrightarrow{z} X_1(z)$ and $x_2(n) \xleftrightarrow{z} X_2(z)$,

then, $x(n) = x_1(n) * x_2(n) \xleftrightarrow{z} X(z) = X_1(z)X_2(z)$ (4.10)

Proof The convolution of $x_1(n)$ and $x_2(n)$ is defined as

$$x(n) = x_1(n) * x_2(n) = \sum_{k=-\infty}^{\infty} x_1(k)x_2(n-k)$$

The z-transform of $x(n)$ is

$$x(n) = \sum_{k=-\infty}^{\infty} x(n)z^{-n} = \sum_{n=-\infty}^{\infty}\left[\sum_{k=-\infty}^{\infty} x_1(k)x_2(n-k)\right]z^{-n}$$

Upon interchanging the order of the summations and applying the time-shifting property, we obtain

$$X(z) = \sum_{k=-\infty}^{\infty} x_1(k)\left[\sum_{n=-\infty}^{\infty} x_2(n-k)z^{-n}\right]$$
$$= X_2(z)\sum_{k=-\infty}^{\infty} x_1(k)z^{-k} = X_2(z)X_1(z)$$

Example 4.15 *Compute the convolution x(n) of the signals*

$$x_1(n) = \{4, \ -2, 1\}$$
$$x_2(n) = \begin{cases} 1, & 0 \le n \le 5 \\ 0, & \text{otherwise} \end{cases}$$

Solution The z-transforms for the given signals are written as

$$X_1(z) = 4 - 2z^{-1} + z^{-2}$$
$$X_2(z) = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5}$$

Therefore,

$X(z) = X_1(z)\, X_2(z) = 4 + 2z^{-1} + 3z^{-2} + 3z^{-3} + 3z^{-4} + 3z^{-5} - z^{-6} + z^{-7}$

Taking inverse z-transform, we obtain

$$x(n) = \left\{ \underset{\uparrow}{4}, 2, 3, 3, 3, 3, -1, 1 \right\}$$

Example 4.16 *Determine the convolution of the two sequences $x(n) = \{2, 1, 0, 0.5\}$ and $h(n) = \{2, 2, 1, 1\}$*

Solution Taking z-transform of the given two sequences $x(n)$ and $h(n)$, we get

$X(z) = 2 + z^{-1} + 0.5z^{-3}$

$H(z) = 2 + 2z^{-1} + z^{-2} + z^{-3}$

$Y(z) = X(z)\, H(z) = (2 + z^{-1} + 0.5z^{-3})(2 + 2z^{-1} + z^{-2} + z^{-3})$

$\quad = 4 + 6z^{-1} + 4z^{-2} + 4z^{-3} + 2z^{-4} + 0.5z^{-5} + 0.5z^{-6}$

Taking inverse z-transform, we get

$$y(n) = \left\{ \underset{\uparrow}{4}, 6, 4, 4, 2, 0.5, 0.5 \right\}$$

Alternate method

Let $y(n) = x(n) * h(n)$

The convolution can be obtained easily using the convolution table.

$y(n) \downarrow$ \ $x(n) \rightarrow$	2	1	0	0.5
2	4	2	0	1
2	4	2	0	1
1	2	1	0	0.5
1	2	1	0	0.5

Thus $y(n)$ can be obtained by adding elements of sequence along the slant lines.

$$y(n) = \left\{ \underset{\uparrow}{4}, 6, 4, 4, 2, 0.5, 0.5 \right\}$$

The convoluted sequence will have in total $(n + m - 1)$ terms where n and m are the number of terms in the sequences to be convolved. However, this table can be applied for convolving two finite sequence signals only.

Example 4.17 *Find x(n) by using convolution for*

$$X(z) = \frac{1}{\left(1 - \frac{1}{2}z^{-1}\right)\left(1 + \frac{1}{4}z^{-1}\right)}$$

Solution Given

$$X(z) = \frac{1}{\left(1 - \frac{1}{2}z^{-1}\right)\left(1 + \frac{1}{4}z^{-1}\right)}$$
$$= X_1(z).X_2(z)$$

where $X_1(z) = \frac{1}{\left(1 - \frac{1}{2}z^{-1}\right)}$ and $X_2 = \frac{1}{\left(1 + \frac{1}{4}z^{-1}\right)}$

Taking inverse z-transform, we get

$$x_1(n) = z^{-1}\left[\frac{1}{\left(1 - \frac{1}{2}z^{-1}\right)}\right] = \left(\frac{1}{2}\right)^n u(n)$$

$$x_2(n) = z^{-1}\left[\frac{1}{\left(1 + \frac{1}{4}z^{-1}\right)}\right] = \left(-\frac{1}{4}\right)^n u(n)$$

Therefore, $x(n) = x_1(n) * x_2(n) = \sum_{k=0}^{n} x_1(n-k)x_2(k)$

$$= \sum_{k=0}^{n}\left(\frac{1}{2}\right)^{n-k}\left(\frac{-1}{4}\right)^k = \left(\frac{1}{2}\right)^n \sum_{k=0}^{n}\left[\frac{(-1/4)}{(1/2)}\right]^k$$

$$= \left(\frac{1}{2}\right)^n \sum_{k=0}^{n}\left(-\frac{1}{2}\right)^k = \left(\frac{1}{2}\right)^n \frac{1-\left(-\frac{1}{2}\right)^{n+1}}{1-\left(-\frac{1}{2}\right)}$$

$$= \left(\frac{1}{2}\right)^n \cdot \frac{2}{3}\left[1-\left(-\frac{1}{2}\right)\left(-\frac{1}{2}\right)^n\right] = \left[\frac{2}{3}\left(\frac{1}{2}\right)^n + \frac{1}{3}\left(\frac{-1}{4}\right)^n\right]u(n)$$

4.3.7 Correlation

If $x_1(n) \overset{z}{\longleftrightarrow} X_1(z)$ and $x_2(n) \overset{z}{\longleftrightarrow} X_2(z)$, then,

$$r_{x_1x_2}(l) = \sum_{n=-\infty}^{\infty} x_1(n)x_2(n-1) \overset{z}{\longleftrightarrow} R_{x_1x_2}(z) = X_1(z)X_2(z^{-1}) \quad (4.11)$$

Proof We know that

$$r_{x_1x_2}(l) = x_1(l)*x_2(-l)$$

Using the convolution and time-reversal properties, we get

$$R_{x_1x_2}(z) = Z\{x_1(l)\}\, Z\{x_2(-l)\} = X_1(z)\, X_2(z^{-1})$$

The ROC of $R_{x_1x_2}(z)$ is at least the intersection of $X_1(z)$ and $X_2(z-1)$.

As in the case of convolution, the cross-correlation of two signals can be readily done via polynomial multiplication and then inverse transforming the result. The convolution is equivalent to the cross-correlation of the two waveforms in which one of the original sequences has been time reversed, with the normalising factor $\frac{1}{N}$ set to unity. Hence, both convolution and correlation can be computed by the same computer program by reversing any one of the sequences.

Example 4.18 *Determine the cross-correlation sequence of* $r_{x_1x_2}(l)$ *the sequences:*

$$x_1(n) = \{1, 2, 3, 4\}$$
$$x_2(n) = \{4, 3, 2, 1\}$$

Solution Cross-correlation sequence can be obtained using the correlation property of z-transform, given in Eq. (4.11). Hence, for the given $x_1(n)$ and $x_2(n)$,

$$X_1(z) = 1 + 2z^{-1} + 3z^{-2} + 4z^{-3}$$
$$X_2(z) = 4 + 3z^{-1} + 2z^{-2} + z^{-3}$$

Therefore, $X_2(z^{-1}) = 4 + 3z + 2z^2 + z^3$

$$R_{x_1x_2}(z) = X_1(z)\, X_2(z^{-1}) = (1 + 2z^{-1} + 3z^{-2} + 4z^{-3})\,(4 + 3z + 2z^2 + z^3)$$
$$= (z^3 + 4z^2 + 10z + 20 + 25z^{-1} + 24z^{-2} + 16z^{-3})$$

Therefore, $r_{x_1x_2}(l) = Z^{-1}[R_{x_1x_2}(z)$

$$= \left\{\begin{matrix} 1, 4, 10, 20, 25, 24, 16 \\ \uparrow \end{matrix}\right\}$$

Alternative Method

From Eq. (4.11), the cross-correlation can be written as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l),\ l = 0, \pm 1,\ \pm 2..$$

or equivalently, as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n+l)y(n),\ l = 0, \pm 1,\ \pm 2..$$

where l is the time shift index.

For $l = 0$, we have $r_{xy}(0) = \sum_{n=-\infty}^{\infty} x(n)y(n)$

The product sequence $P_o(n) = x(n)\, y(n)$ is

$$P_0(n) = \left\{\begin{matrix} 4, 6, 6, 4 \\ \uparrow \end{matrix}\right\}$$

and hence, the sum over all values of n is

$$r_{xy}(0) = 20$$

For $l > 0$, $y(n)$ is shifted to the right relative to $x(n)$ by l units and the product sequence $P_i(n) = x(n)\, y(n - l)$ can be computed. Then all values of the product sequence may be added and we get values of the cross-correlation sequence.

$$r_{xy}(1) = 25, \qquad r_{xy}(2) = 24, \qquad r_{xy}(3) = 16$$

For $l < 0$,

$$r_{xy}(-1) = 10, \quad r_{xy}(-2) = 4, \quad r_{xy}(-3) = 1$$

Hence, the cross-correlation sequence of $x(n)$ and $y(n)$ is

$$r_{xy}(l) = \left\{ \begin{matrix} 1, 4, 10, 20, 25, 24, 16 \\ \uparrow \end{matrix} \right\}$$

4.3.8 Initial Value Theorem

If $x(n)$ is a causal sequence with z-transform $X(z)$, the initial value can be determined by using the expression

$$x(0) = \lim_{n \to 0} x(n) = \lim_{|z| \to \infty} X(z) \tag{4.12}$$

Proof Since $x(n)$ is causal, the definition of z-transform becomes

$$X(z) = \sum_{n=0}^{\infty} x(n) z^{-n} = x(0) + x(1)z^{-1} + x(2)z^{-2} + \ldots$$

Here, as $z \to \infty$, $z^{-n} = 0$, i.e. all terms except the first approach zero

Therefore, $\lim_{|z| \to \infty} X(z) = x(0)$

4.3.9 Final Value Theorem

If $X(z) = Z[x(n)]$ and the poles of $X(z)$ are all inside the unit circle, then the final value of the sequence, $x(\infty)$, can be determined by using the expression

$$\lim_{n \to \infty} x(\infty) = \lim_{z \to 1} x(1 - z^{-1}) X(z), \text{ if } x(\infty) \text{ exists} \tag{4.13}$$

Proof The z-transform of $\{x(n) - x(n - 1)\}$ is

$$Z\{x(n) - x(n-1)\} = X(z) - z^{-1}X(z) = \sum_{n=0}^{\infty} [x(n) - x(n-1)] z^{-n}$$

Using $Z[x(n-k)] = z^{-k} X(z)$, we get

$$(1 - z^{-1}) X(z) = \lim_{N \to \infty} \sum_{n=0}^{N} [x(n) - x(n-1)] z^{-n}$$

Taking the limit as $z \to 1$, we get

$$\lim_{z \to 1} (1 - z^{-1}) X(z) = \lim_{z \to 1} \lim_{N \to \infty} \sum_{n=0}^{N} [x(n) - x(n-1)] z^{-n}$$

Interchanging the summations on the right, we get

$$= \lim_{N \to \infty} \lim_{z \to 1} \sum_{n=0}^{N} [x(n) - x(n-1)] z^{-n}$$

$$= \lim_{N \to \infty} \sum_{n=0}^{N} [x(n) - x(n-1)]$$

$$= \lim_{N \to \infty} [x(0) - x(-1) + x(1) - x(0) + x(2) - x(1) + \ldots]$$

$$= \lim_{N \to \infty} x(N)$$

Since $x(-1) = 0$.The limit $z \to 1$ gives correct results only when the point $z = 1$ is located within the ROC of $X(z)$.

Example 4.19 *If $X(z) = 2 + 3z^{-1} + 4z^{-2}$, find the initial and final values of the corresponding sequence, $x(n)$.*

Solution

$$x(0) = \lim_{|z|\to\infty} [2+3z^{-1}+4z^{-2}] = 2+\frac{3}{\infty}+\frac{4}{\infty} = 2$$

$$x(\infty) = \lim_{|z|\to 1}[(1-z^{-1})(2+3z^{-1}+4z^{-2})]$$

$$= \lim_{|z|\to 1}[2+z^{-1}+z^{-2}-4z^{-3}] = 2+1+1-4 = 0$$

Also, by inspection, the above results are confirmed that the initial value is two as it is the coefficient of z^0 and the final value is 0 as the sequence is a finite one.

Example 4.20 *Determine the causal sequence $x(n)$ for $X(z)$ given by*

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

Solution Multiplying numerator and denominator by z^2 in the given equation, we obtain

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

From Table 4.3, we have

$$Z[a^n \cos\omega_0 n] = \frac{z(z-a\cos\omega_0)}{z^2-2za\cos\omega_0+a^2}$$

$$Z[a^n \sin\omega_0 n] = \frac{za\sin\omega_0}{z^2-2za\cos\omega_0+a^2}$$

The denominator of $X(z)$ gives

$$z^2 - 2z + 4 = z^2 - 2z\,a\cos\omega_0 + a^2$$

Comparing both sides, we get $a^2 = 4$ and hence $a = 2$,

$$a\cos\omega_0 = 1, \text{ i.e. } \cos\omega_0 = \frac{1}{2}, \text{ hence } \omega_0 = \cos^{-1}\left(\frac{1}{2}\right) = 60° = \frac{\pi}{3}$$

$$\sin\omega_0 = \sin\frac{\pi}{3} = \frac{\sqrt{3}}{2}$$

The numerator of $X(z)$ gives

$$z^2 + 2z = z^2 - z\,a\cos\omega_0 + A(za\sin\omega_0) = z^2 - z(2)\left(\frac{1}{2}\right) + A(za\sin\omega_0)$$

Therefore, $3z = A(za\sin\omega_0)$

Hence, $Aa\sin\omega_0 = 3$

$$A\,2\sin\frac{\pi}{3} = A2\frac{\sqrt{3}}{2} = 3. \text{ So, } A = \sqrt{3}$$

Therefore, $$X(z) = \frac{\left[z^2-2z\cos\frac{\pi}{3}\right]+\sqrt{3}\left[2z\sin\frac{\pi}{3}\right]}{z^2-2z\left[2\cos\frac{\pi}{3}\right]+4}$$

Hence, $x(n) = 2^n \left[\cos\frac{n\pi}{3} + \sqrt{3}\sin\frac{n\pi}{3} \right] u(n)$

4.3.10 Multiplication of Two Sequences

If $x_1(n) \xleftrightarrow{z} X_1(z)$

$x_2(n) \xleftrightarrow{z} X_1(z)$

then, $x(n) = x_1(n)x_2(n) \xleftrightarrow{z} X(z) = \dfrac{1}{2\pi j}\oint_C X_1(v)X_2\left(\dfrac{z}{\upsilon}\right)v^{-1}\mathrm{d}v$

where C is a closed contour that encloses the origin and lies within the ROC common to both $X_1(v)$ and $X_2(1/v)$.

4.3.11 Multiplication by *n*

$$Z[nx(n)] = -z\frac{\mathrm{d}}{\mathrm{d}z}X(z)$$

Proof From the definition of z-transform, we have

$$Z\{nx(n)\} = \sum_{n=0}^{\infty} nx(n)z^{-n} = z\sum_{n=0}^{\infty} x(n)(nz^{-n-1})$$

$$= z\sum_{n=0}^{\infty} x(n)\left[-\frac{\mathrm{d}}{\mathrm{d}z}z^{-n}\right] = -z\frac{\mathrm{d}}{\mathrm{d}z}\sum_{n=0}^{\infty} x(n)z^{-n} = -z\frac{\mathrm{d}}{\mathrm{d}z}X(z)$$

4.3.12 Division by *n* + *a* (*a* is any Real Number)

$$Z\left\{\frac{x(n)}{n+a}\right\} = -z^a\int_0^z \frac{X(\tilde{z})}{\tilde{z}^{a+1}}\, d\tilde{z}$$

Proof In a direct manner, where $\tilde{z}$ is a dummy integration variable.

$$Z\left\{\frac{x(n)}{n+a}\right\} = \sum_{n=0}^{\infty}\frac{x(n)}{n+a}z^{-n} = \sum_{n=0}^{\infty} x(n)z^{-a}\left[-\int_0^z \tilde{z}^{-n-a-1}\mathrm{d}\,\tilde{z}\right]$$

This is written as

$$= -z^a\int_0^z \frac{1}{\tilde{z}^{a+1}}\sum_{n=0}^{\infty} x(n)\tilde{z}^{-n}\mathrm{d}\tilde{z} = -\int_0^z \frac{X(\tilde{z})}{\tilde{z}^{a+1}}\mathrm{d}\tilde{z}$$

4.3.13 Time Delay (for One-Sided *z*-Transform)

If $x(n) \xleftrightarrow{z} X(z)$, then

$$x(n-k) \xleftrightarrow{z} z^{-k}\left[X(z) + \sum_{n=1}^{k} x(-n)z^{n}\right],\ k > 0$$

4.3.14 Time Advance

If $x(n) \xleftrightarrow{z} X(z)$, then, $x(n+k) \xleftrightarrow{z} z^k\left[X(z) - \sum_{n=0}^{k-1} x(n)z^{-n}\right]$

EVALUATION OF THE INVERSE z-TRANSFORM 4.4

The inverse z-transform was defined in Sec. 4.2.1. The three basic methods of performing the inverse z-transform, viz. (i) long division method, (ii) partial fraction expansion method, and (iii) residue method are discussed in this section.

4.4.1 Long Division Method

The z-transform of a signal or system which is expressed as the ratio of two polynomials in z, is simply divided out to produce a power series in the form of an equation.

$$X(z) = \sum_{n=0}^{\infty} x(n) z^{-n}$$

with the coefficients representing the sequence values in the time domain, namely

$$X(z) = \frac{N(z)}{D(z)} = \sum_{n=0}^{\infty} a_n z^{-n} = az^0 + a_1 z^{-1} + a_2 z^{-2} + \ldots$$

where the coefficients a_n are the values of $x(n)$.

From the above equation it is clear that expansion does not result in a closed form solution. Hence, if $X(z)$ can be expanded in a power series, the coefficients represent the inverse sequence values. Thus, the coefficient of z^{-K} is the K^{th} term in the sequence. The region of convergence will determine whether the series has positive or negative exponents. For right hand sequences, called causal sequences will have primarily negative exponents, while left hand sequences the anti-causal sequences will have positive exponents. For annular regions of convergence, a Laurent expansion will give both the positive and negative exponents.

This method is only useful for having a quick look at the first few samples of the corresponding signals.

Example 4.21 *A system has an impulse response $h(n) = \{1, 2, 3\}$ and output response $y(n) = \{1, 1, 2, -1, 3\}$. Determine the input sequence $x(n)$.*

Solution Performing the z-transform of $h(n)$ and $y(n)$, we have

$$H(z) = Z[h(n)] = Z[1, 2, 3] = 1 + 2z^{-1} + 3z^{-2}$$

$$Y(z) = Z[y(n)] = Z[1, 1, 2, -1, 3] = 1 + z^{-1} + 2z^{-2} - z^{-3} + 3z^{-4}$$

We know that $H(z) = \dfrac{Y(z)}{X(z)}$

Therefore, $X(z) = \dfrac{Y(z)}{H(z)} = \dfrac{1 + z^{-1} + 2z^{-2} - z^{-3} + 3z^{-4}}{1 + 2z^{-1} + 3z^{-2}}$

$$\begin{array}{r|l}
 & 1 - z^{-1} + z^{-2} \\
\hline
1 + 2z^{-1} + 3z^{-2} & 1 + z^{-1} + 2z^{-2} - z^{-3} + 3z^{-4} \\
 & 1 + 2z^{-1} + 3z^{-2} \\
\hline
 & -z^{-1} - z^{-2} - z^{-3} \\
 & -z^{-1} - 2z^{-2} - 3z^{-3} \\
\hline
 & z^{-2} + 2z^{-3} + 3z^{-4} \\
 & z^{-2} + 2z^{-3} + 3z^{-4} \\
\hline
 & 0
\end{array}$$

Therefore, $X(z) = 1 - z^{-1} + z^{-2}$

Taking inverse z-transform, we get

$$x(n) = \left\{ \begin{matrix} 1, -1, 1 \\ \uparrow \end{matrix} \right\}$$

Example 4.22 *Using long division, determine the inverse z-transform of*

$$X(z) = \frac{1}{1 - (3/2)z^{-1} + (1/2)z^{-2}}$$

when (a) ROC: $|z| > 1$ *and (b) ROC:* $|z| < \dfrac{1}{2}$

Solution

(a) Since the ROC: $|z| > 1$ is the exterior of a circle, $x(n)$ is a causal signal. Thus we seek a power series expansion in negative powers of z. By dividing the numerator of $X(z)$ by its denominator, we obtain

$$\begin{array}{r|l}
 & 1 + \frac{3}{2}z^{-1} + \frac{7}{4}z^{-2} + \frac{15}{8}z^{-3} + \frac{31}{16}z^{-4} + \ldots \\
\hline
1 + \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2} & 1 \\
 & 1 + \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2} \\
\hline
 & \frac{3}{2}z^{-1} - \frac{1}{2}z^{-2} \\
 & \frac{3}{2}z^{-1} - \frac{9}{4}z^{-2} + \frac{3}{4}z^{-3} \\
\hline
 & \frac{7}{4}z^{-2} - \frac{3}{4}z^{-3} \\
 & \frac{7}{4}z^{-2} - \frac{21}{8}z^{-3} + \frac{7}{8}z^{-4} \\
\hline
 & \frac{15}{8}z^{-3} - \frac{7}{8}z^{-4} \\
 & \frac{15}{8}z^{-3} - \frac{45}{16}z^{-4} + \frac{15}{16}z^{-5} \\
\hline
 & \frac{31}{16}z^{-4} - \frac{15}{16}z^{-5}
\end{array}$$

Therefore, $X(z) = 1 + \dfrac{3}{2}z^{-1} + \dfrac{7}{4}z^{-2} + \dfrac{15}{8}z^{-3} + \dfrac{31}{16}z^{-4} + \ldots$

Taking inverse z-transform, we obtain $x(n) = \left\{ \begin{matrix} 1, \frac{3}{2}, \frac{7}{4}, \frac{15}{8}, \frac{31}{16}, \ldots \\ \uparrow \phantom{, \frac{3}{2}, \frac{7}{4}, \frac{15}{8}, \frac{31}{16}, \ldots} \end{matrix} \right\}$

(b) In this case the ROC: $|z| < 0.5$ is the interior of a circle. Consequently, the signal $x(n)$ is anti-causal. To obtain a power series expansion in positive powers of z, we perform long division in the following way.

$$
\begin{array}{r|l}
 & 2z^2 + 6z^3 + 14z^4 + 30z^5 + 62z^6 + \ldots \\
\hline
\frac{1}{2}z^{-2} - \frac{3}{2}z^{-1} + 1 & 1 \\
 & 1 - 3z + 2z^2 \\
\hline
 & 3z - 2z^2 \\
 & 3z - 9z^2 + 6z^3 \\
\hline
 & \quad 7z^2 - 6z^3 \\
 & \quad 7z^2 - 21z^3 + 14z^4 \\
\hline
 & \qquad 15z^3 - 14z^4 \\
 & \qquad 15z^3 - 45z^4 + 30z^5 \\
\hline
 & \qquad\quad 31z^4 - 30z^5
\end{array}
$$

Therefore, $X(z) = 2z^2 + 6z^3 + 14z^4 + 30z^5 + 62z^6 + \cdots$

Taking inverse z-transform, we get $x(n) = \left\{ \ldots, 62, 30, 14, 4, 6, 2, 0, \underset{\uparrow}{0} \right\}$

4.4.2 Partial Fraction Expansion

The partial fraction expansion method is useful to decompose a signal or a system transfer function into a sum of standard functions. First we have to factorise the denominator of the transfer function $H(z)$ into prime factors. Then, for simple poles, $H(z)$ can be expressed in the following form, with $m \le n$,

$$
\begin{aligned}
H(z) &= \frac{a_0 z^m + a_1 z^{m-1} + a_2 z^{m-2} + \ldots + a_m}{(z - p_1)(z - p_2)\ldots(z - p_n)} \\
&= A_0 + \frac{A_1}{z - p_1} + \frac{A_2}{z - p_2} + \ldots + \frac{A_n}{z - p_n}
\end{aligned}
$$

where $A_0 = \lim_{z \to \infty} H(z) = \begin{cases} a_0, & \text{if } m = n \\ 0, & \text{if } m < n \end{cases}$

$$A_i = (z - p_i) H(z)\Big|_{z = p_i} \quad \text{for } i = 1, 2, \ldots n.$$

With repeated linear roots, we have the terms

$$\frac{A_{i1}}{z - p_i} + \frac{A_{i2}}{(z - p_i)^2} + \ldots + \frac{A_{ir}}{(z - p_i)^r}$$

where $A_{ir} = (z - p_i)^r X(z)\Big|_{z = p_i}$

$$A_{i(r-1)} = \frac{\mathrm{d}}{\mathrm{d}z}(z - p_i)^r X(z)\Big|_{z = p_i}$$

$$\vdots \qquad\qquad \vdots$$

$$A_{i(r-k)} = \frac{1}{K!}\frac{\mathrm{d}^k}{\mathrm{d}z^k}(z - p_i)^r X(z)\Big|_{z = p_i}$$

$$\vdots \qquad\qquad \vdots$$

$$A_{i1} = \frac{1}{(r-1)!}\frac{\mathrm{d}^{r-1}}{\mathrm{d}r^{r-1}}(z - p_i)^r X(z)\Big|_{z = p_i}$$

For a quadratic factor $(z^2 + a_1 z + a_2)$, this gives

$$\frac{A_i z + A_2}{z^2 + a_1 z + a_2}$$

Each constant is determined by substituting the appropriate value of z to make the associated prime factor zero. Then the inverse z-transform may be obtained using Table 4.3.

Similarly, the system transfer function $H(z)$ expressed as a ratio of two polynomials in z may be decomposed into a sum of partial fractions. Then the corresponding z-transform can be obtained from Table 4.3.

Example 4.23 *By using partial fraction expansion method, find the inverse z-transform of*

$$H(z) = \frac{-4 + 8z^{-1}}{1 + 6z^{-1} + 8z^{-2}}$$

Solution

$$H(z) = \frac{-4 + 8z^{-1}}{1 + 6z^{-1} + 8z^{-2}} = \frac{-4 + 8z^{-1}}{(1 + 4z^{-1})(1 + 2z^{-1})}$$

$$H(z) = \frac{A_1}{1 + 4z^{-1}} + \frac{A_2}{1 + 2z^{-1}}$$

$$A_1 = \left.\frac{-4 + 8z^{-1}}{1 + 2z^{-1}}\right|_{\text{at } z^{-1} = -1/4} = \frac{-6}{1/2} = -12$$

$$A_2 = \left.\frac{-4 + 8z^{-1}}{1 + 4z^{-1}}\right|_{\text{at } z^{-1} = -1/2} = \frac{-8}{-1} = 8$$

Therefore, $H(z) = \dfrac{-12}{1 + 4z^{-1}} + \dfrac{8}{1 + 2z^{-1}}$

Taking inverse z-transform, we get

$$h(n) = [-12(-4)^n + 8(-2)^n]\, u(n)$$

Example 4.24 *Determine the causal signal x(n) having the z-transform*

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

Solution Expanding the given $X(z)$ in terms of the positive powers of z.

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

Hence, $F(z) = \dfrac{X(z)}{z} = \dfrac{z^2}{(z + 1)(z - 1)^2} = \dfrac{A_1}{(z + 1)} + \dfrac{A_2}{(z - 1)} + \dfrac{A_3}{(z - 1)^2}$

Here, $A_1 = (z + 1)F(z)\big|_{z=-1} = \left.\dfrac{z^2}{(z - 1)^2}\right|_{z=-1} = \dfrac{1}{4}$

$$A_3 = (z-1)^2 F(z)\Big|_{z=-1} = \frac{z^2}{(z+1)}\Bigg|_{z=1} = \frac{1}{2}$$

$$A_2 = \frac{d}{dz}\left[\frac{z^2}{(z+1)}\right]_{z=1} = \frac{(z+1)2z - z^2}{(z+1)^2}\Bigg|_{z=1} = \frac{3}{4}$$

Therefore, $F(z) = \frac{1}{4}\frac{1}{(z+1)} + \frac{3}{4}\frac{1}{(z-1)} + \frac{1}{2}\frac{1}{(z-1)^2}$

Therefore, $X(z) = \frac{1}{4}\frac{z}{(z+1)} + \frac{3}{4}\frac{z}{(z-1)} + \frac{1}{2}\frac{z}{(z-1)^2}$

Taking inverse z-transform of $X(z)$, we obtain

$$x(n) = \frac{1}{4}(-1)^n u(n) + \frac{3}{4}u(n) + \frac{1}{2}nu(n)$$

$$= \left[\frac{1}{4}(-1)^n + \frac{3}{4} + \frac{1}{2}n\right]u(n)$$

Alternate Method

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

$$= \frac{A_1}{1+z^{-1}} + \frac{A_2}{1-z^{-1}} + \frac{A_3}{(1-z^{-1})^2}$$

Equating the numerators, we get

$$1 = A_1(1-z^{-1})^2 + A_2(1+z^{-1})(1-z^{-1}) + A_3(1+z^{-1})$$

$$= A_1(1-2z^{-1}+z^{-2}) + A_2(1-z^{-2}) + A_3(1+z^{-1})$$

Here, $A_1 + A_2 + A_3 = 1$

$A_1 - A_2 = 0$, i.e. $A_1 = A_2$

$-2A_1 + A_3 = 0$

Solving, we get $A_1 = \frac{1}{4}, A_2 = \frac{1}{4}$ and $A_3 = \frac{1}{2}$

Therefore, $X(z) = \frac{1/4}{(1+z^{-1})} + \frac{1/4}{(1-z^{-1})} + \frac{1/2}{(1-z^{-1})^2}$

Here the last term $\frac{1}{2}\left[\frac{1}{(1-z^{-1})^2}\right]$ can be expanded as

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

Therefore, $X(z) = \frac{1/4}{(1+z^{-1})} + \frac{3/4}{(1-z^{-1})} + \frac{1}{2}\frac{z^{-1}}{(1-z^{-1})^2}$

Taking inverse z-transform, we get

$$x(n) = \left[\frac{1}{4}(-1)^n + \frac{3}{4} + \frac{1}{2}n\right]u(n)$$

Example 4.25 *Given that* $H(z) = \dfrac{z+1}{z^2 - 0.9z + 0.81}$ *is a causal system, find its*
(a) *transfer function representation,* (b) *difference equation representation,*
(c) *impulse response representation*

Solution The poles of the system function are at $z = 0.9\left(\frac{1}{2} \pm j\frac{\sqrt{3}}{2}\right) = 0.9 < \pm\pi/3$. Hence the ROC of the above causal system is $|z| > 0.9$. Therefore, the unit circle is in the ROC and the discrete-time Fourier transform $H(e^{j\omega})$ exists.

(a) Substituting in $z = e^{j\omega}$ in $H(z)$, we get

$$H(e^{j\omega}) = \frac{e^{j\omega}+1}{e^{j2\omega} - 0.9e^{j\omega} + 0.81} = \frac{e^{j\omega}+1}{(e^{j\omega} - 0.9e^{j\omega\pi/2})(e^{j\omega} - 0.9e^{-j\omega\pi/3})}$$

(b) Multiplying the denominator and numerator of $H(z)$ by z^{-2}, we get

$$H(z) = \frac{Y(z)}{X(z)} = \frac{z^{-1} + z^{-2}}{1 - 0.9z^{-1} + 0.81z^{-2}}$$

Cross-multiplying, we get

$$Y(z) - 0.9z^{-1}Y(z) + 0.81z^{-2}Y(z) = z^{-1}X(z) + z^{-2}X(z)$$

Taking inverse z-transform, we get the difference equation of the system as

$$y(n) - 0.9y(n-1) + 0.81y(n-2) = x(n-1) + x(n-2)$$

or

$$y(n) = 0.9y(n-1) - 0.81y(n-2) + x(n-1) + x(n-2)$$

(c) We desire the partial fraction expansion of $H(z)/z$, which is

$$\frac{H(z)}{z} = \frac{(z+1)}{z(z^2 - 0.9z + 0.81)}$$

$$= \frac{(z+1)}{z(z - 0.9e^{j\pi/3})(z - 0.9e^{-j\pi/3})} = \frac{A_0}{z} + \frac{A_1}{(z - 0.9e^{j\pi/3})} + \frac{A_2}{(z - 0.9e^{-j\pi/3})}$$

where $A_0 = 1.2346$
$A_1 = -0.6173 + j0.9979$
$A_2 = -0.6173 - j0.9979$

Therefore,

$$H(z) = 1.2346 + \frac{(-0.6173 + j0.9979)z}{(z - 0.9e^{j\pi/3})} + \frac{(-0.6173 - j0.9979)z}{(z - 0.9e^{-j\pi/3})}$$

$$= 1.2346 + \frac{-0.6173 + j0.9979}{(1 - |0.9|e^{j\pi/3}z^{-1})} + \frac{-0.6173 - j0.9979}{1 - |0.9|e^{-j\pi/3}z^{-1}}, \ |z| > 0.9$$

Taking inverse z-transform, we get

$$h(n) = 1.2346\delta(n) + \begin{bmatrix} (-0.6173 + j0.9979)\,|\,0.9\,|^n\ e^{-j\pi n/3} \\ +(-0.6173 - j0.9979)\,|\,0.9\,|^n\ e^{-j\pi n/3} \end{bmatrix} u(n)$$

$$= 1.2346\delta(n) + |\,0.9\,|^n\ [-1.2346\cos(\pi n/3) + j1.99585\sin(\pi n/3)]u(n)$$

Example 4.26 *Determine the inverse z-transform of the system function*

$$H(z) = \frac{1}{(1-0.2z^{-1})z^{-2}}$$

Solution This system function is written as

$$H(z) = \frac{z^3}{z-0.2} = A_1 z^2 + A_2 z + \frac{A_3}{z-0.2}$$

$$= \frac{A_1 z^3 - 0.2A_1 z^2 + A_2 z^2 - 0.2A_2 z + A_3}{z-0.2}$$

Equating the terms having the same powers of z, we get $A_1 = 1$, $A_2 = 0.2$ and $A_3 = 0.04$.

Therefore, $H(z) = z^2 + 0.2z + 0.04\dfrac{z}{z-0.2}$

Taking inverse transform, we get

$$h(n) = \delta(n+2) + 0.2\delta(n+1) + 0.04\,(0.2)^n$$

$$= \delta(n+2) + 0.2\delta(n+1) + (0.2)^{n+2}$$

where the last term is applicable for $n \geq 0$. Therefore, this equation is equivalent to

$$h(n) = \begin{cases} 0.2^{n+2}, & n \geq -2 \\ 0, & n < -2 \end{cases}$$

Example 4.27 *Determine the inverse z-transform of the system function*

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

Solution Expanding $H(z)$ in the partial fraction form, we get

$$\frac{H(z)}{z} = \frac{z^2 - 3z + 8}{z(z-2)(z+2)(z+3)} = \frac{A_0}{z} + \frac{A_1}{z-2} + \frac{A_2}{z+2} + \frac{A_3}{z+3}$$

where

$$A_0 = H(z)\big|_{z=0} = -\frac{2}{3}$$

$$A_1 = \left.\frac{(z-2)H(z)}{z}\right|_{z=2} = \left.\frac{z^2 - 3z + 8}{z(z+2)(z+3)}\right|_{z=2} = \frac{3}{20}$$

$$A_2 = \left.\frac{(z+2)H(z)}{z}\right|_{z=-2} = \left.\frac{z^2 - 3z + 8}{z(z-2)(z+3)}\right|_{z=-2} = \frac{9}{4}$$

$$A_3 = \left.\frac{(z+3)H(z)}{z}\right|_{z=-3} = \left.\frac{z^2 - 3z + 8}{z(z-2)(z+2)}\right|_{z=-3} = -\frac{26}{15}$$

Therefore, $$H(z) = -\frac{2}{3}z + \frac{3}{20}\frac{z}{z-2} + \frac{9}{4}\frac{z}{z+2} - \frac{26}{15}\frac{z}{z+3}$$

Taking inverse z-transform, we get

$$h(n) = -\frac{2}{3}\delta(n) + \frac{3}{20}2^n + \frac{9}{4}(-2)^n - \frac{26}{15}(-3)^n$$

Example 4.28 *Determine y(n) for the discrete time equation given by*

$$y(n) + 2y(n-1) = \frac{7}{2}u(n)$$

with $y(-1) = 0$, *and* $u(n) = \begin{cases} 1, & n = 0, 1, \ldots \\ 0, & n = -1, -2, \ldots \end{cases}$

Solution Taking the z-transform of both sides of the given difference equation, we get

$$Y(z) + 2z^{-1}Y(z) = \frac{7}{2}\frac{1}{(1-z^{-1})}$$

Therefore, $$Y(z) = \frac{7}{2}\cdot\frac{1}{(1-z^{-1})(1+2z^{-1})}$$

We desire the partial fraction expansion of $\frac{Y(z)}{z}$, which is

$$\frac{Y(z)}{z} = \frac{7}{2}\cdot\frac{z}{(z+1)(z+2)} = \frac{A_1}{(z-1)} + \frac{A_2}{(z+2)}$$

Solving, we get

$$A_1 = \frac{7}{6} \text{ and } A_2 = \frac{7}{3}$$

Therefore, $$Y(z) = \frac{7}{6}\cdot\frac{z}{(z-1)} + \frac{7}{3}\cdot\frac{z}{(z+2)}$$

Taking inverse z-transform, we get

$$y(n) = \frac{7}{6}u(n) + \frac{7}{3}(-2)^n$$

Example 4.29 *Determine the inverse z-transform of the following X(z) by the partial fraction expansion method*

$$X(z) = \frac{z+2}{2z^2 - 7z + 3}$$

if the ROCs are (*a*) $|z| > 3$, (*b*) $|z| < 1/2$ *and* (*c*) $1/2 < |z| < 3$.

Solution We desire the partial fraction expansion of $X(z)/z$, which is

$$F(z) = \frac{X(z)}{z} = \frac{z+2}{z(2z^2 - 7z + 3)} = \frac{z+2}{2z\left(z - \frac{1}{2}\right)(z-3)} = \frac{A_0}{z} + \frac{A_1}{z - \frac{1}{2}} + \frac{A_2}{z-3}$$

$$\text{where, } A_0 = zF(z)\big|_{z=0} = \left.\frac{z+2}{2\left(z-\frac{1}{2}\right)(z-3)}\right|_{z=0} = \frac{2}{3}$$

$$A_1 = \left(z-\frac{1}{2}\right)F(z)\bigg|_{z=\frac{1}{2}} = \left.\frac{z+2}{2z(z-3)}\right|_{z=\frac{1}{2}} = -1$$

$$A_2 = (z-3)F(z)\big|_{z=3} = \left.\frac{z+2}{2z\left(z-\frac{1}{2}\right)}\right|_{z=3} = \frac{1}{3}$$

Hence, by multiplying $X(z)/z$ by z, we obtain

$$X(z) = \frac{2}{3} - \frac{z}{z-\frac{1}{2}} + \frac{z/3}{z-3}$$

Here, the given function $X(z)$ has two poles, $p_1 = \frac{1}{2}$ and $p_2 = 3$ and the following three inverse transforms.

(a) In the region $|z| > 3$, all poles are interior, i.e. the signal $x(n)$ is causal, and therefore,

$$x(n) = \frac{2}{3}\delta(n) - \left(\frac{1}{2}\right)^n u(n) + \frac{1}{3}(3)^n u(n).$$

(b) In the region $|z| < \frac{1}{2}$, both the poles are exterior, i.e. $x(n)$ is anti-causal and hence

$$x(n) = \frac{2}{3}\delta(n) - \left(\frac{1}{2}\right)^n u(-n-1) - \frac{1}{3}(3)^n u(-n-1)$$

(c) In the region $\frac{1}{2} < |z| < 3$, i.e. $|z| < 3$, the signal $x(n)$ is anti-causal and in the region $|z| > \frac{1}{2}$, the signal $x(n)$ is causal. Therefore, the pole $p_1 = \frac{1}{2}$ is interior and $p_2 = 3$ is exterior. Hence,

$$x(n) = \frac{2}{3}\delta(n) - \left(\frac{1}{2}\right)^n u(n) - \frac{1}{3}(3)^n u(-n-1)$$

Example 4.30 *Determine the inverse z-transform of*

$$X(z) = \frac{z}{3z^2 - 4z + 1}$$

if the regions of convergence are (a) $|z| > 1$, *(b)* $|z| < \frac{1}{3}$, *and (c)* $\frac{1}{3} < |z| < 1$

Solution The partial fraction expansion of $X(z)$ yields

$$F(z) = \frac{X(z)}{z} = \frac{1}{3z^2 - 4z + 1} = \frac{1}{3(z-1)\left(z-\frac{1}{3}\right)} = \frac{A_1}{(z-1)} + \frac{A_2}{\left(z-\frac{1}{3}\right)}$$

$$A_1 = F(z)(z-1)\big|_{z=1} = \left.\frac{1}{3\left(z-\frac{1}{3}\right)}\right|_{z=1} = \frac{1}{2}$$

$$A_2 = F(z)\left(z-\frac{1}{3}\right)_{z=\frac{1}{3}} = \frac{1}{3(z-1)}\bigg|_{z=\frac{1}{3}} = -\frac{1}{2}$$

$$\frac{X(z)}{z} = \frac{\frac{1}{2}}{(z-1)} + \frac{-\frac{1}{2}}{\left(z-\frac{1}{3}\right)}$$

$$X(z) = \frac{\frac{1}{2}z}{(z-1)} - \frac{\frac{1}{2}z}{\left(z-\frac{1}{3}\right)}$$

(a) When the ROC is $|z| > 1$, the signal $x(n)$ is causal and both terms are causal.

Therefore, $$x(n) = \frac{1}{2}(1)^n u(n) - \frac{1}{2}\left(\frac{1}{3}\right)^n u(n) = \frac{1}{2}\left[1-\left(\frac{1}{3}\right)^n\right]u(n)$$

(b) When the ROC is $|z| < \frac{1}{3}$, the signal $x(n)$ is anti-causal, i.e. the inverse gives negative time sequences. Therefore,

$$x(n) = \left[-\frac{1}{2}(1)^n + \frac{1}{2}\left(\frac{1}{3}\right)^n\right]u(-n-1)$$

(c) Here the ROC $\frac{1}{3} < |z| < 1$ is a ring, which implies that the signal $x(n)$ is two-sided. Therefore one of the terms corresponds to a causal signal and the other to an anti-causal signal. Obviously the given ROC is the overlapping of the regions $|z| > \frac{1}{3}$ and $|z| < 1$. The partial fraction expansion remains the same; however because of ROC, the pole at 1/3 provides the causal part corresponding to positive time and the pole at 1 is the anti-causal part corresponding to negative time. Therefore, $x(n)$ becomes

$$x(n) = -\frac{1}{2}(1)^n u(-n-1) - \frac{1}{2}\left(\frac{1}{3}\right)^n u(n)$$

Example 4.31 *Find x(n) using (a) long division, and (b) partial fraction for X(z) given by*

$$X(z) = \frac{2+3z^{-1}}{(1+z^{-1})\left(1+\frac{1}{2}z^{-1}\right)\left(1-\frac{1}{4}z^{-1}\right)}$$

Also verify the results in each case for $0 \le n \le 3$.

Solution

Long Division Method

$$X(z) = \frac{2+3z^{-1}}{1+\frac{5}{4}z^{-1}+\frac{1}{8}z^{-2}-\frac{1}{8}z^{-3}}$$

$$\begin{array}{r|l} & 2+\frac{1}{2}z^{-1}-\frac{7}{8}z^{-2}+\frac{41}{32}z^{-3} \\ \hline 1+\frac{5}{4}z^{-1}+\frac{1}{8}z^{-2}-\frac{1}{8}z^{-3} & 2+3z^{-1} \\ & 2+\frac{5}{4}z^{-1}+\frac{1}{4}z^{-2}-\frac{1}{4}z^{-3} \\ \hline & \frac{1}{2}z^{-1}-\frac{1}{4}z^{-2}+\frac{1}{4}z^{-3} \\ & \frac{1}{2}z^{-1}+\frac{5}{8}z^{-2}+\frac{1}{16}z^{-3}-\frac{1}{16}z^{-4} \\ \hline & -\frac{7}{8}z^{-2}+\frac{3}{16}z^{-3}+\frac{1}{16}z^{-4} \\ & -\frac{7}{8}z^{-2}+\frac{35}{32}z^{-3}-\frac{7}{64}z^{-4}+\frac{7}{64}z^{-5} \\ \hline & \frac{41}{32}z^{-3}+\ldots \end{array}$$

Therefore, $X(z)=2+\frac{1}{2}z^{-1}-\frac{7}{8}z^{-2}+\frac{41}{32}z^{-3}$

Taking inverse z-transform, we get $x(n)=\left\{\underset{\uparrow}{2},\frac{1}{2},-\frac{7}{8},\frac{41}{32},\ldots\right\}$

Partial Fraction Expansion Method

$$X(z)=\frac{2+3z^{-1}}{(1+z^{-1})\left(1+\frac{1}{2}z^{-1}\right)\left(1-\frac{1}{4}z^{-1}\right)}=\frac{A_1}{1+z^{-1}}+\frac{A_2}{1+\frac{1}{2}z^{-1}}+\frac{A_3}{1-\frac{1}{4}z^{-1}}$$

where $$A_1=\left.\frac{2+3z^{-1}}{\left(1+\frac{1}{2}z^{-1}\right)\left(1-\frac{1}{4}z^{-1}\right)}\right|_{z^{-1}=-1}=-\frac{8}{5}$$

$$A_2=\left.\frac{2+3z^{-1}}{\left(1+z^{-1}\right)\left(1-\frac{1}{4}z^{-1}\right)}\right|_{z^{-1}=-2}=\frac{8}{3}$$

and $$A_3=\left.\frac{2+3z^{-1}}{\left(1+z^{-1}\right)\left(1-\frac{1}{2}z^{-1}\right)}\right|_{z^{-1}=4}=\frac{14}{15}$$

$$X(z)=\frac{\left(-\frac{8}{5}\right)}{1+z^{-1}}+\frac{\left(\frac{8}{3}\right)}{1+\frac{1}{2}z^{-1}}+\frac{\left(\frac{14}{15}\right)}{1-\frac{1}{4}z^{-1}}$$

Hence $x(n)=Z^{-1}[X(z)]=\left[-\frac{8}{5}(-1)^n+\frac{8}{3}\left(-\frac{1}{2}\right)^n+\frac{14}{15}\left(\frac{1}{4}\right)^n\right]u(n)$

When $n = 0$, $x(0) = 2$

When $n = 1$, $x(1) = 1/2$

When $n = 2$, $x(2) = -7/8$

When $n = 3$, $x(3) = 41/32$

Therefore, $x(n) = \left\{ \begin{matrix} 2, \frac{1}{2}, -\frac{7}{8}, \frac{41}{32}, \ldots \\ \uparrow \end{matrix} \right\}$

Therefore, the results obtained in each case are identical.

4.4.3 Contour Integration—Residue Method

The basic definition of z-transform is given by

$$X(z) = \sum_{n=0}^{\infty} x(n) z^{-n}$$

Evaluating $X(z)$ at the complex number $z = re^{j\omega}$ gives

$$X(z)\big|_{z=re^{j\omega}} = \sum_{n=-\infty}^{\infty} x(n)(re^{j\omega})^{-n} = \sum_{n=-\infty}^{\infty} \left[x(n) r^{-n}\right] e^{-j\omega n}$$

Therefore, the basic relationship between z-transform and Fourier transform is given by $X(re^{j\omega}) = F\{x(n)r^{-n}\}$ for any value of r so that $z = re^{j\omega}$ is inside the ROC. Applying the inverse Fourier transform on both sides of the above equation yields

$$x(n)\, r^{-n} = F^{-1}\{X(re^{j\omega})\}$$
$$x(n) = r^{n} F^{-1}\{X(re^{j\omega})\}$$

Using the inverse Fourier transform expression in equation, we have

$$x(n) = r^{n} \frac{1}{2\pi} \int_{2\pi} X(re^{j\omega}) e^{j\omega n} \mathrm{d}\omega$$

Therefore, $x(n) = \frac{1}{2\pi} \int_{2\pi} X(re^{j\omega})(re^{j\omega})^{n} \mathrm{d}\omega$

$x(n)$ can be obtained from its z-transform evaluated along a contour $z = re^{j\omega}$ in the ROC, with r fixed and ω varying over a 2π interval. Here, the variable of integration ω is replaced with z. With $z = re^{j\omega}$ and r fixed, $dz = jre^{j\omega}\,\mathrm{d}\omega = jz\mathrm{d}\omega$. Hence, $\mathrm{d}\omega = \left(\frac{1}{j}\right) z^{-1} dz$. Therefore, the above equation can be rewritten as

$$x(n) = \frac{1}{2\pi j} \oint_{c} X(z) z^{n-1} \mathrm{d}z$$

where the symbol O denotes integration around a counterclockwise closed circular contour entered at the origin and with radius r. The value of r can be chosen as any value for which $X(z)$ converges.

By Cauchy's residue theorem, $\oint_{c} X(z) z^{n-1} dz$ = sum of the residues of $X(z)z^{n-1}$ at the isolated singularities.

This method is useful in determining the time-signal $x(n)$ by summing residues of $[X(z)z^{n-1}]$ at all poles. This is expressed mathematically as

$$x(z) = \sum_{\substack{\text{all poles}\\ X(z)}} \text{residues of } \left[X(z)z^{n-1}\right]$$

where the residue for a pole of order m at $z = \alpha$ is

$$\text{Residues} = \frac{1}{(m-1)!}\lim_{z\to\alpha}\left\{\frac{\mathrm{d}^{m-1}}{\mathrm{d}z^{m-1}}\left[(z-\alpha)^m X(z)z^{n-1}\right]\right\}$$

Example 4.32 *Find x(n) for X(Z) =* $\left[\dfrac{z^2}{(z-a)(z-b)}\right]$ *using Cauchy's residue theorem.*

Solution $x(n) = Z^{-1}\left[\dfrac{z^2}{(z-a)(z-b)}\right] = \dfrac{1}{2\pi i}\oint_C \dfrac{z^{n-1}.z^2}{(z-a)(z-b)}dz$, where C is the circle whose centre is the origin and which includes the singularites $z = a$ and $z = b$. Then $Z^{-1}\left[\dfrac{z^2}{(z-a)(z-b)}\right] = [\text{Res.}]_{z=a}$ $+(\text{Res.})_{z=b}$ of $\dfrac{z^{n+1}}{(z-a)(z-b)}$, by Cauchy's residue theorem.

Since $z = a$ and $z = b$ are simple poles, we get

$$(\text{Res.})_{z=a} = \left[\frac{z^{n+1}}{z-b}\right]_{z=a} = \frac{a^{n+1}}{a-b}$$

and $(\text{Res.})_{z=b} = \left[\dfrac{z^{n+1}}{z-a}\right]_{z=b} = \dfrac{b^{n+1}}{b-a}$

Therefore, $x(n)$ = sum of the residues of $X_n(z)$ at its poles

$$= \frac{1}{a-b}(a^{n+1} - b^{n+1}) = \frac{a}{a-b}.\, a^n - \frac{b}{a-b}.\, b^n$$

Example 4.33 *Determine the inverse z-transform of* $X(z) = \dfrac{z^2}{(z-a)}$, *ROC* $|z| > |a|$ *using contour integration–residue method.*

Solution Given $X(z)z^{n-1} = \dfrac{z^2}{(z-a)^2}z^{n-1} = \dfrac{z^{n+1}}{(z-a)^2}$

Here the pole is at $z = a$ and order $m = 2$. The residue of $X(z)\, z^{n-1}$ at $z = a$ can be calculated as

$$\operatorname*{Res}_{z=a}\left[X(z)z^{n-1}\right] = \frac{1}{(2-1)!}\lim_{z\to p_i}\left\{\frac{d^{2-1}}{dz^{2-1}}(z-p_i)^2 X(z)z^{n-1}\right\}_{z=a}$$

$$= \left\{\frac{d}{dz}(z-a)^2\frac{z^{n+1}}{(z-a)^2}\right\}_{z=a}$$

$$= \left[\frac{d}{dz}z^{n+1}\right]_{z=a} = \left[(n+1)z^n\right]_{z=a} = (n+1)a^n$$

We know that $x(n)$ is given as

$$x(n)=\sum_{i=1}\operatorname*{Res}_{z=a}\left[\,X(z)z^{n-1}\right]$$

Therefore, $x(n) = (n + 1)\, a^n\, u(n)$, since ROC $|z| > |a|$.

Example 4.34 *Using the residue method, determine $x(n)$ for*

$$X(z)=\frac{z}{(z-1)(z-2)}$$

Solution $X(z)$ has two poles of order $m = 1$ at $z = 1$ and at $z = 2$. The corresponding residues can be obtained as follows:

For poles at $z = 1$,

$$\text{Residue } =\frac{1}{0!}\lim_{z\to1}\left\{\frac{d^0}{dz^0}\left[(z-1)^1\frac{z\cdot z^{n-1}}{(z-1)(z-2)}\right]\right\}=\lim_{z\to1}\left[\frac{z}{z-2}z^{n-1}\right]=-1,$$

For poles at $z = 2$,

$$\text{Residue } =\frac{1}{0!}\lim_{z\to2}\left\{\frac{d^0}{dz^0}\left[(z-2)^1\frac{z\cdot z^{n-1}}{(z-1)(z-2)}\right]\right\}=\lim_{z\to2}\left[\frac{z}{z-1}z^{n-1}\right]=2\cdot2^{n-1}=2^n$$

Therefore, $x(n) = (-1 + 2^n)\, u(n)$.

Example 4.35 *Using the residue method, find the inverse z-transform of*

$$X(z)=\frac{1}{(z-0.25)(z-0.5)}\quad \text{ROC: } |z| > 0.5$$

Solution Let $X_0(z) = X(z)z^{n-1} = \dfrac{z^{n-1}}{(z-0.25)(z-0.5)}$

For convenience, let us write this as

$$X_0(z)=\frac{z^n}{z(z-0.25)(z-0.5)}$$

This has poles at $z = 0$, $z = 0.25$ and $z = 0.5$

By the residue theorem,

$$x(n)=\operatorname*{Res}_{z=0}[X_0(z)]+\operatorname*{Res}_{z=0.25}[X_0(z)]+\operatorname*{Res}_{z=0.5}[X_0(z)]$$

$$=\operatorname*{Res}_{z=0.5}\frac{z^n}{z(z-0.25)(z-0.5)}+\operatorname*{Res}_{z=0.25}\frac{z^n}{z(z-0.25)(z-0.5)}+\operatorname*{Res}_{z=0}\frac{z^n}{z(z-0.25)(z-0.5)}$$

(a) For pole at $z = 0$,

$$\operatorname*{Res}_{z=0}X_0(z)=zX_0(z)\Big|_{z=0}$$

$$=\left.\frac{z^n}{(z-0.25)(z-0.5)}\right|_{z=0}=0$$

(b) For pole at $z = 0.25$,

$$\operatorname*{Res}_{z=0.25} X_0(z) = (z-0.25)X_0(z)\Big|_{z=0.25}$$

$$= \frac{(z-0.25)z^n}{z(z-0.25)(z-0.5)}\Bigg|_{z=0.25} = \frac{z^{n-1}}{z-0.5}\Bigg|_{z=0.25} = \frac{z^{-1}z^n}{z-0.5}\Bigg|_{z=0.25}$$

$$\frac{4(1/4)^n}{-(1/4)} = -16(1/40)^n$$

(c) For pole at $z = 0.5$,

$$\operatorname*{Res}_{z=0.5} X_0(z) = (z-0.5)X_0(z)\Big|_{z=0.5}$$

$$= \frac{(z-0.5)z^n}{z(z-0.25)(z-0.5)}\Bigg|_{z=0.5} = \frac{z^{n-1}}{z-0.25}\Bigg|_{z=0.5}$$

$$\frac{2(1/2)^n}{1/4} = 8(1/2)^n$$

Therefore, $x(n) = [8(1/2)^n) - 16(1/4)^n]u(n)$

Example 4.36 *Obtain the inverse z-transform of*

$$X(z) = ln\,(1 + az^{-1}),\ |\,z\,| > |\,a\,|$$

Solution According to logarithmic series expansion,

$$\ln(1+x) = x - \frac{1}{2}x^2 + \frac{1}{3}x^3 - \ldots$$

Therefore, $X(z) = \ln\,(1+az^{-1})$

$$= az^{-1} - \frac{1}{2}(az^{-1})^2 + \frac{1}{3}(az^{-1})^3 - \ldots$$

$$= az^{-1} - \frac{1}{2}a^2z^{-2} + \frac{1}{3}a^3z^{-3} - \ldots$$

Taking inverse z-transform, we obtain

$$x(n) = \left\{\begin{matrix} 0, a, -\frac{1}{2}a^2, \frac{1}{3}a^3, \ldots \\ \uparrow \end{matrix}\right\}$$

Alternate Method

According to power series expansion,

$$\ln(1+x) = \sum_{n=1}^{\infty} \frac{-1}{n}(-x)^n$$

Therefore, $X(z) = \ln(1+az^{-1}) = \sum_{n=1}^{\infty} \frac{-1}{n}(-az^{-1})^n, |\,z\,| > |\,a\,|$

$$= \sum_{n=1}^{\infty} -\frac{1}{n}(-1)^n a^n z^{-n}, |\,z\,| > |\,a\,|$$

Taking inverse z-transform, we get

$$x(n) = \frac{(-1)^{n+1} a^n}{n} u(n-1)$$

Example 4.37 *Determine the inverse z-transform of* $X(z) = \frac{1}{(z+2)^2}, |z| < \frac{1}{2}$.

Solution $X(z) = \frac{1}{(z+2)^2}, |z| < \frac{1}{2}$

Since $|z| < \frac{1}{2}$, then $|z| > 2$

From Table 4.3, $Z^{-1}\left[\frac{az}{(z-a)^2}\right] = na^n u(n)$, ROC: $|z| > a$

Here $a = -2$

Hence $Z^{-1}\left[\frac{-2z}{(z+2)^2}\right] = (-2)^n nu(n)$

i.e. $-2Z^{-1}\left[\frac{z}{(z+2)^2}\right] = (-2)^n nu(n)$

Therefore, $Z^{-1}\left[\frac{z}{(z+2)^2}\right] = (-2)^{n-1} nu(n)$

Here, $X_1(z) = \frac{z}{(z+2)^2}$ and hence $x_1(n) = (-2)^{n-1} n\, u(n)$

Therefore, $X(z) = \frac{z}{(z+2)^2} = z^{-1} X_1(z)$

Using time shifting property, we get

$$Z^{-1}\left[z^{-1}X(z)\right] = Z^{-1}\left[\frac{1}{(z+2)^2}\right] = (-2)^{n-2}(n-1)u(n-1)$$

Example 4.38 *For a low-pass RC network (R = 1 MΩ and C = 1μF) shown in Fig. E4.38, determine the equivalent discrete time expressions for the circuit output response $y(n)$, when the input is $x(t) = e^{-2t}$ and the sampling frequency is f_s = 50 Hz.*

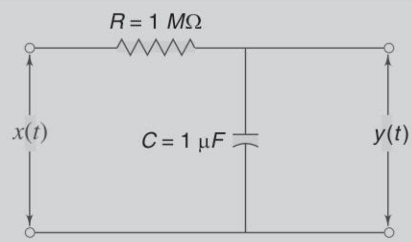


Fig. E4.38

Solution The transfer function of the given circuit in the s-domain can be expressed as

$$H(s) = \frac{1/RC}{s+1/RC} = \frac{1}{s+1}$$

Taking inverse Laplace transform, we get

$$h(t) = e^{-t}$$

***z*-domain Approach**

Using Table 4.3, the above transfer function may be expressed in z-plane as

$$H(z) = \frac{z}{z - e^{-T}}$$

Also, the given input function $x(t) = e^{-2t}$ may be expressed in the z-plane as

$$X(z) = \frac{z}{z - e^{-2T}}$$

We know that the output function $Y(z) = H(z)\,X(z)$.
Therefore,

$$Y(z) = \frac{z}{(z - e^{-T})} \cdot \frac{z}{(z - e^{-2T})}$$

$$\frac{Y(z)}{z} = \frac{z}{(z - e^{-T})(z - e^{-2T})}$$

Expressing the above as partial fractions

$$\frac{Y(z)}{z} = \frac{A_1}{z - e^{-T}} + \frac{A_2}{z - e^{-2T}}$$, where A_1 and A_2 are constants.

The constants A_1 and A_2 are evaluated as $A_1 = \dfrac{1}{1 - e^{-T}}$ and $A_2 = \dfrac{1}{1 - e^{T}}$

Thus, $\dfrac{Y(z)}{z} = \dfrac{\frac{1}{1 - e^{-T}}}{z - e^{-T}} + \dfrac{\frac{1}{1 - e^{T}}}{z - e^{-2T}}$

So, $Y(z) = \dfrac{1}{1 - e^{-T}} \cdot \dfrac{z}{z - e^{-T}} + \dfrac{1}{1 - e^{-T}} \cdot \dfrac{z}{z - e^{-2T}}$

Taking inverse z-transform, we get

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

Substituting $T = 1/50$, we get

$$y(n) = 50.5(0.980)^n - 49.5(0.961)^n$$

Thus the required output response is

$$y(n) = 50.5(0.980)^n - 49.5(0.961)^n$$

***s*-domain approach**

The above problem can be solved in the s-domain as given below.
The analog transfer function $H(s)$ is given by

$$H(s) = \frac{1}{s + 1}$$

The given input function is $x(t) = e^{-2t}$
Taking Laplace transform, we get

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

Hence, in the s-plane the output function is given by

$$Y(s) = X(s) \cdot H(s) = \frac{1}{(s+2)(s+1)}$$

In order to have (ideally) the same frequency response at $\omega = 0$ for both digital and analog systems, it is necessary to multiply the digital transfer function by a factor of T (as w increases, the two frequency response functions generally diverge). That is,

$$TY(z) = Y_T(z) = Y(s), \text{ where } T = \frac{1}{f_s}$$

$$Y_T(z) = Y(s)$$

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

Expressing as a partial fraction,

$$Y(s) = \frac{1}{s+1} - \frac{1}{s+2}$$

Hence, $Y(z) = \frac{1}{T} Y(s) = \frac{1}{T}\left[\frac{1}{s+1} - \frac{1}{s+2}\right]$

Using Table 4.3, we have

$$y(nT) = \frac{1}{T}[e^{-nT} - e^{-2nT}]$$

Substituting $T = 1/50$, we get

$$y(n) = 50[e^{-n/50} - e^{-2n/50}]$$
$$= 50[(1.0202)^{-n} - (1.0408)^{-n}]$$
$$= 50(0.980)^n - 50(0.961)^n$$

Thus the required output function is

$$y(nT) = 50(0.980)^n - 50(0.961)^n$$

Note: Due to the approximation (introduced by the multiplication of T) in the s-domain approach, we get a sequence that closely matches the sequence obtained through z-domain approach. Determination of value of T which gives the same sequence as that in z-domain is left as an exercise to the students.

REVIEW QUESTIONS

4.1 How is z-transform obtained from Laplace transform?
4.2 Describe the relationship between Laplace transform and z-transform.
4.3 Explain z-plane and s-plane correspondence.
4.4 Define z-transform.
4.5 Explain the use of z-transform.
4.6 State the properties of convergence for the z-transform.
4.7 State and explain the properties of z-transform.
4.8 Discuss the properties of two-sided z-transform and compare them with those of one-sided z-transform.
4.9 What is the condition for z-transform to exist?
4.10 Explain the shift property of z-transform.

4.11 With reference to z-transform, state the initial and final value theorems.
4.12 What is the z-transform convolution in time domain?
4.13 Describe the process of discrete convolution.
4.14 Define and explain (i) convolution, and (ii) correlation.
4.15 Define and explain cross-correlation and auto-correlation of sampled signals.
4.16 Define inverse z-transform.
4.17 Explain various methods of finding inverse z-transform.
4.18 Describe the signals shown in Fig. Q4.18 using sets of weighted, shifted, unit impulse functions.

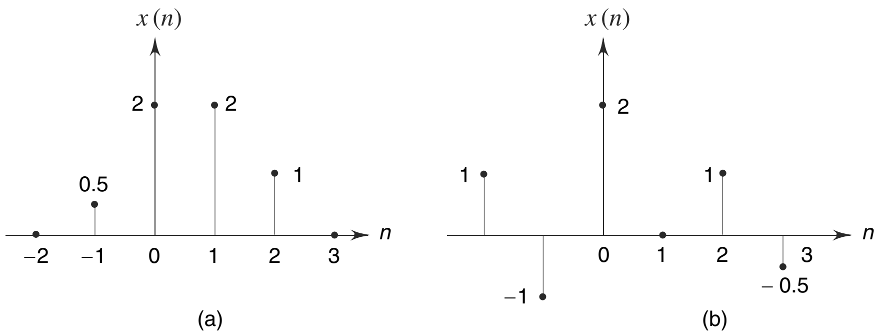


Fig. Q4.18

Ans: (a) $x(n) = 0.5\delta(n+1) + 2\delta(n) + 2\delta(n-1) + \delta(n-2)$
(b) $x(n) = \delta(n+2) - \delta(n+1) + 2\delta(n) + \delta(n-2) - 0.5\delta(n-3)$

4.19 Draw the pole-zero plot for the system described by the difference equation.

$$y(n) - \frac{3}{4}y(n-1) + \frac{1}{8}y(n-2) = x(n) - x(n-1)$$

4.20 Derive the z-transform of $f(nT) = \sin \omega nT$
4.21 Derive the z-transform of $f(nT) = \cos \omega nT$
4.22 Find the z-transform of $t^2 e^{-\alpha t}$.
Ans: $X(z) = 2[z^{-4} - z^{-3}(1 - e^{-\alpha} z^{-1})]/(1 - e^{-\alpha} z^{-1})$
4.23 Obtain the z-transform of $x(n) = -\alpha^n u(-n-1)$. Sketch the ROC.

Ans: $X(z) = \dfrac{1}{1 - az^{-1}}, |z| < |a|$

4.24 Find the z-transform of the following discrete-time signals including the region of convergence.

(i) $x(n) = e^{-3n}u(n-1)$ **Ans:** $X(z) = \dfrac{1}{1 - e^{-3}z^{-1}}, \text{ROC} |z| > |e^{-3}|$

(ii) $x(nT) = (nT)^2 u(nT)$ **Ans:** $X(z) = \dfrac{z^{-2}}{(1 - z^{-1})^2}, \text{ROC} |z| > 1$

(iii) $x(nT) = 0$, for $n < 0$
$= 1$, for $0 \le n \le 5$
$= 2$, for $6 \le n \le 10$
$= 3$, for $n > 10$

Ans: $X(z) = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5} + 2(z^{-6} + z^{-7} + z^{-8} + \ldots + z^{-10}) + 3(z^{-11} + z^{-12} + \cdots)$

4.25 Determine the z-transform of the following sequences

(a) $u(n-4)$ **Ans:** $\frac{z^{-4}}{1-z^{-1}}, |z|>1$

(b) $e^{jn\pi/4}u(n)$ **Ans:** $\frac{1}{1-e^{j\pi/4}z^{-1}}$

(c) $\delta(n-5)$ **Ans:** z^{-5}

(d) $\left(\frac{1}{3}\right)^n u(-n)$ **Ans:** $\frac{1}{1-3z}, |z|<\frac{1}{3}$

(e) $3^n u(n-2)$ **Ans:** $\frac{9z^{-2}}{1-3z^{-1}}$

4.26 Find the z-transform of the sequence $x(n) = n\alpha^n u(n)$

Ans: $X(z) = \frac{az^{-1}}{(1-az^{-1})^2}, |z|>|a|$

4.27 Find the two-sided z-transform of

$x(n) = (1/3)^n, \quad n \geq 0$

$= (-2)^n, \quad n \leq -1$

Ans: $X(z) = \frac{z}{z-\frac{1}{3}} + \frac{z}{z+2}$

4.28 Use convolution to find $x(n)$ if $X(z)$ is given by

$$X(z) = \frac{1}{\left(1-\frac{1}{2}z^{-1}\right)\left(1+\frac{1}{4}z^{-1}\right)}$$

Ans: $x(n) = \frac{2}{3}\left(\frac{1}{2}\right)^n u(n) + \frac{1}{3}\left(-\frac{1}{4}\right)^n u(n)$

4.29 Find $y(n)$ using the convolution property of z-transform when $x(n) = \{1, 2, 3, 1, -1, 1\}$ and $h(n) = \{1, 1, 1\}$

Ans: $y(n) = \left\{\begin{matrix}1,3,6,6,3,1,0,1\\ \uparrow\end{matrix}\right\}$

4.30 Convolve the sequences $x(n)$ and $h(n)$ where

$x(n) = 0, n < 0$

$= a^n, n \geq 0$

$h(n) = 0, n < 0$

$= b^n, n \geq 0$

Specify the answers if (i) $a \neq b$ and (ii) $a = b$.

Ans: (i) $y(n) = \frac{1}{a-b}\{a^n - b^n\}u(n), \quad \text{if } a \neq b$

(ii) $y(n) = n\,a^{n-1}\,u(n)$ or $n\,b^{n-1}\,u(n)$, if $a = b$

Note: $(a^n - b^n) = (a-b)(a^{n-1} + a^{n-2}b + a^{n-3}b^2 + \ldots + a\,b^{n-2} + b^{n-1})$

4.31 Compute the cross-correlation sequence $r_{xy}(l)$ of the sequences

$$x(n) = \left\{2, -1, 3, \underset{\uparrow}{7}, 1, 2, -3\right\}$$

$$y(n) = \left\{1, -1, 2, -2, \underset{\uparrow}{4}, 1, -2, 5\right\}$$

Ans : $r_{xy}(l) = \left\{10, -9, 19, 36, -14, 33, \underset{\uparrow}{0}, 7, 13, -18, 16, -7, 5, -3\right\}$

4.32 Compute and sketch the convolution $y(n)$ and correlation $r(n)$ sequences for the following pair of signals.

(a) $x_2(n) = \left\{\underset{\uparrow}{1}, 2, 3, 4\right\}, \quad h_2(n) = \left\{1, 2, 3, \underset{\uparrow}{4}\right\}$

Ans: $y_2(n) = \left\{\underset{\uparrow}{1}, 4, 10, 20, 25, 24, 16\right\}$

$r_2(n) = \left\{\underset{\uparrow}{4}, 11, 20, 30, 20, 11, 14\right\}$

(b) $x_3(n) = \left\{\underset{\uparrow}{1}, 2, 3, 4\right\}, \quad h(n) = \left\{\underset{\uparrow}{4}, 3, 2, 1\right\}$

Ans: $y_3(n) = \left\{\underset{\uparrow}{4}, 11, 20, 30, 20, 11, 4\right\}$

$r_3(n) = \left\{1, 4, 10, \underset{\uparrow}{20}, 25, 24, 16\right\}$

4.33 Find the autocorrelation sequence of the signal

$x(n) = a^n u(n) \quad \text{for} \quad -1 < a < 1$

Ans: $r_{xx}(l) = \dfrac{1}{1-a^2} a^{|l|} \quad \text{for} \quad -\infty < l < \infty$

4.34 Find the final value of the signal corresponding to the z-transform

$$X(z) = \frac{2z^{-1}}{1 - 1.8z^{-1} + 0.8z^{-2}}$$

Ans: 10

4.35 Prove that the final value of $x(n)$ for $X(z) = X(z) = \dfrac{z^2}{(z-1)(z-0.2)}$ is 1.25 and its initial value is unity.

4.36 Determine z-transform for the sequences given below

$x(0) = 1$
$x(1) = 4.7$
$x(2) = 0$
$x(3) = 0$
$x(4) = 0.75$
$x(5) = \sqrt{2}$
$x(n) = 0, \; n \geq 6$

Ans: $X(z) = 1 + 4.7z^{-1} + 0.75z^{-4} + \sqrt{2}\, z^{-5}$

4.37 Determine $x(0)$, $x(1)$, $x(2)$, $x(3)$, $x(4)$ for the following functions of z.

(i) $X(z) = \left(\dfrac{2z + z^{-2}}{z + z^{-1}}\right)$

Ans: $x(0) = 2$, $x(1) = 0$, $x(2) = -2$, $x(3) = 1$ and $x(4) = 2$

(ii) $X(z) = \dfrac{1}{1 - \frac{1}{2}z^{-1}}$

Ans: $x(0) = 1, x(1) = \frac{1}{2},\ x(2) = \frac{1}{4},\ x(3) = \frac{1}{8}$ and $x(4) = \frac{1}{16}$

4.38 Find the inverse z-transform of the following

(i) $X(z) = \dfrac{z^2}{(z-1)(z-0.2)}$

Ans: $x(n) = [1.25 - 0.25\,(0.2)^n]\,u(n)$
$x(n) = \{1, 1.2, 1.24, 1.248, ...\}$

(ii) $X(z) = \dfrac{z^2}{(z - 0.5 - j0.5)(z - 0.5 + j0.5)}$

Ans: $x(n) = -j(0.5 + j\,0.5)^{n+1} + j(0.5 - j\,0.5)^{n+1}$

(iii) $X(z) = \dfrac{z}{(z-1)^3}$

Ans: $x(n) = \dfrac{n(n-1)u(n)}{2}$

(iv) $X(Z) = \dfrac{z(1-e^{-T})}{(z-1)(z-e^{-T})}$

Ans: $x(n) = (1 - e^{-nT})u(n)$

4.39 Find the inverse of z-transform of $\dfrac{1}{1-4z^{-1}}$.

Ans: $(4)^n\,u(n)$

4.40 Determine $x(n)$ for $X(z) = \dfrac{z^{-1}}{1-3z^{-1}}$ with ROC: $|z| < 3$

Ans: $x(n) = (-3)^{n-1}\,u(n)$

4.41 Find the sequence $x(n)$, $n \geq 0$ given that $X(z) = \dfrac{1}{\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{1}{4}z^{-1}\right)}$

Ans: $\left[2\left(\frac{1}{2}\right)^n - \left(\frac{1}{4}\right)^n\right]u(n)$

4.42 Using partial fraction, find $x(n)$ for $X(z) = \dfrac{1-z^{-1}}{(1+z^{-1})\left(1+\frac{1}{2}z^{-1}\right)}$

4.43 Using partial fraction expansion, determine $x(n)$ for $X(z)$ given by

(a) $X(z) = \dfrac{(z-0.5)}{z(z-0.8)(z-1)}$

Ans: $x(n) = [2.5 - 1.5(0.8)^{n-2}]u(n-2)$

(b) $X(z)=\dfrac{0.5z}{z^2-z+0.5}$

Ans: $x(n)=\left(\dfrac{1}{\sqrt{2}}\right)^n \sin\left(\dfrac{n\pi}{4}\right)u(n)$

4.44 Find $x(n)$ if $X(z)=\dfrac{z^{-3}}{1-\frac{1}{4}z^{-1}}$

Ans: $x(n)=\left(\dfrac{1}{4}\right)^{n-3} u(n-3)$

4.45 Find $x(n)$ if $X(z)=\dfrac{z+3}{z^7\left(z-\frac{1}{2}\right)}$

Ans: $x(n)=\left(\dfrac{1}{2}\right)^{n-7} u(n-7)+3\left(\dfrac{1}{2}\right)^{n-8} u(n-8)$

4.46 Find the causal signal $x(n)$ for $X(z)=\dfrac{1+z^{-1}}{1-z^{-1}+0.5z^{-2}}$

Ans: $x(n)=\sqrt{10}\left(\dfrac{1}{\sqrt{2}}\right)^n \cos\left(\dfrac{\pi n}{4}-71.565^\circ\right)u(n)$

4.47 Find the inverse z-transform of $X(z)=\dfrac{z}{3z^2-4z+1}$

where the ROC is (i) $|z|>1$ and (ii) $|z|<\dfrac{1}{3}$ using the long division method.

Ans: (i) $x(n)=\left\{\underset{\uparrow}{0}, \dfrac{1}{3}, \dfrac{4}{9}, \dfrac{13}{27}, \dfrac{40}{81}, \ldots\right\}$

(ii) $x(n)=\left\{\ldots 121, 40, 13, 4, 1, \underset{\uparrow}{0}\right\}$

4.48 Using long division, determine the inverse z-transform of

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

if (a) $x(n)$ is causal and (b) $x(n)$ is anti-causal

Ans: (a) $x(n)=\left\{\underset{\uparrow}{1}, 4, 7, 10, 13\ldots\right\}$ (b) $x(n)=\left\{\ldots 14, 11, 8, 5, 2, \underset{\uparrow}{0}\right\}$

4.49 Determine the causal signal $x(n)$ having the z-transform

$X(z)=\dfrac{z^2+z}{\left(z-\frac{1}{2}\right)^3\left(z-\frac{1}{4}\right)}$ for the region of convergence $|z|>\dfrac{1}{2}$

Ans: $x(n)=\left\{80\left(\dfrac{1}{2}\right)^n-20n\left(\dfrac{1}{2}\right)^{n-1}+6[n(n-1)/2]\left(\dfrac{1}{2}\right)^{n-2}-80\left(\dfrac{1}{4}^n\right)\right\}u(n)$

4.50 Using (i) the long division method, (ii) partial fraction method, and (iii) residue method, find $x(n)$ and verify the results in each case for n in the range $0 \le n \le 3$.

(a) $X(z) = \dfrac{z+3}{z-0.25}$

Ans: (i) $x(0) = 1$, $x(1) = 3.25$, $x(2) = 0.8125$, $x(3) = 0.203125$

(ii) $x(n) = -12\delta(n) + 13(0.25)^n$

$x(0) = 1$, $x(1) = 3.25$, $x(2) = 0.8125$, $x(3) = 0.203125$

(iii) $x(n) = 13(0.25)^n$

$x(0) = 13$, $x(1) = 3.25$, $x(2) = 0.8125$, $x(3) = 0.203125$

Here $x(0) = 13$ is not correct. The correct initial value [$x(0) = 1$] can be obtained using the initial value theorem.

(b) $X(z) = \dfrac{1.8z(z+0.1667)}{z^2 - 0.4z - 0.05}$

Ans: (i) $x(0) = 1.8$, $x(1) = 1.02$, $x(2) = 0.498$, $x(3) = 0.2502$

(ii) $x(n) = 2(0.5)^n - 0.2(-0.1)^n$

$x(0) = 1.8$, $x(1) = 1.02$, $x(2) = 0.498$, $x(3) = 0.2502$

(iii) $x(n) = 2(0.5)^n - 0.2(-0.1)^n$

$x(0) = 1.8$, $x(1) = 1.02$, $x(2) = 0.498$, $x(3) = 0.2502$

4.51 For a low-pass RC network ($R = 1$ MΩ and $C = 1$ μF). Determine the output response for n in the range $0 \le n \le 3$ when the input has a step response of magnitude 2 V and the sampling frequency $f_s = 50$ Hz. Confirm the calculated values using the principle of discrete-time convolution.

Ans: $y(0) = 2$, $y(1) = 3.96$, $y(2) = 5.88$, and $y(3) = 7.77$

Chapter 5 Linear Time Invariant Systems

INTRODUCTION 5.1

The techniques and applications of Digital Signal Processing (DSP) require the understanding of basic concepts and terminology of discrete-time, causal, linear and time invariant systems. A DSP system can be represented by the block diagram shown in Fig. 5.1. The input signal $x(n)$, is the system excitation, and $y(n)$ is the response of the system. The system output $y(n)$ is related to the excitation mathematically as

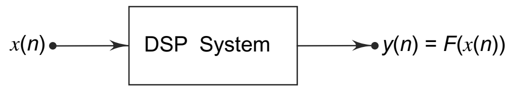


Fig. 5.1 *A DSP System*

$$y(n) = F[x(n)] \tag{5.1}$$

where F is an operator.

5.1.1 Impulse Response

The unit sample response or simply the impulse response $h(n)$ of a linear time invariant system is the system's response to a unit impulse input signal $[\delta(n)]$ located at $n=0$, when the initial conditions of the system are zero.

$$\delta(n) = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \tag{5.2}$$

If the input to the system is the time shifted version of the δ function, the output is a time shifted version of the impulse response. The result shown in Fig. 5.2 is due to the time invariance of the system. In this linear system, application of $\alpha\,\delta(n-k)$ at the input will produce $\alpha h(n-k)$ at the output as shown in Fig. 5.3.

The shifted impulse sequence is given by

$$\delta(n-k) = \begin{cases} 1, & n = k \\ 0, & n \neq k \end{cases}$$

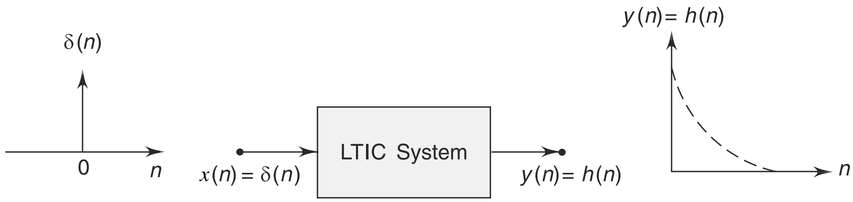


Fig. 5.2 *Impulse Response of a Linear Time Invariant Causal (LTIC) System*

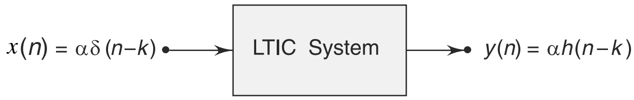


Fig. 5.3 *Time Shifted Impulse Response*

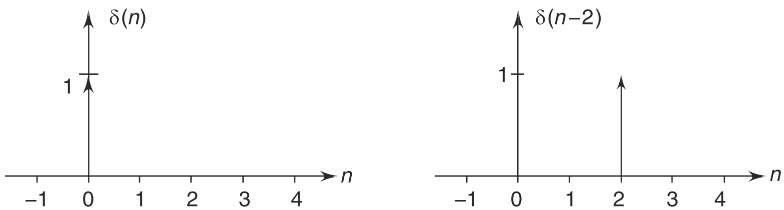


Fig. 5.4 *(a) Graphical Representation of the Unit-impulse Sequence δ(n) and (b) The Shifted Unit-impulse Sequence δ(n− 2)*

where k is an integer. The graphs of $\delta(n)$ and $\delta(n-2)$ are shown in Fig. 5.4.

The response of the LTI system to the unit impulse sequence $\delta(n)$ is represented by $h(n)$, i.e., $h(n) = F[\delta(n)]$ and hence $h(n-k) = F[\delta(n-k)]$.

Therefore,
$$y(n) = F[x(n)] = F\left[\sum_{k=-\infty}^{\infty} x(k)\delta(n-k)\right]$$
$$= \sum_{k=-\infty}^{\infty} x(k)F[\delta(n-k)] = \sum_{k=-\infty}^{\infty} x(k)h(n-k) = x(n) * h(n)$$

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

This expression gives the output response $y(n)$ of the LTI system as a function of the input signal $x(n)$ and the unit impulse (sample) response $h(n)$ and is referred to as a **convolution sum**.

5.1.2 Unit Step Response [$u(n)$]

The unit step sequence $u(n)$ is defined by

$$u(n) = \begin{cases} 0, & n < 0 \\ 1, & n \geq 0 \end{cases} \tag{5.3}$$

The shifted unit step sequence $u(n-k)$ is given by

$$u(n-k) = \begin{cases} 0, & n < k \\ 1, & n \geq k \end{cases}$$

The graphical representations of $u(n)$ and $u(n-2)$ are shown in Fig. 5.5.

The step response can be obtained by exciting the input of the system by a unit-step sequence, i.e., $x(n) = u(n)$. Hence, the output response $y(n)$ is obtained by using the convolution formula as

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

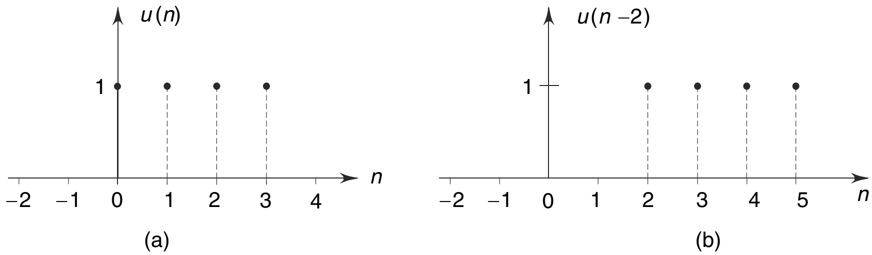


Fig. 5.5 *(a) The Unit-step Sequence u(n) and (b) The Shifted Unit-step Sequence u(n − 2)*

Relation between the Unit Sample and the Unit-step Sequences

The unit sample sequence $\delta(n)$ and the unit-step sequence $u(n)$ are related as

$$u(n) = \sum_{m=0}^{\infty} \delta(m), \quad \delta(n) = u(n) - u(n-1) \tag{5.4}$$

PROPERTIES OF A DSP SYSTEM 5.2

The properties of linearity, time invariance, causality and stability of the difference equations are required for a DSP system to be practically realisable.

5.2.1 Linearity

If $y_1(n)$ and $y_2(n)$ are the output sequences in response to the two arbitrary input sequences $x_1(n)$ and $x_2(n)$, respectively, then the condition for the DSP system to be linear is given by

$$\begin{aligned} F[a_1x_1(n)+a_2x_2(n)] &= a_1F[x_1(n)]+a_2F[x_2(n)] \\ &= a_1y_1(n)+a_2y_2(n) \end{aligned} \tag{5.5}$$

where a_1 and a_2 are two arbitrary constants and $F[\ ]$ denotes a signal processing operator performed by the DSP system. This condition is often called the principle of **superposition**. All linear systems possess the property of superposition.

Example 5.1 *Check whether the following systems are linear.*

(a) $F[x(n)] = a\,n\,x(n) + b$ (b) $F[x(n)] = e^{x(n)}$

Solution

(a) $F[x(n)] = a\,n\,x(n) + b$

$F[x_1(n)+x_2(n)] = a\,n[x_1(n)+x_2(n)] + b$

$F[x_1(n)]+F[x_2(n)] = [a\,n\,x_1(n)+b]+[an\,x_2(n)+b]$

Since $F[x_1(n)+x_2(n)] \neq F[x_1(n)]+F[x_2(n)]$, the system is non-linear, when $b \neq 0$.

(b) Given $F[x(n)] = e^{x(n)}$

$$F[x_1(n)+x_2(n)] = e^{x_1(n)+x_2(n)} = e^{x_1(n)}e^{x_2(n)}$$

$$F[x_1(n)]+F[x_2(n)] = e^{x_1(n)} + e^{x_2(n)}$$

Since $F[x_1(n)+x_2(n)] \neq F[x_1(n)]+F[x_2(n)]$, the system is non-linear.

Example 5.2 *Check whether the following systems are linear.*

(a) $y(n)=\frac{1}{N}\sum_{m=0}^{N-1}x(n-m)$ (b) $y(n)=[x(n)]^2$

Solution

(a) Let $y_1(n)=\frac{1}{N}\sum_{m=0}^{N-1}x_1(n-m)$ and $y_2(n)=\frac{1}{N}\sum_{m=0}^{N-1}x_2(n-m)$

then, $y(n)=y_1(n)+y_2(n)=\frac{1}{N}\sum_{m=0}^{N-1}x_1(n-m)+\frac{1}{N}\sum_{m=0}^{N-1}x_2(n-m)$

$$=\frac{1}{N}\sum_{m=0}^{N-1}\left[x_1(n-m)+x_2(n-m)\right]=\frac{1}{N}\sum_{m=0}^{N-1}x(n-m)$$

Therefore, this system is linear.

(b) $y(n) = [x_1(n)+x_2(n)]^2$
$= [x_1(n)]^2+2x_1(n)\,x_2(n)+[x_2(n)]^2$
$\neq [x_1(n)]^2+[x_2(n)]^2$

Therefore, the system is non-linear.

Example 5.3 *Check the following responses of the digital filters for linearity,*
(a) $y(nT)=F[x(nT)]=9x^2(nT-T)$ (b) $y(nT)=F[x(nT)]=(nT)^2\,x(nT+2T)$

Solution

(a) The condition for linearity is,
$F[ax_1(nT)+bx_2(nT)] = aF[x_1(nT)]+bF[x_2(nT)]$
Let the response of the system to $x_1(nT)$ be $y_1(nT)$ and the response of the system to $x_2(nT)$ be $y_2(nT)$. Thus for the input $x_1(nT)$, the describing equation is,

$$y_1(nT) = 9x_1^2(nT-T)$$
$$y_2(nT) = 9x_2^2(nT-T)$$

Multiplying these equations by a and b respectively and adding yields,

$$ay_1(nT) + by_2(nT) = 9ax_1^2(nT-T) + 9bx_2^2(nT-T) \quad (1)$$

and $$F[ax_1(nT) + bx_2(nT)] = 9[ax_1(nT-T) + bx_2(nT-T)]^2 \quad (2)$$

On comparing (1) and (2), the system is non-linear.

(b) In this case,

$$F[ax_1(nT) + bx_2(nT)] = (nT)^2\,[ax_1(nT+2T) + bx_2(nT+2T)]$$
$$= a(nT)^2\,x_1(nT+2T) + b(nT)^2\,x_2(nT+2T)$$
$$= aF\,[x_1(nT)] + bF[x_2(nT)]$$

Therefore, the filter is linear.

Example 5.4 *Determine whether the system described by the differential equation*

$y(n)=2x(n)+\frac{1}{x(n-1)}$ is linear.

Solution Let the response of the system to $x_1(n)$ be $y_1(n)$ and the response of the system to $x_2(n)$ be $y_2(n)$. Thus, for input $x_1(n)$, the describing equation is

$$y_1(n) = 2x_1(n) + \frac{1}{x_1(n-1)} \text{ and for input } x_2(n),$$

$$y_2(n) = 2x_2(n) + \frac{1}{x_2(n-1)}$$

Multiplying these equations by 'a' and 'b' respectively and adding, we get

$$\left[ay_1(n) + by_2(n)\right] = 2\left[ax_1(n) + bx_2(n)\right] + \frac{a}{x_1(n-1)} + \frac{b}{x_2(n-1)}$$

This equation cannot be put into the same form as the original difference equation describing the system. Hence, the system is non-linear.

Example 5.5 *Test the following systems for linearity*
(a) $y(t) = 5 \sin x(t)$ *and* (b) $y(t) = 7\,x(t) + 5$

Solution $y(t) = 5 \sin x(t)$

(a) The condition for linearity is,

$$F[ax_1(t) + bx_2(t)] = aF[x_1(t)] + bF[x_2(t)]$$

Let the response of the system to $x_1(t)$ be $y_1(t)$ and the response of the system to $x_2(t)$ be $y_2(t)$. Thus, for the input $x_1(t)$, the describing equation is,

$$y_1(t) = F[x_1(t)] = 5 \sin x_1(t) \text{ and for input } x_2(t) \text{ the equation is}$$

$$y_2(t) = F[x_2(t)] = 5 \sin x_2(t)$$

Multiplying these equations by a and b respectively and adding yields,

$$ay_1(t) + by_2(t) = 5[a \sin x_1(t) + b \sin x_2(t)] \quad (1)$$

whereas, $F[ax_1(t) + bx_2(t)] = 5 \sin [ax_1(t) + bx_2(t)]$ (2)

On comparing (1) and (2), the system is non-linear.

(b) $y(t) = 7x(t) + 5$

Hence,
$$y_1(t) = F[x_1(t)] = 7x_1(t) + 5$$
$$y_2(t) = F[x_2(t)] = 7x_2(t) + 5$$

Multiplying these equations by a and b respectively and adding yields,

$$ay_1(t) + by_2(t) = a[7x_1(t) + 5] + b[7x_2(t) + 5]$$
$$= 7[ax_1(t) + bx_2(t)] + 5[a + b] \quad (1)$$

whereas, $F[ax_1(t) + bx_2(t)] = 7[ax_1(t) + bx_2(t)] + 5$ (2)

On comparing (1) and (2), the system is non-linear.

Example 5.6 *Show that the system described by the differential equation*

$\frac{dy(t)}{dt} + 10y(t) + 5 = x(t)$ *is non-linear.*

Solution For the input $x_1(t)$, the describing equation is

$$\frac{dy_1(t)}{dt} + 10y_1(t) + 5 = x_1(t)$$

and for the input $x_2(t)$,

$$\frac{dy_2(t)}{dt} + 10y_2(t) + 5 = x_2(t)$$

Multiplying the above equations by a and b respectively and adding yields,

$$a\frac{dy_1(t)}{dt} + 10ay_1(t) + 5a + b\frac{dy_2(t)}{dt} + 10by_2(t) + 5b = ax_1(t) + bx_2(t)$$

i.e. $$\frac{d}{dt}\left[ay_1(t) + by_2(t)\right] + 10\left[ay_1(t) + by_2(t)\right] + 5\left[a + b\right] = ax_1(t) + bx_2(t)$$

This equation cannot be put into the same form as the original differential equation describing the system. Hence the system is non-linear.

Example 5.7 *Show that the system described by the differential equation* $\frac{dy(t)}{dt} + ty(t) = x(t)$ *is linear.*

Solution For the input $x_1(t)$, the describing equation is

$$\frac{dy_1(t)}{dt} + ty_1(t) = x_1(t)$$

and for the input $x_2(t)$,

$$\frac{dy_2(t)}{dt} + ty_2(t) = x_2(t)$$

Multiplying the above equations by a and b respectively and adding yields,

$$a\frac{dy_1(t)}{dt} + tay_1(t) + b\frac{dy_2(t)}{dt} + tby_2(t) = ax_1(t) + bx_2(t)$$

i.e. $$\frac{d}{dt}\left[ay_1(t) + by_2(t)\right] + t\left[ay_1(t) + by_2(t)\right] = ax_1(t) + bx_2(t)$$

This equation satisfies the superposition theorem and hence the system is linear.

Example 5.8 *Show that* $\frac{d^2y(t)}{dt^2} + y(t)\frac{dy(t)}{dt} + y(t) = x(t)$ *is non-linear.*

Solution The response to $x_1(t)$, $y_1(t)$ satisfies

$$F[x_1(t)] = \frac{d^2y_1(t)}{dt^2} + y_1(t)\frac{dy_1(t)}{dt} + y_1(t)$$

and the response to $x_2(t)$, $y_2(t)$ satisfies

$$F[x_2(t)] = \frac{d^2 y_2(t)}{dt^2} + y_2(t)\frac{dy_2(t)}{dt} + y_2(t)$$

Therefore,

$$aF\left[x_1(t)\right] + bF\left[x_2(t)\right] = a\frac{d^2 y_1(t)}{dt^2} + b\frac{d^2 y_2(t)}{dt^2} + ay_1(t)\frac{dy_1(t)}{dt} + by_2(t)\frac{dy_2(t)}{dt} + ay_1(t) + by_2(t)$$

$$F\left[ax_1(t) + bx_2(t)\right] = \frac{d^2}{dt^2}\left[ay_1(t) + by_2(t)\right] + \left[ay_1(t) + by_2(t)\right]\left[\frac{d}{dt}\left(ay_1(t) + by_2(t)\right) + 1\right]$$

Here $aF[x_1(t)] + bF[x_2(t)] \neq F[ax_1(t) + bx_2(t)]$ and hence the system is non-linear.

Example 5.9 *Determine whether the system described by the differential equation* $\frac{dy(t)}{dt} + 3ty(t) = t^2 x(t)$ *is linear.*

Solution

Let the response of the system to $x_1(t)$ be $y_1(t)$ and response of the system to $x_2(t)$ be $y_2(t)$.

$$\frac{dy_1(t)}{dt} + 3ty_1(t) = t^2 x_1(t)$$

and for the input $x_2(t)$,

$$\frac{dy_2(t)}{dt} + 3ty_2(t) = t^2 x_2(t)$$

Multiplying these equations by 'a' and 'b' respectively and adding, we get

$$\frac{d}{dt}\left[ay_1(t) + by_2(t)\right] + 3t\left[ay_1(t) + by_2(t)\right] = t^2\left[ax_1(t) + bx_2(t)\right]$$

The response of the system to the input $ax_1(t) + bx_2(t)$ is $ay_1(t) + by_2(t)$. Thus, the superposition condition is satisfied and hence, the system is linear.

5.2.2 Time-Invariance

A DSP system is said to be time-invariant if the relationship between the input and output does not change with time. It is mathematically defined as

If $y(n) = F[x(n)]$, then $y(n-k) = F[x(n-k)] = z^{-k} F[x(n)]$ (5.6)

for all values of k. This is true for all possible excitations. The operator z^{-k} represents a signal delay of k samples.

Example 5.10 *Determine whether the following systems are time variant or not:*

(a) $y(t) = tx(t)$ (b) $y(n) = x(n^2)$ (c) $y(n) = 2x(n)$ (d) $y(n) = x(n) + nx(n+1)$

Solution

(a) Given $y(t) = tx(t)$.

The response of the system to the shifted input is

$$F[x(t-\tau)] = tx(t-\tau)$$

If the response of the system is shifted by the same amount,

$$y(t - \tau) = (t - \tau)\, x(t - \tau)$$

Here, $y(t - \tau) \neq F[x(t - \tau)]$

Hence, the system is time variant.

(b) Given $y(n) = x(n^2)$

The response of the system to the shifted input is

$$F[x(n - k)] = [x(n - k)^2]$$

If the response of the system is shifted by the same amount,

$$y(n - k) = x\,[n^2 - k]$$

Here, $F[x(n - k)] \neq y(n - k)$

Hence, the system is time variant.

(c) Given $y(n) = 2x(n)$

The response of the system to the shifted input is

$$F[x(n - k)] = 2x(n - k)$$

If the response of the system is shifted by the same amount,

$$y(n - k) = 2x(n - k)$$

Here, $y(n - k)] = F[x(n - k)]$

Hence, the system is time in variant.

(d) Given $y(n) = x(n) + nx(n + 1)$

The response of the system to the shifted input is

$$F[x(n - k)] = x(n - k) + nx[n - k + 1]$$

If the response of the system is shifted by the same amount,

$$y(n - k) = x(n - k) + (n - k)\, x[n - k + 1]$$

Here, $F[x(n - k)] \neq y(n - k)$

Hence, the system is time variant.

Example 5.11 *Determine whether the DSP systems described by the following equations are time invariant.*

(a) $y(n) = F[x(n)] = a\, n\, x(n)$ (b) $y(n) = F[x(n)] = a\, x(n-1) + b\, x(n-2)$

Solution

(a) The response to a delayed excitation is

$$F[x(n-k)] = an\,[x(n-k)]$$

The delayed response is $y(n-k) = a(n-k)\,[x(n-k)]$

Here $F[x(n-k)] \neq y(n-k)$ and hence the system is not time invariant, i.e. the system is time dependent.

(b) Here, $F[x(n-k)] = ax[(n-k)-1] + bx[(n-k)-2]$

$$= y(n-k)$$

Hence the system is time invariant.

Example 5.12 *Check whether the following systems are linear and time invariant.*

(a) $F[x(n)] = n[x(n)]^2$ (b) $F[x(n)] = a[x(n)]^2 + b\,x(n)$

Solution

(a) (i) $F[x(n)] = n[x(n)]^2$

Here, $F[x_1(n)] = n[x_1(n)]^2$ and

$F[x_2(n)] = n\,[x_2(n)]^2$

Therefore, $F[x_1(n)] + F[x_2(n)] = n[\{x_1(n)\}^2 + \{x_2(n)\}^2]$

Further, $F[x_1(n) + x_2(n)] = n[x_1(n) + x_2(n)]^2$

$= n[\{x_1(n)\}^2 + \{x_2(n)\}^2 + 2x_1(n)x_2(n)]$

Here, $F[x_1(n) + x_2(n)] \neq F[x_1(n)] + F[x_2(n)]$ and hence the system is non-linear.

(ii) $F[x(n)] = n[x(n)]^2 = y(n)$

The response of delayed excitation is

$F[x(n-k)] = n[x(n-k)]^2$

The delayed response is

$y(n-k) = (n-k)\,[x(n-k)]^2$

Here, $y(n-k) \neq F[x(n-k)]$

Therefore, the system is not time invariant, i.e. time dependent.

(b) (i) $F[x(n)] = a[x(n)]^2 + bx(n)$

Here, $F[x_1(n)] = a[x_1(n)]^2 + bx_1(n)$ and

$F[x_2(n)] = a[x_2(n)]^2 + bx_2(n)$

Therefore, $F[x_1(n)] + F[x_2(n)] = a[\{x_1(n)\}^2 + \{x_2(n)\}^2] + b[x_1(n) + x_2(n)]$

Also, $F[x_1(n) + x_2(n)] = a[x_1(n) + x_2(n)]^2 + b[x_1(n) + x_2(n)]$

$= a[\{x_1(n)\}^2 + \{x_2(n)\}^2 + 2x_1(n)\,x_2(n)] + bx_1(n) + bx_2(n)$

Here, $F[x_1(n)\} + x_2(n)] \neq F[x_1(n)] + F[x_2(n)]$

Therefore, the system is non-linear.

(ii) $F[x(n)] = y(n) = a[x(n)]^2 + b[x(n)]$

The response of delayed excitation is

$F[x(n-k)] = a[x(n-k)]^2 + bx(n-k)$

The delayed response is

$y(n-k) = a[x(n-k)]^2 + b[x(n-k)]$

Since $y(n-k)] = F[x(n-k)]$, the system is time invariant.

5.2.3 Causality

Causal systems are those in which changes in the output are dependent only on the present and past values of the input and/or previous output values, and are not dependent on future input values. For a linear time invariant system, the causality condition is given by

$$h(n) = 0 \text{ for } n < 0 \tag{5.7}$$

Example 5.13 *Show that a relaxed linear time invariant system is causal if and only if $h(n) = 0$, for $n < 0$.*

Proof

Condition for an LTI Discrete-Time System to be Causal

Consider $x_1(n)$ and $x_2(n)$ are two input sequences with $x_1(n) = x_2(n)$ for $n \le n_0$. (1)

Using convolution sum, the corresponding output samples at $n=n_0$ of an LTI discrete-time system with an impulse response $h(n)$ are obtained by

$$y_1(n_0) = \sum_{k=-\infty}^{\infty} h(k)\,x_1(n_0-k)$$

$$= \sum_{k=0}^{\infty} h(k)\,x_1(n_0-k) + \sum_{k=-\infty}^{-1} h(k)\,x_1(n_0-k) \tag{2}$$

$$y_2(n_0) = \sum_{k=-\infty}^{\infty} h(k)\,x_2(n_0-k)$$

$$= \sum_{k=0}^{\infty} h(k)\,x_2(n_0-k) + \sum_{k=-\infty}^{-1} h(k)\,x_2(n_0-k) \tag{3}$$

For the LTI discrete-time system to be causal, $y_1(n_0)$ should be equal to $y_2(n_0)$. Due to Eq. (1), the first sum on the right hand side of Eq. (2) is equal to the first sum on the right hand side of Eq. (3). Hence, the second sums on the right hand side of Eqs. (2) and (3) should be equal. When $x_1(n)$ is not equal to $x_2(n)$ for $n>n_0$, these two sums will be equal if both are equal to zero, which is satisfied if

$$h(k) = 0 \text{ for } k<0 \tag{4}$$

Hence, an LTI discrete time system is causal if and only if its impulse response sequence $h(n)$ satisfies the condition of Eq. (4).

Note: A system with all zero initial conditions is called a relaxed system. The causal discrete-time system is linear only for zero initial conditions.

Example 5.14 *Check whether the DSP systems described by the following equations are causal.*

(a) $y(n)=3\,x(n-2)+3\,x(n+2)$ (b) $y(n)=x(n-1)+a\,x(n-2)$ (c) $y(n)=x(-n)$

Solution

(a) $y(n) = 3\,x(n-2)+3\,x(n+2)$

Here, $y(n)$ is determined using the past input sample value $3\,x(n-2)$ and future input sample value $3\,x(n+2)$ and hence the system is non-causal.

(b) $y(n) = x(n-1)+ax(n-2)$

Here, $y(n)$ is computed using only the previous input sample values $x(n-1)$ and $ax(n-2)$. Hence, the system is causal.

(c) $y(n) = x(-n)$

Here, the input sample value is located on the negative time axis and the sample values cannot be obtained before $t=0$. Hence the system is non-causal.

Example 5.15 *The discrete-time systems are represented by the following difference equations in which $x(n)$ is input and $y(n)$ is output.*

(a) $y(n)=3y^2(n-1)-nx(n)+4\,x(n-1)-2x(n+1)$ *and*

(b) $y(n)=x(n+1)-3x(n)+x(n-1);\ n\geq 0$

Are these systems linear? shift-invariant? causal? In each case, justify your answer.

Solution

(i) *Linearity*

(a) $y(n)=3y^2(n-1)-nx(n)+4x(n-1)-2x(n+1)$

The condition for linearity is,

$$F[ax_1(n)+bx_2(n)]=aF[x_1(n)]+bF[x_2(n)]$$

Let the response of the system to $x_1(n)$ be $y_1(n)$ and the response of the system to $x_2(n)$ be $y_2(n)$. Thus for the input $x_1(n)$, the describing equation is,

$$y_1(n) = 3y_1^2(n-1) - nx_1(n) + 4x_1(n-1) -2x_1(n-1)$$

and for the input $x_2(n)$

$$y_2(n) = 3y_2^2(n-1) - nx_2(n) + x_2(n-1) -2x_2(n+1)$$

Multiplying these equations by a and b respectively and adding yields,

$$ay_1(n) + by_2(n) = 3ay_1^2(n-1) - anx_1(n) + 4ax_1(n-1) - 2x_1(n-1)$$
$$+ 3by_2^2(n-1) - bnx_2(n) + 4bx_2(n-1) - 2bx_2(n+1)$$

$$ay_1(n) + by_2(n) = 3\left(ay_1^2(n-1)+y_2^2(n-1)\right)-n\left(ax_1(n)+bx_2(n)\right)$$
$$+4\left(ax_1(n-1)+bx_2(n-1)\right)-2\left(ax_1(n+1)+bx_2(n+1)\right)$$

The superposition theorem is not satisfied.

Hence the system is non-linear.

(b) $y(n)=x(n+1)-3x(n)+x(n-1);\ n\geq 0$

Let the response of the system to the input $x_1(n)$ by $y_1(n)$ and the response of the system to the input $x_2(n)$ be $y_2(n)$.

Thus for the input $x_1(n)$, the difference equation is,

$$y_1(n)=x_1(n+1)-3x_1(n)+x_1(n-1)$$

and the difference equation for the input $x_2(n)$ is

$$y_2(n)=x_2(n+1)-3x_2(n)+x_2(n-1)$$

Multiplying the above equations by a and b respectively and adding yields,

$$ay_1(n) + by_2(n) = ax_1(n+1) - 3ax_1(n) + ax_1(n-1) + bx_2(n+1) - 3bx_2(n) + bx_2(n-1)$$
$$ay_1(n) + by_2(n) = [ax_1(n+1) + bx_2(n+1)] - 3[ax_1(n) + bx_2(n)] + [ax_1(n-1) + bx_2(n-1)]$$

Thus, the superposition condition is satisfied (i.e.) the response of the system to the input $ax_1(n)+bx_2(n)$ is $ay_1(n)+by_2(n)$.

Hence, the system is linear.

(ii) *Shift Invariance*

The necessary condition for time invariance is

$$y(n-k)=F[x(n-k)]$$

(a) Now, $F[x(n-k)]=3y^2(n-k-1)-nx(n-k)+4x(n-k-1)-2x(n-k+1)$

$$y(n-k)=3y^2(n-k-1)-(n-k)\,x(n-k)+4x(n-k-1)-2x(n-k+1)$$

As $y(n-k) \neq F[x(n-k)]$, the system is shift variant.

(b) $F[x(n-k)]=x(n-k+1)-3x(n-k)+x(n-k-1)$

$$y(n-k)=x(n-k+1)-3\,x(n-k)+x(n-k-1)$$

As $y(n-k)=F[x(n-k)]$, the system is time-invariant.

(iii) ***Causality***

The required condition for causality is that the output of a causal system must be dependent only on the present and past values of the input.

(a) From the given equation, it is clear that the output $y(n)$ is dependent on a future input sample value $x(n+1)$. Hence the system is non-causal.

(b) Since the output $y(n)$ of the given difference equation is dependent on a future input sample value $x(n+1)$, the system is non-causal.

Paley–Wiener Criterion

The modulus of the frequency response (or the amplitude response) is an even function of frequency, whereas the phase angle, i.e. the phase response is an odd function of frequency.

The Paley–Wiener Criterion gives the frequency-domain equivalence for the causality condition of the time domain. ***The Paley–Wiener Criterion states that a necessary and sufficient condition for an amplitude response to be realisable is that***

$$\int_{-\infty}^{\infty} \frac{|\ln|H(j\omega)||}{1+\omega^2}\,\mathrm{d}\omega < \infty$$

or equivalently,

$$\int_{-\infty}^{\infty} \frac{|\ln|H(f)||}{1+f^2}\,\mathrm{d}f < \infty$$

Requirements of amplitude response $|H(j\omega)|$:

(i) The magnitude $|H(j\omega)|$ of a realisable network can be zero at a discrete set of frequencies, but it cannot have zero magnitude over a finite band of frequencies; otherwise, Paley–Wiener criterion is invalid. That is, if $|H(j\omega)| = 0$ over any finite band, $||\ln|H(j\omega)|| = \infty$ over that band, and consequently $H(j\omega)$ is unrealisable.

(ii) According to the Paley–Wiener Criterion, the amplitude function $|H(j\omega)|$ must not fall to zero faster than the exponential order. Hence, any ideal filter is non-causal.

If the Paley–Wiener Criterion is satisfied, an exact synthesis can be accomplished theoretically with a finite time delay. If the Paley–Wiener Criterion is not satisfied, an exact synthesis requires an infinite time delay, but an approximation can be obtained with finite time delay.

Example 5.16 *Using Paley–Wiener Criterion, show that $|H(j\omega)| = e^{-\omega^2}$ is not a suitable amplitude response for a causal LTI system.*

Solution According to Paley–Wiener Criterion, the following condition should be satisfied for an amplitude response $|H(j\omega)|$ to be realisable.

$$\int_{-\infty}^{\infty} \frac{|\ln|H(j\omega)||}{1+\omega^2}\,\mathrm{d}\omega < \infty$$

Here, the given amplitude response is $|H(j\omega)| = e^{-\omega^2}$

Therefore,

$$\int_{-\infty}^{\infty} \frac{\left|\ln\left|e^{-\omega^2}\right|\right|}{1+\omega^2}\,\mathrm{d}\omega = \int_{-\infty}^{\infty} \frac{\omega^2}{1+\omega^2}\,\mathrm{d}\omega$$

$$= 2\int_0^{\infty} \frac{\omega^2}{1+\omega^2} d\omega \left[\text{since } \frac{\omega^2}{1+\omega^2} \text{ is even}\right]$$

$$= 2\int_0^{\infty} \left[1 - \frac{1}{1+\omega^2}\right] d\omega$$

$$= 2\left(\omega - \tan^{-1}\omega\right)_0^{\infty}$$

$$= 2\left[\left(\infty - \tan^{-1}\infty\right) - \left(0 - \tan^{-1}0\right)\right] = 2[\infty - \pi/2] = \infty$$

Since the Paley–Wiener Criterion is not satisfied, the amplitude function is not a suitable amplitude response for a causal LTI system.

Example 5.17 *Using Paley–Wiener Criterion, determine whether the magnitude function* $|H(j\omega)| = \frac{1}{\sqrt{1+\omega^2}}$ *is realisable.*

Solution According to Paley–Wiener Criterion, the magnitude function $|H(j\omega)|$ is realisable only when the following condition is satisfied.

$$\int_{-\infty}^{\infty} \frac{\left|\ln|H(j\omega)|\right|}{1+\omega^2} d\omega < \infty$$

Here, the given magnitude function is $|H(j\omega)| = \frac{1}{\sqrt{1+\omega^2}}$

$$\ln\left(\frac{1}{\sqrt{1+\omega^2}}\right) = \ln 1 - \log\left(1+\omega^2\right)^{1/2} = -\frac{1}{2}\log\left(1+\omega^2\right)$$

$$\left|\ln\left(\frac{1}{\sqrt{1+\omega^2}}\right)\right| = \frac{1}{2}\log\left(1+\omega^2\right)$$

Therefore,

$$\int_{-\infty}^{\infty} \frac{\left|\ln\left(\frac{1}{\sqrt{1+\omega^2}}\right)\right|}{1+\omega^2} d\omega = \int_{-\infty}^{\infty} \frac{\frac{1}{2}\log\left(1+\omega^2\right)}{1+\omega^2} d\omega$$

$$= 2\cdot\frac{1}{2}\int_0^{\infty} \frac{\log\left(1+\omega^2\right)}{1+\omega^2} d\omega$$

$$= \int_0^{\infty} \frac{\log\left(1+\omega^2\right)}{1+\omega^2} d\omega$$

Substituting $\omega = \tan\theta$, we get

$$1 + \omega^2 = 1 + \tan^2\theta = \sec^2\theta,$$

$$d\omega = \sec^2\theta\, d\theta \text{ and } \theta \text{ becomes 0 to } \pi/2$$

$$= \int_0^{\pi/2} \frac{\log \sec^2 \theta}{\sec^2 \theta} . \sec^2 \theta \, d\theta = 2 \int_0^{\pi/2} \log(\sec\theta)\, d\theta = 2 \int_0^{\pi/2} \log\left(\frac{1}{\cos\theta}\right) d\theta$$

$$= -2 \int_0^{\pi/2} \log\cos(\pi/2 - \theta)\, d\theta \left[\text{since} \int_0^a f(x)\,dx = \int_0^a f(a-x)\,dx \right]$$

$$= -2 \int_0^{\pi/2} \log\sin\theta \, d\theta$$

$$= -2\left(-\frac{\pi}{2} . \log 2\right) = \pi \log 2 = 0.945 \left[\text{since} \int_0^{\pi/2} \log\sin x \, dx = -\frac{\pi}{2} \log 2 \right]$$

Since the Paley–Wiener Criterion is satisfied, the given magnitude function is realisable.

5.2.4 Stability

The transfer function of the discrete time system is a rational function of z with real coefficients. For causal filters, the degree of the numerator polynomial is equal to or less than that of the denominator polynomial. The poles of the transfer function $H(z)$ determine whether the filter is stable or not, as in analog systems.

A necessary and sufficient condition for a DSP system to be stable is that the poles of the transfer function $H(z)$, $|p_i| < 1$ for $i = 1, 2, \ldots N$, i.e.

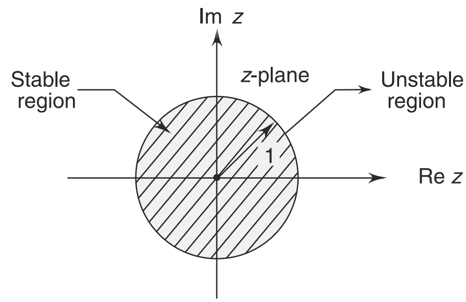


Fig. 5.6 *z-plane Regions for Stable and Unstable Systems*

$$\sum_{n=-\infty}^{\infty} |h(n)| < \infty \qquad (5.8)$$

The permissible regions for the location of poles for stable and unstable systems are illustrated in Fig. 5.6.

As the poles of non-recursive filters are located at the origin of the z-plane, these filters are always stable.

Example 5.18 *Check the stability of the filter for* $H(z) = \dfrac{z^2 - z + 1}{z^2 - z + \frac{1}{2}}$.

Solution $H(z) = \dfrac{z^2 - z + 1}{z^2 - z + \frac{1}{2}} = \dfrac{z^2 - z + 1}{(z - p_1)(z - p_2)}$

where the poles are at $z = p_1, p_2 = +\frac{1}{2} \pm j\frac{1}{2}$. Since $|p_1|, |p_2| < 1$, the filter is stable.

Example 5.19 (a) *Find the impulse response for the causal system* $y(n) - y(n-1) = x(n) + x(n-1)$
(b) *Find the response of the system to inputs* $x(n) = u(n)$ *and* $x(n) = 2^{-n}\,u(n)$. *Test its stability.*

Solution

(a) $y(n) - y(n-1) = x(n) + x(n-1)$

For the impulse response, $x(n) = \delta(n)$ and hence $y(n) = h(n)$

$$h(n) - h(n-1) = \delta(n) + \delta(n-1)$$

Taking z-transform, we get

$$H(z) - z^{-1} H(z) = 1 + z^{-1}$$

Therefore, $H(z) = \dfrac{1+z^{-1}}{1-z^{-1}} = \dfrac{1}{1-z^{-1}} + \dfrac{z^{-1}}{1-z^{-1}}$

Taking inverse z-transform, we get the impulse response $h(n) = u(n) + u(n-1)$.

(b) *Case* (*i*) Given $x(n) = u(n)$

Taking z-transform, we get $X(z) = \dfrac{z}{1-z}$, $\quad H(z) = \dfrac{1+z^{-1}}{1-z^{-1}}$

Therefore, $Y(z) = H(z)X(z) = \dfrac{z}{z-1}\dfrac{1+z^{-1}}{1-z^{-1}} = \dfrac{z(z+1)}{(z-1)^2}$

$$F(z) = \frac{Y(z)}{z} = \frac{z+1}{(z-1)^2} = \frac{A_1}{(z-1)} + \frac{A_2}{(z-1)^2}$$

where $\quad A_2 = (z-1)^2\, F(z)|_{z=1} = 2$

$$A_1 = \frac{d}{dz}[(z-1)^2\, F(z)]|_{z=1} = 1$$

Therefore, $\dfrac{Y(z)}{z} = \dfrac{1}{(z-1)} + \dfrac{2}{(z-1)^2}$

i.e, $\quad Y(z) = \dfrac{z}{(z-1)} + \dfrac{2z}{(z-1)^2}$

Taking inverse z-transform, we get

$$y(n) = u(n) + 2nu(n)$$

Here, since the pole is at $z = 1$, the system is unstable.

Case (*ii*) Given $x(n) = 2^{-n}\,u(n)$

Taking z-transform, we get $X(z) = \dfrac{z}{\left(z - \dfrac{1}{2}\right)}$

$$Y(z) = H(z)X(z) = \frac{1+z^{-1}}{1-z^{-1}}\,\frac{z}{\left(z-\dfrac{1}{2}\right)} = \frac{z(z+1)}{(z-1)\left(z-\dfrac{1}{2}\right)}$$

$$F(z)=\frac{y(z)}{z}=\frac{(z+1)}{(z-1)\left(z-\frac{1}{2}\right)}=\frac{A_1}{(z-1)}+\frac{A_2}{\left(z-\frac{1}{2}\right)}$$

$$A_1=F(z)(z-1)|_{z=1}=\left.\frac{(z+1)}{\left(z-\frac{1}{2}\right)}\right|_{z=1}=4$$

$$A_2=F(z)\left.\left(z-\frac{1}{2}\right)\right|_{z=\frac{1}{2}}=\left.\frac{(z+1)}{(z-1)}\right|_{z=\frac{1}{2}}=-3$$

Hence, $$\frac{Y(z)}{z}=\frac{4}{(z-1)}+\frac{-3}{\left(z-\frac{1}{2}\right)}$$

$$Y(z)=\frac{4z}{(z-1)}-\frac{3z}{\left(z-\frac{1}{2}\right)}$$

Taking inverse z-transform, we have

$$y(n)=4u(n)-3\left(\frac{1}{2}\right)^n u(n)$$

Here poles of $Y(z)$ are at $z=1$ and $z=\frac{1}{2}$, and hence the system is unstable.

Example 5.20 *Check the stability condition for the DSP systems described by the following equations:* (a) $y(n)=a^n u(n)$ (b) $y(n)=x(n)+e^a y(n-1)$

Solution

(a) $y(n)=a^n u(n)$

Taking z-transform, we have $Y(z)=\frac{1}{1-az^{-1}}=\frac{z}{z-a}$

Here the pole is at $z=a$ and hence for the system to be stable, $|a|<1$.

(b) $y(n)=x(n)+e^a y(n-1)$

Taking z-transform, we have

$$Y(z)=X(z)+e^a z^{-1} Y(z)$$
$$Y(z)\,[1-e^a z^{-1}]=X(z)$$

Therefore, $H(z)=\frac{Y(z)}{X(z)}=\frac{1}{1-e^a z^{-1}}=\frac{z}{z-e^a}$

Here the pole is at $z=e^a$ and hence $|e^a|<1$, i.e. $a<0$ for stability.

5.2.5 Bounded Input–Bounded Output(BIBO) Stability

Stability is of utmost importance in any system design. There are many definitions for stability. One of them is BIBO stability. A sequence $x(n)$ is bounded if there exists a finite M such that $|x(n)|<M$ for all n. Any system is said to be BIBO stable if and only if every bounded input gives a bounded output.

For any linear time invariant (LTI) system, the BIBO stability depends on the impulse response of that system. To obtain the necessary and sufficient condition for BIBO stability, consider the convolution property which relates the input and the output of an LTI system as

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

It follows that

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

$$\le M \sum_{k=-\infty}^{\infty} |h(k)| \left[\text{since } |x(n)| < M \text{ for all } n\right]$$

where M is a finite constant.

Therefore, the output is bounded if and only if $\sum_{k=-\infty}^{\infty} |h(k)|$ is bounded.

If the system is also causal, then the output is bounded if and only if, $\sum_{0}^{\infty} |h(k)|$ is bounded.
or

Since $h(n) = 0$ for $n < 0$, $\sum_{k=0}^{\infty} |h(k)| < \infty$

This is the sufficient condition for BIBO stability.

To show that it is also a necessary condition for BIBO stability, consider the input,

$$x(n-k) = \begin{cases} 1, & \text{if } h(k) > 0 \\ 0, & \text{if } h(k) = 0 \\ -1, & \text{if } h(k) < 0 \end{cases}$$

From this, $$|y(n)| = \left| \sum_{k=0}^{\infty} |h(k)| \right|$$

For any fixed value n, the summation is always non-negative. Thus the output will be unbounded if the sufficient condition is not satisfied. So this is also a necessary condition for BIBO stability.

Therefore, the necessary and sufficient condition for BIBO stability is

$$\sum_{k=0}^{\infty} |h(k)| < \infty \tag{5.9}$$

All non-recursive filters are stable since the poles lie only at the origin of the z-plane.

Example 5.21 *Digital filter is characterised by the difference equation,* $y(n) = x(n) + e^{\alpha} y(n-1)$. *Check the filter for BIBO stability.*

Solution $x(n) = x(n) + e^{\alpha} y(n-1)$

If $x(n) = \delta(n)$, then $y(n) = h(n)$

The impulse response of the digital filter is,

$$h(n) = \delta(n) + e^{\alpha} h(n-1)$$

When $n = 0$, $h(0) = \delta(0) + e^{\alpha} h(-1) = 1$

When $n = 1$, $h(1) = \delta(1) + e^{\alpha} h(0) = e^{\alpha}$

When $n=2,\ h(2)=\delta(2)+e^{\alpha}\,h(1)=e^{2\alpha}$

Similarly, $h(n)=e^{n\alpha}$

To check The BIBO stability, the condition is $\sum_{k=0}^{\infty}|h(k)|<\infty$.

Here, $$\sum_{k=0}^{\infty}|h(k)|=|\,1\,|+|\,e^{\alpha}\,|+|\,e^{2\alpha}\,|+\cdots+|e^{k\alpha}+\cdots=\sum_{k=0}^{\infty}|e^{k\alpha}|=\left|\frac{1}{1-e^{\alpha}}\right|$$

Therefore, the given system is BIBO stable only when $e^{\alpha}<1$ or $\alpha<0$.

Example 5.22 *Check whether the following digital systems are BIBO stable or not.*

(a) $y(n)=ax^2(n)$

(b) $y(n)=ax(n)+b$

(c) $y(n)=e^{-x(n)}$

(d) $y(n)=ax(n+1)+b\,x(n-1)$

(e) $y(n)=ax(n)\,.\,x(n-1)$

(f) $y(n)=\text{max. of }[x(n),\ x(n-1),\ x(n-2)]$

(g) $y(n)=\text{average of }[x(n+1),\ x(n),\ x(n-1)]$

(h) $y(n)=ax(n)+bx^2(n-1)$

Solution

(a) $y(n)=ax^2(n)$

If $x(n)=\delta(n)$, then $y(n)=h\,(n)$

Hence, the impulse response is $h\,(n)=a\delta^2(n)$

When $n=0,\ h\,(0)=a\delta^2(0)=a$

When $n=1,\ h\,(1)=a\delta^2(1)=0.$

In general,

$$h(n)=a,\ \text{when } n=0$$
$$=0,\ \text{when } n\neq 0$$

The necessary and sufficient condition for BIBO stability is

$$\sum_{k=0}^{\infty}|h(k)|<\infty$$

Here, $$\sum_{k=0}^{\infty}|h(k)|=|\ h(0)\ |+|\ h(1)|+|h(2)|+\cdots+|h(k)|+\cdots=|a|$$

Hence, the system is BIBO stable if $a<\infty$.

(b) $y(n)=ax(n)+b$

If $x(n)=\delta(n)$, then $y(n)=h\,(n)$.

Hence, the impulse response is $h\,(n)=a\delta(n)+b$

When $n=0,\ h(0)=a\delta(0)+b=a+b$

When $n=1,\ h(1)=a\delta(1)+b=b$

Here, $h\,(1)=h(2)=\cdots=h\,(k)=b$

Therefore,

$$h(n)=a+b,\quad \text{when } n=0$$
$$=b,\quad \text{when } n\neq 0$$

The necessary and sufficient condition for BIBO stability is

$$\sum_{k=0}^{\infty}|h(k)|<\infty$$

Therefore, $\sum_{k=0}^{\infty}|h(k)| = |\ h(0)\ | + |\ h(1)| + |h(2)| + \cdots + |h(k)| + \cdots$

$= |a+b| + |b| + |b| + \cdots + |b| + \cdots$

This series never converges since the ratio between the successive terms is one. Hence, the given system is BIBO unstable.

(c) $y(n) = e^{-x(n)}$.

If $\quad x(n) = \delta(n)$, then $y(n) = h\ (n)$.

Hence, the impulse response is $h(n) = e^{-\delta \leq n)}$

When $n = 0$, $h(0) = e^{-\delta(0)} = e^{-1}$

When $n = 1$, $h(1) = e^{-\delta(1)} = e^{0} = 1$

In general,

$$h(n) = e^{-1} \quad \text{when } n = 0$$
$$= 1 \quad \text{when } n \neq 0 .$$

The necessary and sufficient condition for BIBO stability is $\sum_{k=0}^{\infty}|h(k)| < \infty$.

Therefore, $\sum_{k=0}^{\infty}|h(k)| = |\ h(0)\ | + |\ h(1)| + |h(2)| + \cdots + |h(k)| + \cdots$

$= e^{-1} + 1 + 1 + 1 + \cdots + 1 + \cdots$

Since the given system never converges, it is BIBO unstable.

(d) $y(n) = ax(n+1) + bx(n-1)$.

If $\quad x(n) = \delta(n)$, then $y(n) = h\ (n)$.

Hence, the impluse response is $h\ (n) = a\delta(n+1) + b\delta(n-1)$.

When $n = 0$, $h\ (0) = a\delta(1) + b\delta(-1) = 0$

When $n = 1$, $h\ (1) = a\delta(2) + b\delta(0) = b$

When $n = 2$, $h\ (2) = a\delta(3) + b\delta(1) = 0$

In general,

$$h\ (n) = b, \quad \text{when } n = 1$$
$$= 0, \quad \text{otherwise}$$

The necessary and sufficient condition for BIBO stability is

$$\sum_{k=0}^{\infty}|h(k)| < \infty$$

Here, $\sum_{k=0}^{\infty}|h(k)| = |\ b\ |$

Hence, the given system is BIBO stable if $|b| < \infty$.

(e) $y(n) = ax(n)\ .\ x(n — 1)$

If $x(n) = \delta(n)$, then $y(n) = h(n)$.

The above equation can be changed into $h(n) = a\delta(n)\ .\ \delta(n — 1)$

When $n = 0$, $h\ (0) = a\delta(0)\ .\ \delta(-1) = 0$

When $n = 1$, $h\ (1) = a\delta(1)\ .\ \delta(0) = 0$

When $n = 2$, $h\ (2) = a\delta(2)\ .\ \delta(1) = 0$

The necessary and sufficient condition for BIBO stability is

$$\sum_{k=0}^{\infty}|h(k)| < \infty$$

Here, $\sum_{k=0}^{\infty}|h(k)|=0$

So the given system is BIBO stable .

(f) $y(n)$ = max. of $[x(n), x(n-1), x(n-2)]$.

If $x(n)=\delta(n)$, then $y(n)=h(n)$.

The above equation can be changed into

$h(n)=$max. of $[\delta(n), \delta(n-1), \delta(n-2)]$

When $n=0$, $h(0)=$max. of $[\delta(0), \delta(-1), \delta(-2)]=1$

When $n=1$, $h(1)=$max. of $[\delta(1), \delta(0), \delta(-1)]=1$

When $n=2$, $h(2)=$max. of $[\delta(2), \delta(1), \delta(0)]=1$

When $n=3$, $h(3)=$max. of $[\delta(3), \delta(2), \delta(1)]=0$

In general,

$$h(n)=\begin{cases}1, & \text{for } n=0,1,2\\ 0, & \text{otherwise (i.e., } n>2)\end{cases}$$

The necessary and sufficient condition for BIBO stability is

$$\sum_{k=0}^{\infty}|h(k)|<\infty$$

Here, $$\sum_{k=0}^{\infty}|h(k)|=|\ h(0)\ |+|\ h(1)|+|h(2)|+\cdots+|h(k)|+\cdots$$
$$=1+1+1+0+\cdots=3.$$

So, the given system is BIBO stable.

(g) $y(n)=$average of $[x(n+1), x(n), x(n-1)]$.

If $x(n)=\delta(n)$, then $y(n)=h(n)$.

The above equation can be changed into

$$h(n)=\text{average of } [\delta(n+1), \delta(n), \delta(n-1)]$$

When $n=0$, $h(0)=$average of $[\delta(1), \delta(0), \delta(-1)]=\frac{1}{3}$

When $n=1$, $h(1)=$average of $[\delta(2), \delta(1), \delta(0)]=\frac{1}{3}$

When $n=2$, $h(2)=$average of $[\delta(3), \delta(2), \delta(1)]=0$

In general,

$$h(n)=\begin{cases}\frac{1}{3}, & \text{for } n=0,1\\ 0, & \text{for } n>1\end{cases}$$

The necessary and sufficient condition for BIBO stability is

$$\sum_{k=0}^{\infty}|h(k)|<\infty$$

Hence, $$\sum_{k=0}^{\infty}|h(k)|=|\ h(0)\ |+|\ h(1)|+|h(2)|+\cdots+|h(k)|+\cdots$$
$$=\frac{1}{3}+\frac{1}{3}+0+\cdots\frac{2}{3}$$

Hence the given system is BIBO stable.

(h) $y(n) = ax(n) + bx^2(n-1)$

If $x(n) = \delta(n)$, then $y(n) = h(n)$.

The above equation can be changed into $h(n) = a\delta(n) + b\delta^2(n-1)$

When $n = 0$, $h(0) = a\delta(0) + b\delta^2(-1) = a$

When $n = 1$, $h(1) = a\delta(1) + b\delta^2(0) = b$

When $n = 2$, $h(2) = a\delta(2) + b\delta^2(1) = 0$

Hence, $$\sum_{k=0}^{\infty} |h(k)| = |h(0)| + |h(1)| + |h(2)| + \cdots + |h(k)| + \cdots$$
$$= |a| + |b| + 0 + 0$$

Hence, the given system is BIBO stable if $|a| + |b| < \infty$.

Example 5.23 *Check the BIBO stability for the impulse response of a digital system given by* $h(n) = a^n u(n)$

Solution $h(n) = a^n u(n) \Rightarrow h(k) = a^k u(k)$

$$\sum_{k=0}^{\infty} |h(k)| = |a|^k + |a^0| + |a^1| + |a^2| + \cdots + |a^k| + \cdots = \left|\frac{1}{1-a}\right|$$

Hence, the given system is stable if $a < 1$, i.e. a lies inside the unit circle of the complex plane.

Example 5.24 *Determine the range of values of parameters 'a' for the LTI system with impulse response*

$$h(n) = \begin{cases} a^n, & n \geq 0 \; n \text{ even} \\ 0, & \text{otherwise} \end{cases}$$

to be stable.

Solution $h(n) = a^n \Rightarrow h(k) = a^k$

The condition for stability is

$$\sum_{k=-\infty}^{\infty} |h(k)| < \infty$$

Therefore, $$\sum_{k=-\infty}^{\infty} |a^k| < \infty$$

Since $h(n) = 0$ for $n < 0$, the above condition can be written as

$$\sum_{k=0}^{\infty} |a^k| < \infty$$

Since k is even, the above summation can be written as $\sum_{k=0}^{\infty} |a^{2k}|$

Therefore, $$\sum_{k=0}^{\infty} \left|(a^2)^k\right| = \frac{1}{1-|a^2|}$$

If $|a^2| < 1$, then the above summation is finite and hence the system is stable.

DISCRETE CONVOLUTION 5.3

Convolution is a mathematical operation equivalent to Finite Impulse Response (FIR) filtering. Convolution is important in digital signal processing because convolving two sequences in the time domain is equivalent to multiplying the sequences in the frequency-domain. Convolution finds its application in processing signals especially analysing the output of a system. Consider the signals $x_1(n)$ and $x_2(n)$. The convolution of these two signals is given by

$$x_3(n) = x_1(n) * x_2(n) = \sum_{k=-\infty}^{\infty} x_1(k)x_2(n-k) \tag{5.10}$$

5.3.1 Representation of Discrete-Time Signals in Terms of Impulse

An arbitrary signal $x(n)$ can be resolved into a sum of unit sample sequences. The signal $x(n)$ shown in Fig. 5.7 can be expressed in terms of unit sample sequences as

$$\begin{aligned} x(n) &= \sum_{k=-2}^{\infty} x(k)\delta(n-k) \\ &= x(-2)\,\delta(n+2)+x(-1)\,\delta(n+1)+x(0)\,\delta(n)+x(1)\,\delta(n-1)+x(2)\,\delta(n-2)+x(3)\,\delta(n-3) \\ &= (1)\,\delta(n+2)+(-2)\,\delta(n+1)+(2)\,\delta(n)+(1)\,\delta(n-1)+(-1)\,\delta(n-2)+(1)\,\delta(n-3) \end{aligned}$$

Hence, any arbitrary signal can be represented in terms of the weighted sum of impulses.

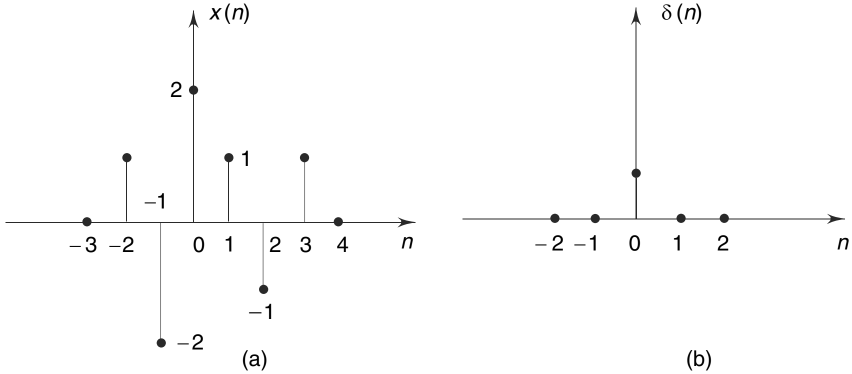


Fig. 5.7 *Representation of a Signal as Unit Sample Sequences*

5.3.2 Response of LTI Systems to Arbitrary Inputs (Convolution Sum)

Consider a linear system whose behaviour is specified by the impulse response $h(n)$. Let $x(n)$ be the input signal to this system and corresponding output signal be $y(n)$ as shown in Fig. 5.8.

To find the output signal $y(n)$, resolve the signal $x(n)$ into a weighted sum of impulses, and the linearity and time shift property of the LTI system can be used.

If $x(n) = \delta(n)$, then $y(n) = h(n) = H[\delta(n)]$

For any arbitrary signal $x(n)$,

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

where $x(k)$ is the weightage to the impulses at $n = k$.

The response of the system to the signal $x(n)$ is given by

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

Note: $H[\delta(n-k)] = h(n-k)$ by using the time invariance property.

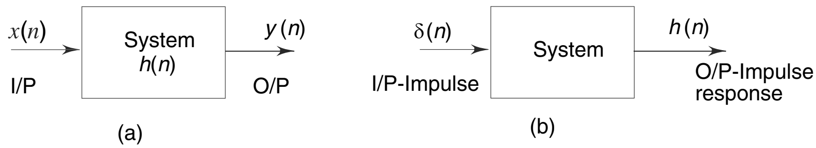


Fig. 5.8

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

The various steps involved in finding out the convolution sum are given below

1. **Folding:** Fold the signal $h(k)$ about the origin, i.e. at $k = 0$.
2. **Shifting:** Shift $h(-k)$ to the right by n_0 if n_0 is positive or shift $h(-k)$ to the left by n_0 if n_0 is negative to obtain $h(n_0 - k)$.
3. **Multiplication:** Multiply $x(k)$ by $h(n_0 - k)$ to obtain the product sequence $y_0(k) = x(k)\,h(n_0 - k)$
4. **Summation:** Sum all the values of the product sequence $y_0(k)$ to obtain the value of the output at time $n = n_0$.

5.3.3 Properties of Convolution

The convolution of two signals $x(n)$ and $h(n)$ is given by

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \quad \text{or} \quad y(n) = h(n) * x(n)$$
$$= \sum_{k=-\infty}^{\infty} h(k)\,x(n-k) \tag{5.11}$$

In the first case the impulse response $h(n)$ is folded and shifted, and $x(n)$ is the excitation signal. In the second case, the input signal $x(n)$ is folded and shifted. Here $h(n)$ acts as the excitation signal.

Commutative Law

Convolution satisfies commutative law, i.e. $x(n) * h(n) = h(n) * x(n)$, which is shown in Fig. 5.9.

Associative Law

$$[x(n) * h_1(n)] * h_2(n) = x(n) * [h_1(n) * h_2(n)]$$

Take LHS of the above equation.

Consider $x(n)$ to be the input signal to the LTI system with impulse response $h_1(n)$. The output $y_1(n)$ is given by

$$y_1(n) = x(n) * h_1(n)$$

This $y_1(n)$ signal now acts as the input signal to the second LTI system with impulse response $h_2(n)$.

Therefore,
$$y(n) = y_1(n) * h_2(n)$$
$$= [x(n) * h_1(n)] * h_2(n)$$

Now consider the RHS of the equation, which indicates that the input $x(n)$ is applied to an equivalent system $h(n)$ and is given by

$$h(n) = h_1(n) * h_2(n)$$

and the output of the equivalent system to the input $x(n)$ is given by

$$y(n) = x(n) * h(n)$$
$$= x(n) * [h_1(n) * h_2(n)]$$

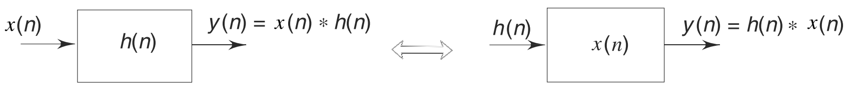


Fig. 5.9 *Commutative Property*

Since convolution satisfies commutative property, the cascading of two systems can be interchanged as shown in Fig. 5.10.

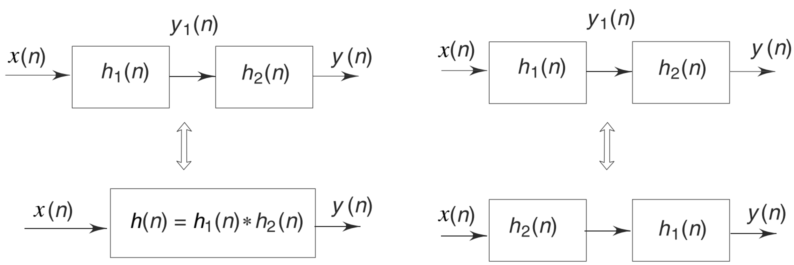


Fig. 5.10 *Associative Property*

If N linear time invariant systems are in cascade with impulse responses $h_1(n)$, $h_2(n)$,...$h_N(n)$, then the equivalent system impulse response is given by

$$h(n) = h_1(n) * h_2(n) * \cdots * h_N(n)$$

Distributive Law

$$x(n) * [h_1(n)+h_2(n)] = x(n) * h_1(n)+x(n) * h_2(n)$$

The distributive law states that if there are two LTI systems with impulse responses $h_1(n)$ and $h_2(n)$ and are excited by the same input $x(n)$, then the equivalent system has the impulse response as

$$h(n) = h_1(n)+h_2(n)$$

In general, if there are N number of LTI systems with impulse responses $h_1(n)$, $h_2(n)$,..., $h_N(n)$ excited by the same input $x(n)$, then the equivalent system impulse response is given by

$$h(n) = \sum_{l=1}^{N} h_l(n)$$

This is nothing but the parallel interconnection of individual systems.

Example 5.25 *Find the convolution of two finite duration sequences*

$$x(n) = \begin{cases} 1, & -1 \le n \le 1 \\ 0, & \text{otherwise} \end{cases} \quad \text{and} \quad h(n) = \begin{cases} 1, & -1 \le n \le 1 \\ 0, & \text{otherwise} \end{cases}$$

Solution The convolution of two finite duration sequences is given by

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

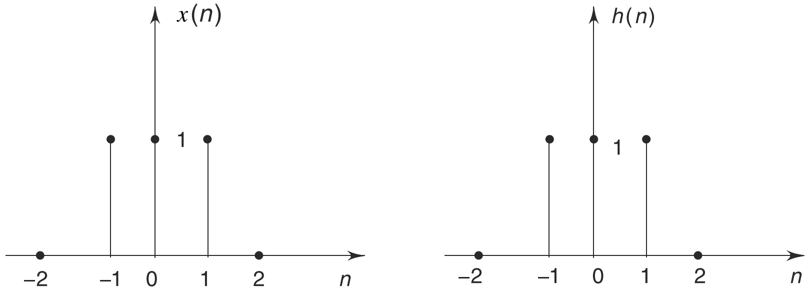


Fig. E5.25(a)

Step 1 Plot the given sequence, as shown in Fig. E5.25(a).

Step 2 To find the convolution sum $y(n)$

When $n=0$,

$$y(0)=\sum_{k=-\infty}^{\infty} x(k)h(-k)$$
$$=\cdots+x(-1)\,h(1)+x(0)\,h(0)+x(1)\,h(-1)+\cdots$$
$$=0+(1)\,(1)+(1)\,(1)+(1)\,(1)+0\,\cdots=3$$

When $n=1$,

$$y(1)=\sum_{k=-\infty}^{\infty} x(k)\,h\,(1-k)$$
$$=\cdots+x(-1)\,h\,(2)+x(0)\,h\,(1)+x(1)\,h\,(0)+\cdots$$
$$=0+(1)\,(1)+(1)\,(1)+0\,\cdots=2$$

When $n=2$,

$$y(2)=\sum_{k=-\infty}^{\infty} x(k)\,h\,(2-k)$$
$$=\cdots+x(-1)\,h\,(3)+x(0)\,h\,(2)+x(1)\,h\,(1)+\cdots$$
$$=0+(1)\,(1)+0\,\cdots=1$$

When $n=3$,

$$y(3)=\sum_{k=-\infty}^{\infty} x(k)\,h\,(3-k)=0$$

When $n=-1$,

$$y(-1)=\sum_{k=-\infty}^{\infty} x(k)\,h\,(-1-k)$$
$$=\cdots+x(-1)\,h\,(1)+x(0)\,h\,(-1)+x(1)\,h\,(-2)+\cdots$$
$$=0+(1)\,(1)+(1)\,(1)+0\,\cdots=2$$

When $n=-2$,

$$y(-2)=\sum_{k=-\infty}^{\infty} x(k)\,h\,(-2-k)$$
$$=\cdots+x(-1)\,h\,(-1)+x(0)\,h\,(-2)+x(1)\,h\,(-3)+\cdots$$
$$=0+(1)\,(1)+0\,\cdots=1$$

When $n = -3$

$$y(-3) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(-3-k) = 0$$

The convolution signal $y(n)$ is [See Fig. E5.25(b)]

$$y(n) = 0,\ n \leq -3 \text{ and } n \geq 3$$
$$y(n) = 1,\ n = \pm 2$$
$$y(n) = 2,\ n = \pm 1$$
$$y(n) = 3,\ n = 0$$

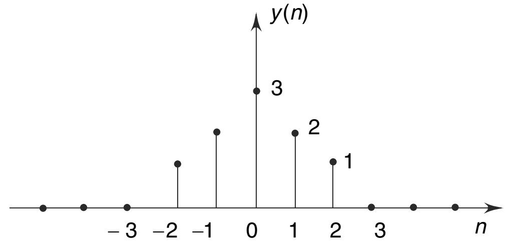


Fig. E5.25(b)

Note: For the convolved signal, the left extreme and the right extreme can be found using the left and right extremes of the two sequences to be convolved. That is,

$$y_l = x_l + h_l$$
$$y_r = x_r + h_r$$

where x_l, h_l and y_l are the left extremes of the signals x, h and y respectively. Similarly x_r, h_r and y_r are the right extremes of the signals x, h and y respectively.

Alternate (Graphical) Method

The given problem can be solved by using the graphical method as shown in Fig. E5.25(c). We know that $y(n) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(n-k)$

When $n = 0,\ y(0) = \sum_{k=-\infty}^{\infty} y_0(k) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(-k) = 3$

When $n = 1,\ y(1) = \sum_{k=-\infty}^{\infty} y_1(k) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(1-k) = 2$

When $n = 2,\ y(2) = \sum_{k=-\infty}^{\infty} y_2(k) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(2-k) = 1$

When $n = 3,\ y(3) = \sum_{k=-\infty}^{\infty} y_3(k) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(3-k) = 0$

When $n = -1,\ y(-1) = \sum_{k=-\infty}^{\infty} y_{-1}(k) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(-1-k) = 2$

When $n = -2,\ y(-2) = \sum_{k=-\infty}^{\infty} y_{-2}(k) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(-2-k) = 1$

When $n = -3,\ y(-3) = \sum_{k=-\infty}^{\infty} y_{-3}(k) = \sum_{k=-\infty}^{\infty} x(k)\, h\,(-3-k) = 0$

When these sequence values are plotted in Fig. E5.25(c), we find that the result is identical to the result shown in Fig. E5.25(b).

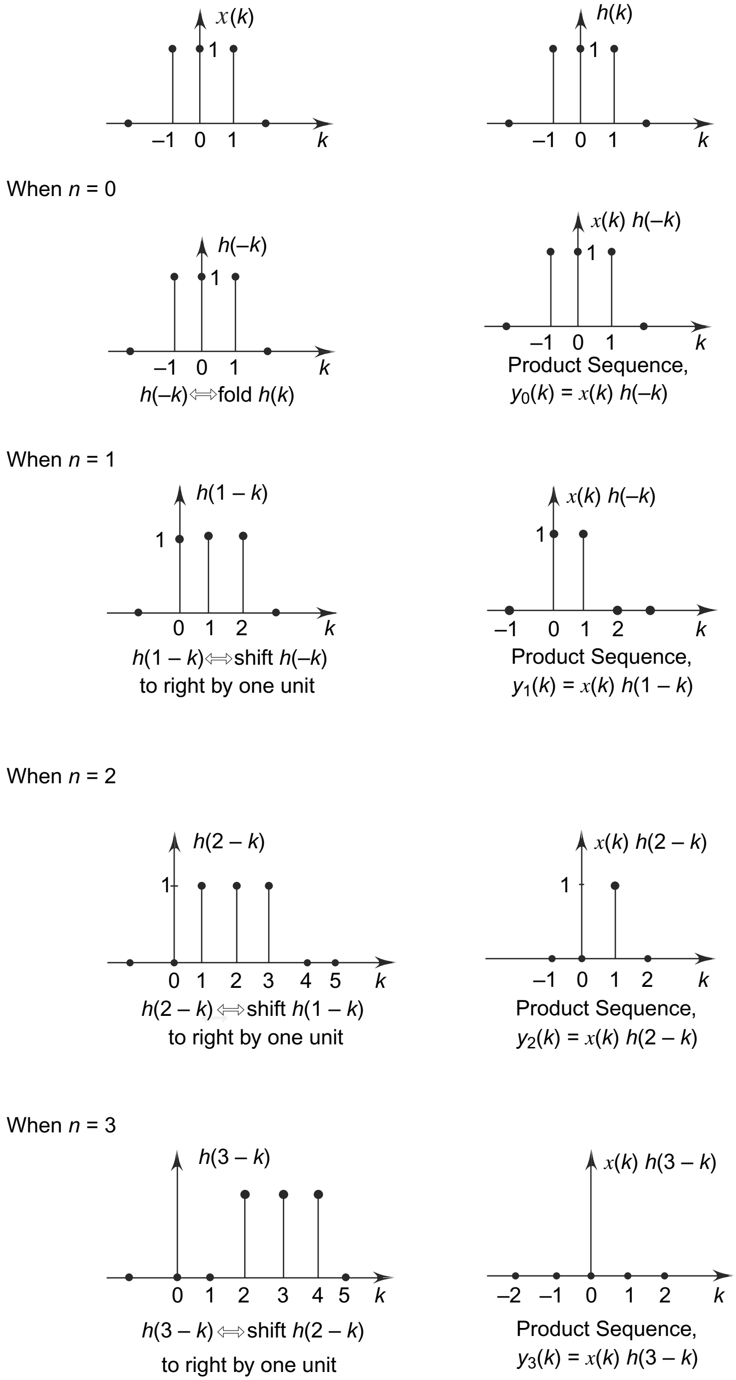
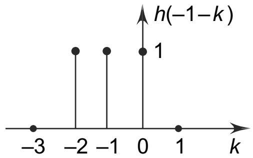
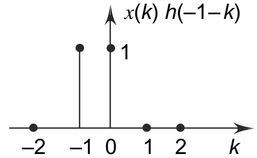


Fig. E5.25(c) *Graphical Method (Contd.)*

When $n = -1$

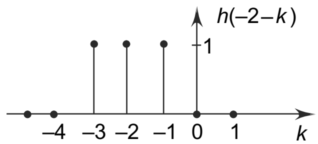


$h(-1-k)$ shift $h \Longleftrightarrow (-k)$ to left by one unit

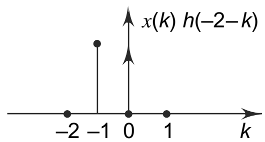


Product Sequence, $y_{-1}(k) = x(k)\, h(-1-k)$

When $n = -2$

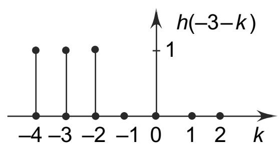


$h(-2-k)$ shift $h \Longleftrightarrow (-1-k)$ to left by one unit

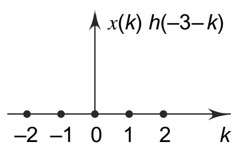


Product Sequence, $y_{-2}(k) = x(k)\, h(-2-k)$

When $n = -3$



$h(-3-k)$ shift $h \Longleftrightarrow (-2-k)$ to left by one unit



Product Sequence, $y_{-3}(k) = x(k)\, h(-3-k)$

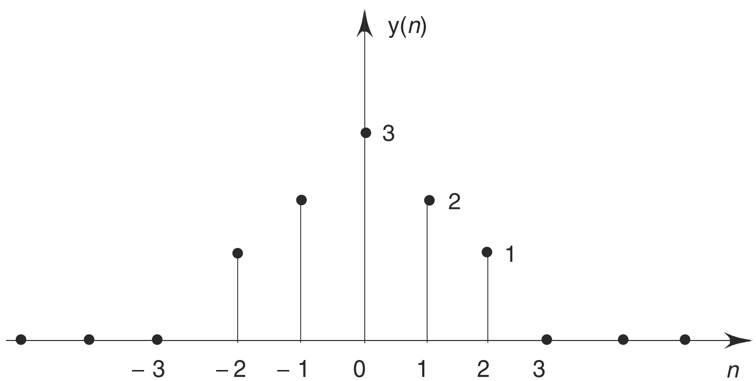


Fig. E5.25(c) *Graphical Method*

Example 5.26 *Find the convolution of the two signals $x(n) = u(n)$ and $h(n) = a^n\, u(n)$, ROC: $|a| < 1$; $n \geq 0$.*

Solution Plot the sequence values of the two signals as shown in Fig. E5.26(a).

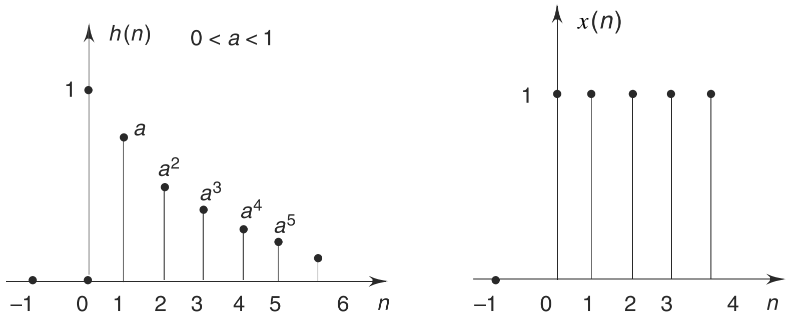


Fig. E5.26(a)

The convolution of the two signals is given by

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

When $n = 0$,

$$\begin{aligned} y(0) &= \sum_{k=-\infty}^{\infty} h(k)x(-k) \\ &= \cdots + h\,(0)\,x\,(0) + h\,(1)\,x\,(-1) + \cdots \\ &= 0 + 1 + 0 + \cdots = 1 \end{aligned}$$

When $n = 1$,

$$\begin{aligned} y(1) &= \sum_{k=-\infty}^{\infty} h(k)x(1-k) \\ &= \cdots + h(0)\,x(1) + h(1)\,x(0) + h(2)\,x(-1) + \cdots \\ &= 0 + (1)\,(1) + (a)\,(1) + 0\,\cdots = 1 + a \end{aligned}$$

When $n = 2$,

$$\begin{aligned} y(2) &= \sum_{k=-\infty}^{\infty} h(k)x(2-k) \\ &= \cdots + h(0)\,x(2) + h(1)\,x(1) + h(2)\,x(0) + h(3)\,x(-1) + \cdots \\ &= 0 + (1)\,(1) + (a)\,(1) + (a^2)\,(1) + 0 + \cdots = 1 + a + a^2 \end{aligned}$$

When $n = 3$,

$$\begin{aligned} y(3) &= \sum_{k=-\infty}^{\infty} h(k)x(3-k) \\ &= \cdots + h(0)\,x(3) + h(1)\,x(2) + h\,(2)\,x(1) + h(3)x(0) + h(4)\,x(-1) + \cdots \\ &= 0 + (1)\,(1) + (a)\,(1) + (a^2)\,(1) + (a^3)\,(1) + 0 + \cdots \\ &= 1 + a + a^2 + a^3 \end{aligned}$$

Generally, for $n > 0$,

$$\begin{aligned} y(n) &= 1 + a + a^2 + a^3 + \cdots + a^n \\ &= \frac{1 - a^{n+1}}{1 - a} \end{aligned}$$

For $n<0,\ y(n)=0$

When $n\to\infty,\ y(n)=1+a+a^2+a^3+\cdots+a^n$

$$=\sum_{n=0}^{\infty}a^n=\frac{1}{1-a},\quad |a|<1$$

These sequence values are plotted in Fig. E5.26(b).

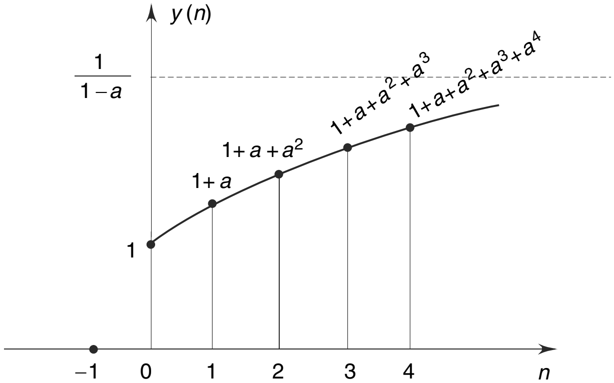
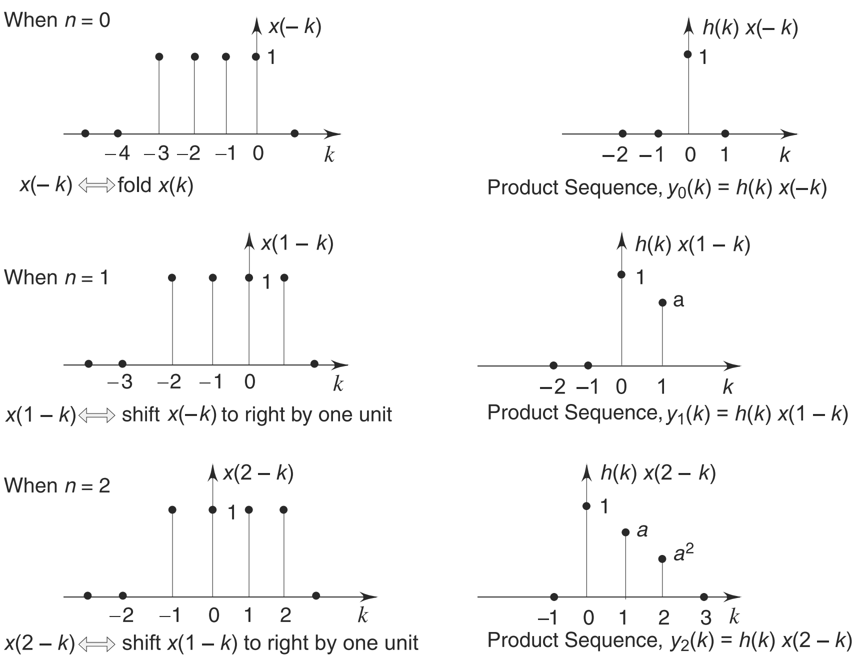


Fig. E5.26(b)

Graphical Convolution

The convolution of the given two signals can be determined by using the graphical method as shown in Fig. E5.26(c).



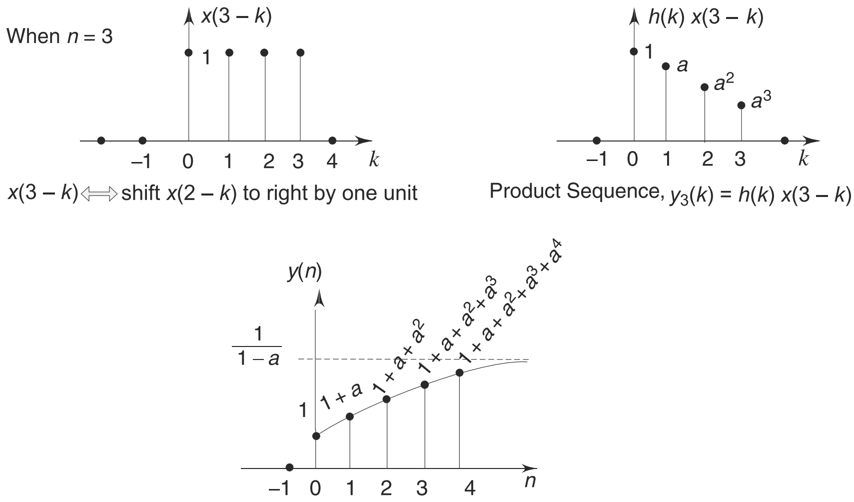


Fig. E5.26(c)

Example 5.27 *Find the convolution of the two signals shown in Fig. E5.27(a).*

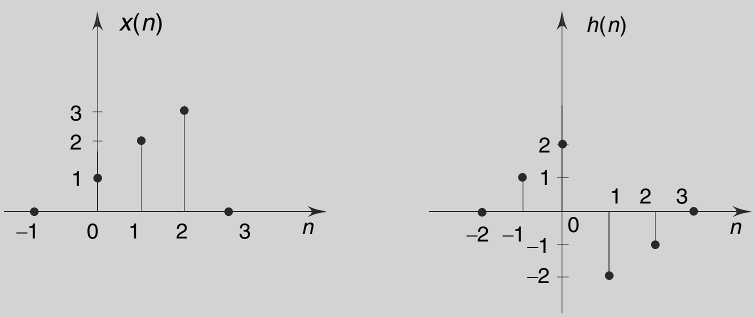


Fig. E5.27(a)

Solution Let $y(n)$ be the resultant convoluted signal. Then

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

From the graph,

$$x_l = 0,\ x_r = 2,\ h_l = -1,\ h_r = 2$$

Therefore, the left and right extremes of $y(n)$ are found to be

$$y_l = x_l + h_l = 0 + (-1) = -1$$
$$y_r = x_r + h_r = 2 + 2 = 4$$

When $n = -1$

$$y(-1) = \sum_{k=-\infty}^{\infty} x(k)\,h(-1-k)$$
$$= \cdots + x(0)\,h(-1) + x(1)\,h(-2) + x(1)\,h(-3) + \cdots$$
$$= 0 + (1)\,(1) + (2)\,(0) + 0 \cdots = 1$$

When $n=0$

$$y(0) = \sum_{k=-\infty}^{\infty} x(k)h(-k)$$
$$= \cdots + x(0)\,h\,(0) + x(1)\,h\,(-1) + x(2)\,h\,(-2) + \cdots$$
$$= 0 + (1)\,(2) + (2)\,(1) + 0 \cdots = 4$$

When $n=1$

$$y(1) = \sum_{k=-\infty}^{\infty} x(k)h(1-k)$$
$$= \cdots + x(0)\,h\,(1) + x(1)\,h\,(0) + x(2)\,h\,(-1) + x(3)\,h\,(-2) + \cdots$$
$$= 0 + (1)\,(-2) + (2)\,(2) + (3)\,(1) + \cdots = 5$$

When $n=2$

$$y(2) = \sum_{k=-\infty}^{\infty} x(k)h(2-k)$$
$$= \cdots + x(0)\,h\,(2) + x(1)\,h\,(1) + x(2)\,h\,(0) + \cdots$$
$$= 0 + (1)\,(-1) + (2)\,(-2) + 3(2) + \cdots = 1$$

When $n=3$

$$y(3) = \sum_{k=-\infty}^{\infty} x(k)h(3-k)$$
$$= \cdots + x(0)\,h\,(3) + x(1)\,h\,(2) + x(2)\,h\,(1) + \cdots$$
$$= 0 + (1)\,(0) + (2)\,(-1) + 3(-2) + 0 \cdots = -8$$

When $n=4$

$$y(4) = \sum_{k=-\infty}^{\infty} x(k)h(4-k)$$
$$= \cdots + x(0)\,h\,(4) + x(1)\,h\,(3) + x(2)\,h\,(2) + \cdots$$
$$= 0 + (1)\,(0) + (2)\,(0) + 3(-1) + 0 \cdots = -3$$

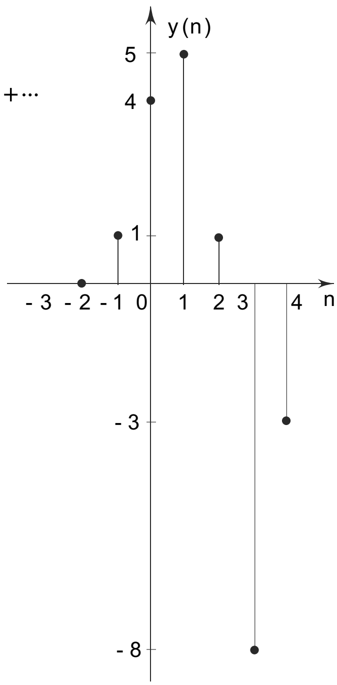


Fig. E5.27(b)

These sequence values are plotted in Fig. E5.27(b).

Example 5.28 *Compute the convolution $y(n) = x(n) * h(n)$ of the signals*

$$x(n) = \left\{\begin{matrix}1,1,0,1,1,\\ \uparrow\end{matrix}\right\} \quad \text{and} \quad h(n) = \left\{\begin{matrix}1,-2,-3,4\\ \qquad\quad\uparrow\end{matrix}\right\}$$

Solution The sequences of the given two signals are plotted in Fig. E5.28(a).
From the graph,

$$x_l = -2,\ x_r = 2,\ h_l = -3,\ h_r = 0$$

Hence the left and right extremes of the convoluted signal $y(n)$ are calculated as

$$y_l = x_l + h_l = -2 + (-3) = -5$$
$$y_r = x_r + h_r = 2 + 0 = 2$$

The convolution signal $y(n)$ is given as

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

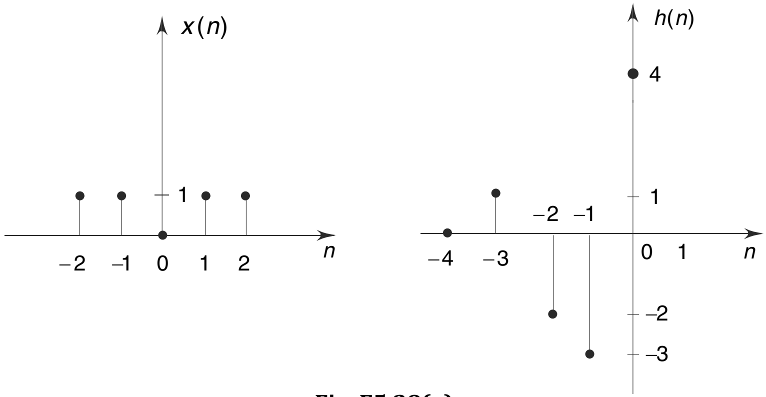


Fig. E5.28(a)

When $n=0$

$$y(0) = \sum_{k=-\infty}^{\infty} h(k)x(-k)$$
$$= \cdots + 0 + x(-2)\,h(2) + x(-1)\,h(1) + x(0)\,h(0) + x(1)\,h(-1) + x(2)\,h(-2) + 0 + \cdots$$
$$= 0 + (1)\,(0) + (1)\,(0) + (0)\,(4) + (1)\,(-3) + (1)\,(-2) + 0 + \cdots = -5$$

When $n=1$

$$y(1) = \sum_{k=-\infty}^{\infty} h(k)x(1-k)$$
$$= \cdots + x(-2)\,h(3) + x(-1)\,h(2) + x(0)\,h(1) + x(1)\,h(0) + x(2)\,h(-1) + \cdots$$
$$= 0 + (1)\,(0) + (1)\,(0) + (0)\,(0) + (1)\,(4) + (1)\,(-3) + 0 + \cdots = 1$$

When $n=2$

$$y(2) = \sum_{k=-\infty}^{\infty} h(k)x(2-k)$$
$$= \cdots + x(-2)\,h(4) + x(-1)\,h(3) + x(0)\,h(2) + x(1)\,h(1) + x(2)\,h(0) + \cdots$$
$$= 0 + (1)\,(0) + (1)\,(0) + (0)\,(0) + (1)\,(4) + (1)\,(0) + \cdots = 4$$

When $n=-1$

$$y(-1) = \sum_{k=-\infty}^{\infty} h(k)x(-1-k)$$
$$= \cdots + x(-2)\,h(1) + x(-1)\,h(0) + x(0)\,h(-1) + x(1)\,h(-2) + x(2)\,h(-3) + \cdots$$
$$= 0 + (1)\,(0) + (1)\,(4) + (0)\,(-3) + (1)\,(-2) + (1)\,(1) + 0 + \cdots = 3$$

When $n=-2$

$$y(-2) = \sum_{k=-\infty}^{\infty} h(k)x(-2-k)$$
$$= \cdots + x(-2)\,h(0) + x(-1)\,h(-1) + x(0)\,h(-2) + x(1)\,h(-3) + x(2)\,h(-4) + \cdots$$
$$= 0 + (1)\,(4) + (1)\,(-3) + (0)\,(-2) + (1)\,(1) + (1)\,(0) + 0 \cdots = 2$$

When $n=-3$

$$y(-3) = \sum_{k=-\infty}^{\infty} h(k)x(-3-k)$$
$$= \cdots + x(-2)\,h(-1) + x(-1)\,h(-2) + x(0)\,h(-3) + x(1)\,h(-4) + x(2)\,h(-5) + \cdots$$
$$= 0 + (1)\,(-3) + (1)\,(-2) + (0)\,(1) + (1)\,(0) + (1)\,(0) + 0 \cdots = -5$$

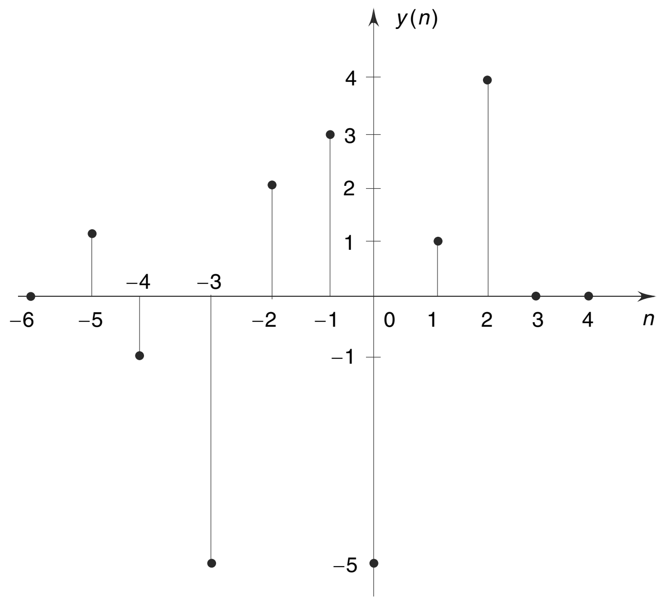


Fig. E5.28(b)

When $n=-4$

$$y(-4) = \sum_{k=-\infty}^{\infty} h(k)x(-4-k)$$
$$= \cdots + x(-2)\,h(-2) + x(-1)\,h(-3) + x(0)\,h(-4) + x(1)\,h(-5) + x(2)\,h(-6) + \cdots$$
$$= 0 + (1)\,(-2) + (1)\,(1) + (0)\,(0) + (1)\,(0) + (1)\,(0) + 0 + \cdots = -1$$

When $n=-5$

$$y(-5) = \sum_{k=-\infty}^{\infty} h(k)x(-5-k)$$
$$= \cdots + x(-2)\,h(-3) + x(-1)\,h(-4) + x(0)\,h(-5) + x(1)\,h(-6) + x(2)\,h(-7) + \cdots$$
$$= 0 + (1)\,(1) + 0 + \cdots = 1$$

These sequence values are plotted in Fig. E5.28(b).

5.4 SOLUTION OF LINEAR CONSTANT COEFFICIENT DIFFERENCE EQUATION

A discrete-time system transforms an input sequence $x(n)$ into an output sequence according to the recursion formula that represents the solution of a difference equation.

The general form of a difference equation is

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \tag{5.12}$$

where N is the order of the difference equation. The output at time n is a weighted sum of past inputs, the present input and past outputs. The solution of the difference equation consists of two parts, i.e.

$$y(n) = y_h(n) + y_p(n)$$

where the subscript h on $y(n)$ denotes the solution to the homogeneous difference equation and the subscript p on $y(n)$ denotes the particular solution to the difference equation.

The homogeneous solution is obtained by setting terms involving the input $x(n)$ to zero. Hence, from Eq. (5.12), we have

$$\sum_{k=0}^{N} a_k y(n-k) = 0 \tag{5.13}$$

where $a_0 = 1$

For solving the above equation, let us assume that

$$y_h(n) = \lambda^n \tag{5.14}$$

Substituting Eq. (5.14) into Eq. (5.13), we get

$$\sum_{k=0}^{N} a_k \lambda^{n-k} = 0 \qquad \text{or}$$

that is, $\lambda^{n-N}[\lambda^N + a_1\lambda^{N-1} + \cdots + a_{N-1}\lambda + a_N] = 0$

Therefore, $$\lambda^N + a_1\lambda^{N-1} + \cdots + a_{N-1}\lambda + a_N = 0 \tag{5.15}$$

Equation (5.15) is known as the *characteristic equation* which has N number of roots denoted as $\lambda_1, \lambda_2, \ldots, \lambda_N$

If $\lambda_1, \lambda_2, \ldots, \lambda_N$ are distinct, the complete homogeneous solution is of the form

$$y_n(n) = C_1\lambda_1^n + C_2\lambda_2^n + \cdots + C_N\lambda_N^n \tag{5.16}$$

The initial conditions given are used to determine the arbitrary constants $C_1, C_2, \ldots, C_N$ in the homogeneous solution.

The particular solution $y_p(n)$ is to satisfy the difference equation for the specific input signal $x(n)$, $n \geq 0$. In other words $y_p(n)$ is any solution satisfying

$$1 + \sum_{k=0}^{N} a_k y_p(n-k) = \sum_{k=0}^{M} b_k x(n-k) \tag{5.17a}$$

For solving Eq. (5.17), we assume that $y_p(n)$ depends on the input $x(n)$.

The general form of the particular solution for several inputs are shown in Table 5.1.

Combining both the homogeneous solution and particular solution, we obtain the complete solution as

$$y(n) = y_h(n) + y_p(n) \tag{5.17b}$$

Table 5.1 *General form of particular solution for several types of input*

Input Signal x(n)	*Particular Solution $y_p(n)$*
A (Step input)	K
AM^n	KM^n
An^M	$K_0 n^M + K_1 n^{M-1} \cdots + K_M$
$A^n n^M$	$A^n(K_0 n^M + K_1 n^{M-1} \cdots + K_M)$
$A\cos\omega_0 n$ $A\sin\omega_0 n$	$K_1\cos\omega_0 n + K_2\sin\omega_0 n$

Impulse Response

We know that the general form of difference equation is

$$y(n) = \sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k)$$

For the input $x(n) = \delta(n)$

$$\sum_{k=0}^{N} b_k x(n-k) = 0, \text{ for } n > M \tag{5.18}$$

Equation (5.15) can be written as

$$\sum_{k=0}^{N} a_k y(n-k) = 0, \text{ where } a_0 = 1 \tag{5.19}$$

The solution of Eq. (5.19) is known as the homogeneous solution. The particular solution is zero since $x(n) = 0$ for $n > 0$, that is

$$y_p(n) = 0 \tag{5.20}$$

Hence, we can obtain the impulse response by solving the homogeneous equation and imposing the initial conditions to determine the arbitrary constants.

Step Response

The step response $s(n)$ can be expressed in terms of the impulse response $h(n)$ of a discrete-time system using convolution sum as

$$s(n) = h(n) * u(n)$$

$$= \sum_{k=-\infty}^{\infty} h(k) u(n-k) \tag{5.21}$$

Here $u(n-k) = 0$ for $k > n$ and $u(n-k) = 1$ for $k \leq n$; we therefore have

$$s(n) = \sum_{k=0}^{n} h(k) \text{ for } k \leq n \tag{5.22}$$

Hence, the step response is the running sum of the impulse response.

Example 5.29 *Consider a causal and stable LTI system whose input $x(n)$ and output $y(n)$ are related through the second order difference equation.*

$$y(n) - \frac{1}{12} y(n-1) - \frac{1}{12} y(n-2) = x(n)$$

Determine the impulse response for the system.

Solution The given difference equation is

$$y(n) - \frac{1}{12} y(n-1) - \frac{1}{12} y(n-2) = x(n)$$

For an impulse response, the particular solution $y_p(n) = 0$

Therefore, $y(n) = y_h(n)$

Let $y_h(n) = \lambda^n$. Substituting this into the given difference equation, we get

$\lambda^n - \frac{1}{12} \lambda^{n-1} - \frac{1}{12} \lambda^{n-2} = 0$, (since $x(n) = 0$, for homogeneous solution) i.e. $\lambda^2 - \frac{1}{12}\lambda - \frac{1}{12} = 0.$

The roots of the above characteristic equation are

$$\lambda_1 = \frac{1}{3} \text{ and } \lambda_2 = -\frac{1}{4}$$

The homogeneous solution is $y_h(n) = C_1\left(\frac{1}{3}\right)^n + C_2\left(-\frac{1}{4}\right)^n$

Therefore, the complete solution becomes

$$y_n(n) = C_1\left(\frac{1}{3}\right)^n + C_2\left(-\frac{1}{4}\right)^n$$

For impulse response, $x(n) = \delta(n)$; i.e. $x(n) = \begin{cases} 1, & n = 0 \\ 0, & \text{otherwise} \end{cases}$

For $n=0$, the given difference equation becomes

$$y(0) - \frac{1}{12}y(-1) - \frac{1}{12}y(-2) = x(0) = 1$$

$$y(0) = 1$$

Similarly, for $n=1$,

$$y(1) - \frac{1}{12}y(0) - \frac{1}{12}y(-1) = x(1) = 0$$

$$y(1) - \frac{1}{12} = 0 \quad \text{or} \quad y(1) = \frac{1}{12}$$

For $n=0$, $y(0) = C_1 + C_2$

For $n = 1$, $y(1) = \frac{1}{3}C_1 - \frac{1}{4}C_2$

Then, we have $C_1 + C_2 = 1$

and $\frac{1}{3}C_1 - \frac{1}{4}C_2 = \frac{1}{12}$

Solving for C_1 and C_2, we get

$$C_1 = \frac{4}{7};\ C_2 = \frac{3}{7}$$

Therefore, the impulse response is

$$h(n) = y(n) = \frac{4}{7}\left(\frac{1}{3}\right)^n + \frac{3}{7}\left(-\frac{1}{4}\right)^n$$

Example 5.30 *Find the step response of the system*

$$y(n) - \frac{1}{12}y(n-1) - \frac{1}{12}y(n-2) = x(n)$$

Solution From Table 5.1, the particular solution for the step input is

$$y_p(n) = K$$

Given $y(n)-\frac{1}{12}y(n-1)-\frac{1}{12}y(n-2)=x(n)$

For $n>2$, substituting $y_p(n)=K$ and $x(n)=1$ into the above equation, we have

$$K-\frac{1}{12}K-\frac{1}{12}K=1$$

Therefore, $K=\frac{6}{5}$, i.e. $y_p(n)=\frac{6}{5}$

From the previous example, the homogeneous solution to the given difference equation is

$$y_h(n)=C_1\left(\frac{1}{3}\right)^n+C_2\left(-\frac{1}{4}\right)^n$$

We know that the complete solution is $y(n)=y_h(n)+y_p(n)$

Therefore, $y(n)=C_1\left(\frac{1}{3}\right)^n+C_2\left(\frac{1}{4}\right)^n+\frac{6}{5}$

For $n=0,\ y(0)=C_1+C_2+\frac{6}{5}$

For $n=1,\ y(1)=\frac{1}{3}C_1-\frac{1}{4}C_2+\frac{6}{5}$

From the given difference equation, assuming $y(-1)=y(-2)=0$, we have

$$y(0)=x(0)=1$$

and $y(1)-\frac{1}{12}y(0)-\frac{1}{12}y(-1)=x(1)=1$

i.e. $y(1)=\frac{13}{12}$

Therefore, $C_1+C_2+\frac{6}{5}=1$

and $\frac{1}{3}C_1-\frac{1}{4}C_2+\frac{6}{5}=\frac{13}{12}$

Solving for C_1 and C_2, we get

$$C_1=-\frac{2}{7};\ C_2=\frac{3}{35}$$

Therefore, the step response is

$$s(n)=y(n)=\left[-\frac{2}{7}\left(\frac{1}{3}\right)^n+\frac{3}{35}\left(-\frac{1}{4}\right)^n+\frac{6}{5}\right]u(n)$$

Example 5.31 *Solve the difference equation*

$$y(n)=x(n)+\frac{5}{6}y(n-1)-\frac{1}{6}y(n-2)$$

for the input sequence $x(n)=\begin{cases}3^n; & n\geq 0\\ 0; & n<0\end{cases}$. *Assume the initial condition* $y(n)=0$ *for* $n<0$.

Solution Given $y(n)=x(n)+\frac{5}{6}y(n-1)-\frac{1}{6}y(n-2)$

$$y(n) - \frac{5}{6}y(n-1) + \frac{1}{6}y(n-2) = x(n)$$

Let $y_h(n) = \lambda^n$.

Substituting this in the above equation, we obtain $\lambda^2 - \frac{5}{6}\lambda + \frac{1}{6} = 0$ (since $x(n) = 0$ for homogeneous solution).

The roots of this equation are

$$\lambda_1 = \frac{1}{2} \quad \text{and} \quad \lambda_2 = \frac{1}{3}$$

Therefore, the homogeneous solution is

$$y_h(n) = C_1\left(\frac{1}{2}\right)^n + C_2\left(\frac{1}{3}\right)^n; \; n \geq 0$$

For particular solution $y_p(n) = K3^n$ is the probable output sequence in response to the input sequence $x(n) = 3^n$.

Substituting $y_p(n)$ into the original equation, we have

$$K3^n - \frac{5}{6}K3^{n-1} + \frac{1}{6}K3^{n-2} = 3^n$$

$$3^n\left(K - \frac{5}{6}K3^{-1} + \frac{1}{6}K3^{-2}\right) = 3^n$$

$$K - \frac{5}{6}K3^{-1} + \frac{1}{6}K3^{-2} = 1$$

i.e. $K = \frac{27}{20}$

Therefore, the particular solution is $y_p(n) = \left[\frac{27}{20}\right][3]^n$

Hence, the complete solution is

$$y(n) = C_1\left(\frac{1}{2}\right)^n + C_2\left(\frac{1}{3}\right)^n + \left(\frac{27}{30}\right)(3)^n; n \geq 0$$

To find the constants C_1 and C_2, we use the initial conditions, i.e. $y(-1) = y(-2) = 0$. Substituting $y(-1) = 0$ in the above equation, we get

$$y(-1) = 0 = C_1\left(\frac{1}{2}\right)^{-1} + C_2\left(\frac{1}{3}\right)^{-1} + \left(\frac{27}{20}\right)(3)^{-1} = 2C_1 + 3C_2 + \frac{9}{20}$$

$$y(-2) = 0 = C_1\left(\frac{1}{2}\right)^{-2} + C_2\left(\frac{1}{3}\right)^{-2} + \left(\frac{27}{20}\right)(3)^{-2} = 4C_1 + 9C_2 + \frac{3}{20}$$

Solving for C_1 and C_2, we get $C_1 = -\frac{3}{5}$ and $C_2 = \frac{1}{4}$.

Therefore, the general solution is

$$y(n) = \left(-\frac{3}{5}\right)\left(\frac{1}{2}\right)^n + \frac{1}{4}\left(\frac{1}{3}\right)^n + \frac{27}{20}(3)^n; \quad \text{for} \quad n \geq 0$$

Example 5.32 *Find the response of the following difference equation*

$y(n) - 5\,y(n-1) + 6y(n-2) = x(n)$, for $x(n) = u(n)$

Solution Given, $y(n) - 5\,y(n-1) + 6y(n-2) = x(n)$, for $x(n) = u(n)$.

For a step input, $y_p(n) = k$.

For $n > 2$ where none of the terms in the difference equation vanish, we have

$k - 5k + 6k = 1$, i.e. $k = \frac{1}{2}$. Therefore, $y_p(n) = \frac{1}{2}$.

Let $y_h(n) = \lambda^n$. Substituting this into the given difference equation, we get

$$\lambda^n - 5\lambda^{n-1} + 6\lambda^{n-2} = 0 \text{ [since } x(n) = 0 \text{ for homogeneous solution]}$$

i.e. $\lambda^2 - 5\lambda + 6 = 0$

The roots of the characteristic equation are

$$\lambda_1 = 2 \text{ and } \lambda_2 = 3$$

The homogeneous solution is

$$y_h(n) = C_1(2)^n + C_2(3)^n$$

We know that

$$y(n) = y_h(n) + y_p(n)$$

$$= C_1(2)^n + C_2(3)^n + \frac{1}{2}$$

Therefore,

for $n = 0$, $y(0) = C_1 + C_2 + \frac{1}{2}$

for $n = 1$, $y(1) = 2C_1 + 3C_2 + \frac{1}{2}$

From the difference equation, we get

$$y(0) = 1 \text{ and } y(1) = 6$$

$$C_1 + C_2 + \frac{1}{2} = 1 \qquad \text{i.e.} \quad C_1 + C_2 = \frac{1}{2}$$

$$2C_1 + 3C_2 + \frac{1}{2} = 6 \qquad \text{i.e.} \quad 2C_1 + 3C_2 = \frac{11}{2}$$

Solving for C_1 and C_2, we get

$$C_1 = -4; \; C_2 = \frac{9}{2}$$

$$y(n) = -4(2)^n + \frac{9}{2}(3)^n + \frac{1}{2}$$

$$= \left[-4(2)^n + \frac{9}{2}(3)^n + \frac{1}{2}\right]u(n), \quad \text{for} \quad n \geq 0$$

FREQUENCY DOMAIN REPRESENTATION OF DISCRETE-TIME SIGNALS AND SYSTEMS 5.5

If an input signal of the form $e^{j\omega t}$ is applied to a relaxed linear continuous time-variant system, the resulting output will be of the form $H(\omega)e^{j\omega t}$, where $H(\omega)$, the system function, may be a complex function.

Similarly, in discrete time-invariant systems, if the input is of form $e^{j\omega n}$, the output is $H(\omega)e^{j\omega n}$. $H(e^{j\omega})$ is a function of ω, which denotes the frequency response of the system. A schematic representation of these input-output relationships is shown in Fig. 5.11. $H(e^{j\omega})$ is a complex function given by either

$$H(e^{j\omega}) = H_r(e^{j\omega}) + jH_i(e^{j\omega}) \tag{5.23}$$

or

$$H(e^{j\omega}) = \left|H_r(e^{j\omega})\right| e^{j\phi} \quad \text{where} \quad \phi = \tan^{-1}\frac{H_i(e^{j\omega})}{H_r(e^{j\omega})} \tag{5.24}$$

where $H_r(\cdot)$ and $H_i(\cdot)$ are real functions. Thus the input-output relation is

$$y(n) = H(e^{j\omega})\, e^{j\omega n} \tag{5.25}$$

For $T \neq 1$, the input-output relation becomes

$$y(nT) = H(e^{j\omega T})\, e^{j\omega T} \tag{5.26}$$

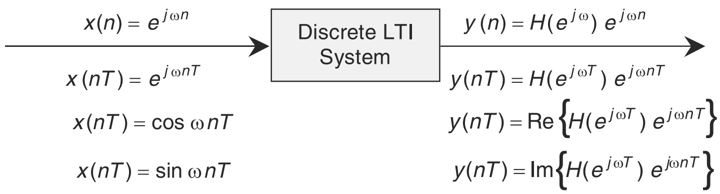


Fig. 5.11 *Schematic Representation of a Discrete-time System with an $e^{j\omega n}$ Input*

Example 5.33 *Find the output of a discrete LTI system for which the input is $x(n) = A\cos\omega_0 n$*

Solution The input is

$$x(n) = A\cos\omega_0 n$$
$$= \frac{A}{2}e^{j\omega_0 n} + \frac{A}{2}e^{-j\omega_0 n}$$

The output is

$$y(n) = \frac{A}{2}H(e^{j\omega_0})e^{j\omega_0 n} + \frac{A}{2}H(e^{-j\omega_0})e^{-j\omega_0 n}$$
$$= \frac{A}{2}H(e^{j\omega_0})e^{j\omega_0 n} + \frac{A}{2}\left[H(e^{-j\omega_0})e^{-j\omega_0 n}\right]^*$$
$$= A\left|H(e^{j\omega_0})\right|\cos(\omega_0 n + \phi)$$

where $$\phi = \tan^{-1}\frac{H_i(e^{j\omega_0})}{H_r(e^{j\omega_0})}$$

From this equation, we find that $H(e^{j\omega_0})$ is a continuous function of ω. $H(e^{j\omega_0})$ is a periodic function of ω_0 with period 2π and for a sinusoidal input to a linear discrete and time-invariant system, the output is sinusoidal, modified in amplitude and with a phase shift.

Example 5.34 *Find the frequency response of the system shown in Fig. E5.34(a) for an input $\upsilon(n) = e^{j\omega n}$.*

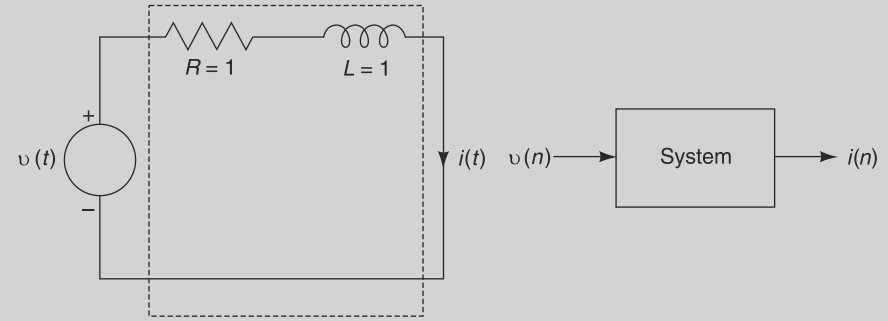


Fig. E5.34(a) *The System Function of an RL Circuit*

Solution Applying Kirchhoff's Voltage law, we get

$$L\frac{\mathrm{d}i(t)}{\mathrm{d}t}+Ri(t)=\upsilon(t)$$

Therefore, $$\frac{\mathrm{d}i(t)}{\mathrm{d}t}+\frac{R}{L}i(t)=\frac{1}{L}\upsilon(t)$$

The difference equation approximation of this equation is

$$\frac{i(nT)-i(nT-T)}{T}=\frac{1}{L}\upsilon(nT)-\frac{R}{L}i(nT)$$

Simplifying, we get

$$i(nT)=\frac{\frac{T}{L}}{1+\frac{R}{L}T}\upsilon(nT)+\frac{1}{1+\frac{R}{L}T}i(nT-T)$$

Assume $T = 1$ sec. Then this equation becomes

$$i(n)=\frac{\frac{1}{L}}{1+\frac{R}{L}}\upsilon(n)+\frac{1}{1+\frac{R}{L}}i(n-1)$$

Substituting $R = 1\,\Omega$ and $L = 1\,\mathrm{H}$ and $v(n) = e^{j\omega n}$, we get

$$2i(n) - i(n-1) = e^{j\omega n}$$

Therefore, $$2i(n) - e^{-j\omega}i(n) = e^{j\omega n}$$

$$i(n)=\frac{1}{2-e^{-j\omega}}e^{j\omega n}$$

To use the method of undetermined coefficients, we assume a solution of the form

$$i(n) = Ae^{j\omega n}$$

where A is to be determined. Equating the coefficients of the above two equations, we get

$$A=\frac{1}{2-e^{-j\omega}}$$

Therefore, the output function is

$$i(n)=\frac{1}{2-e^{-j\omega}}e^{j\omega n}$$

Since $i(n)=H(e^{j\omega})\,v(n)=H(e^{j\omega})e^{j\omega n}$ then the system function is

$$H\left(e^{j\omega}\right)=\frac{1}{2-e^{-j\omega}}=\frac{1}{(2-\cos\omega)+j\sin\omega}$$

Therefore, $$\left|H\left(e^{j\omega}\right)\right|=\frac{1}{\sqrt{(2-\cos\omega)^2+\sin^2\omega}}$$

$$\angle H(e^{j\omega})=-\tan^{-1}\left(\frac{\sin\omega}{2-\cos\omega}\right)$$

The magnitude function $|H(e^{j\omega n})|$ shown in Fig. E5.34(b), is *periodic* with period $2\pi(=360^\circ)$ and *symmetric*, that is $|H(e^{j\omega})|=|H(e^{-j\omega})|$. The phase response of the system function shown in Fig. E5.34(c), shows that $H(e^{j\omega})$ is *periodic* with period 2π, and it possesses *odd symmetry* with respect to the origin, that is $H(e^{-j\omega})=-H(e^{j\omega})$. The points $\pm\pi$ are called the *fold-over frequencies* of the frequency response.

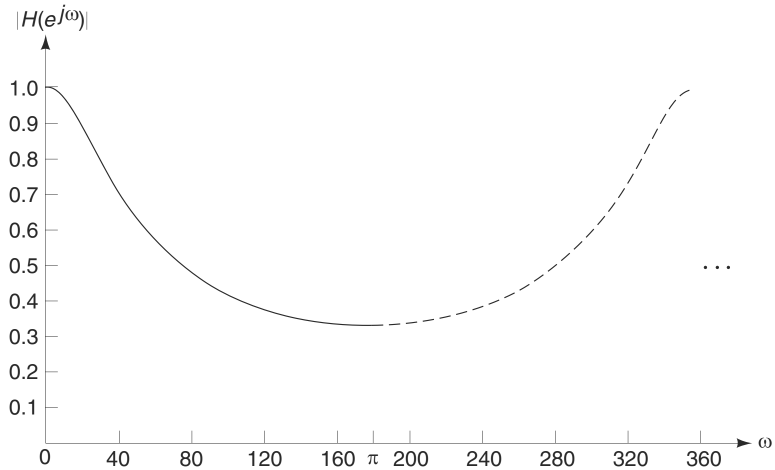


Fig. E5.34(b) *The Magnitude Response*

System Function of a Digital System in Frequency Domain

The k^{th} order of a discrete system is characterised by difference equation of the form

$$y(n)+a_1y(n-1)+a_2y(n-2)+\cdots+a_ky(n-k)$$
$$=b_0x(n)+b_1x(n-1)+\cdots+b_mx(n-m)$$

where the coefficients a_i and b_i are assumed to be constant. Since the particular solution to any linear discrete and time-invariant system is $y(n)=H(e^{j\omega})\,e^{j\omega n}$ when the input is $e^{j\omega n}$, the above equation becomes

$$H(e^{j\omega})e^{j\omega n}+a_1H(e^{j\omega})e^{j\omega(n-1)}+\cdots+a_kH(e^{j\omega})e^{j\omega(n-k)}$$
$$=b_0e^{j\omega n}+b_1e^{j\omega(n-1)}+\cdots+b_me^{j\omega(n-m)}$$

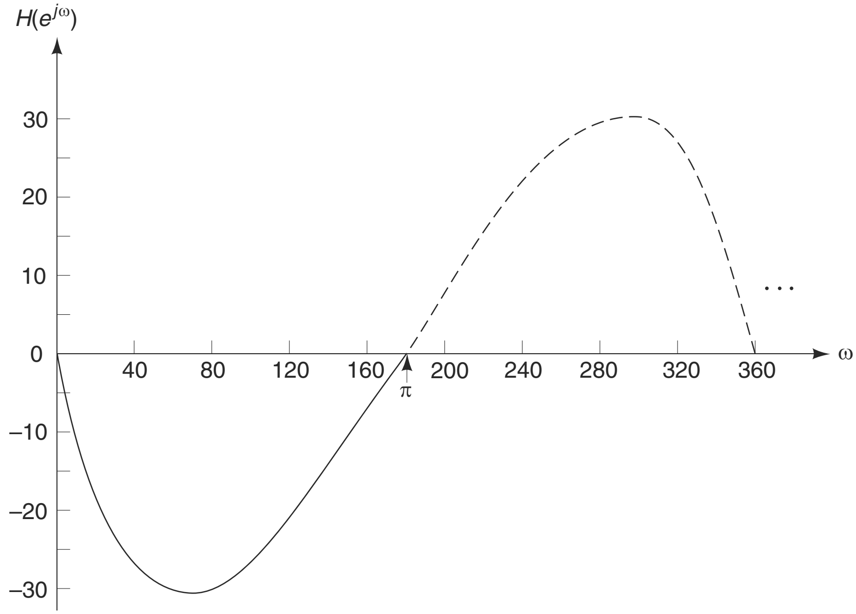


Fig. E5.34(c) *The Phase Response*

Simplifying, we get

$$= \frac{H_0 \prod_{i=1}^{P} \{\text{Vector magnitude from the } i^{\text{th}} \text{ zero to the frequency point on the circumference of the unit circle}\}}{\prod_{i=1}^{q} \{\text{Vector magnitude from the } i^{\text{th}} \text{ pole to the frequency point on the circumference of the unit circle}\}}$$

Using this formula, we can find the transfer function of a given system without finding its particular solution, as given in Example 5.35.

Example 5.35 *Find the transfer function for the system shown in Fig. E5.35.*

Solution The difference equation for the system shown in Fig. E5.35 is

$$y(n) = -3y(n-1) - 4y(n-2) + x(n) + x(n-1)$$

or

$$y(n) + 3y(n-1) + 4y(n-2) = x(n) + x(n-1)$$

Here, $a_1 = 3$, $a_2 = 4$, $b_0 = 1$. Hence, the transfer function for the system is

$$H(e^{j\omega}) = \frac{1 + e^{-j\omega}}{1 + 3e^{-j\omega} + 4e^{-j2\omega}}.$$

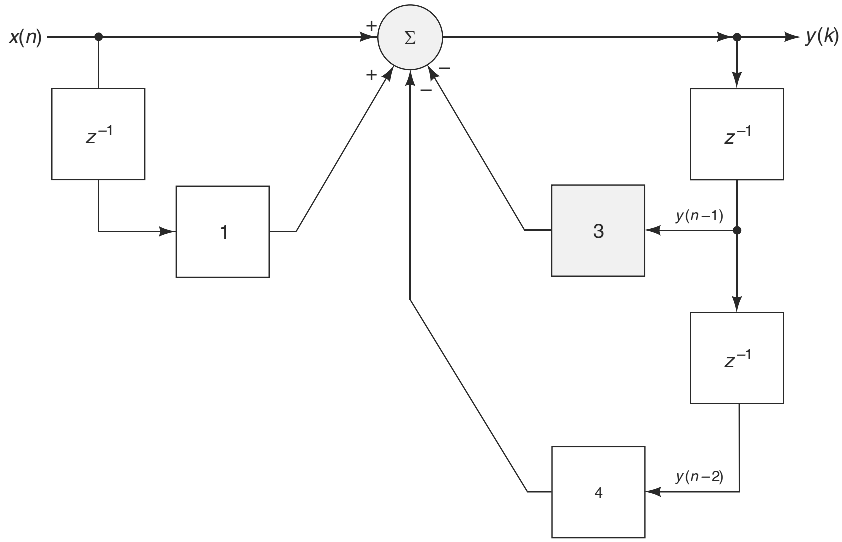


Fig. E5.35 *Block Diagram Representation of Discrete System*

DIFFERENCE EQUATION AND ITS RELATIONSHIP WITH SYSTEM FUNCTION, IMPULSE RESPONSE AND FREQUENCY RESPONSE 5.6

In general, a causal LTI system may be characterized by a linear constant coefficient difference equation given by

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_x x(n-k) \tag{5.27}$$

where terms a_k represent recursive, and b_k non-recursive, multiplier coefficients. The corresponding system function, impulse response and frequency response are determined as given below.

Taking z-transform of both sides of Eq. (5.27) using the linearity property, we obtain

$$\sum_{k=0}^{N} a_k Z[y(n-k)] = \sum_{k=0}^{M} b_x Z[x(n-k)]$$

Therefore, the system function $H(z)$ is

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} \tag{5.28}$$

Taking inverse z-transform of the system function $H(z)$ given in Eq. (5.28), we obtain the unit sample response, $h(n)$,

$$h(n) = Z^{-1}[H(z)] \tag{5.29}$$

We know that the output $Y(z) = H(z) \cdot X(z)$

If $x(n)$ is a unit sample response or impulse response $X(z) = 1$ and hence $Y(z) = H(z)$. Therefore, $y(n) = Z^{-1}[H(z)]$.

If $H(z)$ is known, a difference equation characterization can be determined by reversing the above operations. First $H(z)$ is expanded in terms of negative powers of z and made equal to $Y(z) / X(z)$. Then cross multiplying given Eq. (5.28) and the inverse transform gives the difference equation shown in Eq. (5.27).

Taking Fourier transform of the unit sample response $h(n)$, we obtain the frequency response given by

$$H(e^{j\omega}) = \sum_{k=-\infty}^{\infty} h(n)e^{-j\omega n} \tag{5.30}$$

$$= H(z)\big|_{z=e^{j\omega}}$$

z-Transform Solutions of Linear Difference Equations

The z-transform can be used to determine the solution $y(n)$, for $n \geq 0$, of the linear, time-invariant, constant coefficient difference equation that characterises a causal discrete-time system. For finding the solution of an Mth order difference equation, M initial conditions and the input signal $x(n)$ must be known. Using the time delay property of the z-transform, the initial conditions $y(-1)$, $y(-2)$… $y(-M)$ are incorporated directly in the solution obtained with z-transforms, as given below.

$$Z[y(n-1)] = z^{-1}\,Y(z) + y(-1) \tag{5.31a}$$

$$Z[y(n-2)] = z^{-2}\,Y(z) + z^{-1}\,y(-1) + y(-2) \tag{5.31b}$$

$$Z[y(n-3)] = z^{-3}Y(z) + z^{-2}\,y(-1) + z^{-1}\,y(-2) + y(-3) \tag{5.31c}$$

Here, the negative powers of z get decreased and negative arguments of $y(n)$ are increased. The initial conditions are not required for the input signal when the z-transform of the delayed input signals $x(n-k)$ in the difference equation are computed. The time-delay theorem applies to the input signal in which the input signal is equal to zero for $n < 0$ when signal-sided z-transforms are used.

Infinite Impulse Response (IIR) System

An LTI system is said to be an ***infinite impulse response*** (IIR) system if its unit sample response $h(n)$ is of infinite duration. ***A recursive filter*** that involves ***feedback*** has an impulse response which theoretically continues for ever, and hence it is referred to as an IIR filter.

Finite Impulse Response (IIR) System

An LTI system is said to be a ***finite impulse response*** (FIR) system if its unit sample response $h(n)$ is of infinite duration. Since the number of coefficients must be finite, a practical ***non-recursive filter*** is referred to as FIR filter.

The difference equation given in Eq. (5.27) represents an FIR system if $a_0 \neq 0$ and $a_k = 0$ for $k = 1, 2, \ldots, N$. Otherwise, it represents either an IIR or FIR system.

Substituting $a_k = 0$ for $k = 1, 2, \ldots, N$ in Eq. (5.27) gives

$$a_0 y(n-0) = \sum_{k=0}^{M} b_k x(n-k)$$

Rearranging, we get

$$y(n) = \sum_{k=0}^{M} \left(\frac{b_k}{a_0}\right) x(n-k) \tag{5.32}$$

From this equation, it is clear that the present output sample value depends only on present and previous inputs.

Comparing this equation to that of convolution, we can recognise b_k / a_0 as $h(k)$, the value of the unit sample response $h(n)$ at time k. Hence $h(n)$ is given by

$$h(n) = \begin{cases} b_k / a_0, & 0 \le n \le m \\ 0, & \text{otherwise} \end{cases} \tag{5.33}$$

As this is obviously of finite duration, it represents an FIR system.

Example 5.36 *A DSP system is described by the linear difference equation*

$$y(n) = 0.2\,x(n) - 0.5\,x(n-2) + 0.4\,x(n-3)$$

Given that the digital input sequence $\{-1, 1, 0, -1\}$ *is applied to this DSP system, determine the corresponding digital output sequence.*

Solution Taking z-transform of the given linear difference equation, we get

$$Y(z) = 0.2\,X(z) - 0.5z^{-2}X(z) + 0.4\,z^{-3}X(z)$$

Therefore,

$$H(z) = \frac{Y(z)}{X(z)} = 0.2 - 0.5z^{-2} + 0.4z^{-3}$$

The given input sequence is $x(n) = \{-1, 1, 0, -1\}$ and its z-transform is

$$X(z) = -1 + z^{-1} - z^{-3}$$

Therefore, $Y(z) = H(z) \cdot X(z)$

$$= -0.2 + 0.2z^{-1} + 0.5z^{-2} - 1.1z^{-3} + 0.4z^{-4} + 0.5z^{-5} - 0.4z^{-6}$$

Taking inverse z-transform, we get the digital output sequence

$$y(n) = \{-0.2, 0.2, 0.5, -1.1, 0.4, 0.5, -0.4\}$$

Example 5.37 *Determine H (z) and its poles and zeros if*

$$y(n) + \frac{3}{4}y(n-1) + \frac{1}{8}y(n-2) = x(n) + x(n-1)$$

Solution Given $y(n) + \frac{3}{4}y(n-1) + \frac{1}{8}y(n-2) = x(n) + x(n-1)$

Taking z-transform, we get

$$Y(z) + \frac{3}{4}z^{-1}Y(z) + \frac{1}{8}z^{-2}y(z) = X(z) + z^{-1}X(z)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + z^{-1}}{1 + \frac{3}{4}z^{-1} + 1 + \frac{1}{8}z^{-2}} = \frac{z(z+1)}{z^2 + \frac{3}{4}z + \frac{1}{8}} = \frac{z(z+1)}{\left(z + \frac{1}{2}\right)\left(z + \frac{1}{4}\right)}$$

The poles are at $z = -\frac{1}{2}, -\frac{1}{4}$.

The zeros are at $z = 0, -1$.

Example 5.38 *Determine the impulse response for the systems given by the following difference equations.*

(a) $y(n)+3y(n-1)+2y(n-2)=2x(n)-x(n-1)$

(b) $y(n)=x(n)+3x(n-1)-4x(n-2)+2x(n-3)$

Solution

(a) For the impulse response if $x(n)=\delta(n)$, then $y(n)=h(n)$

$$h(n)+3h(n-1)+2h(n-2)=2\delta(n)-\delta(n-1)$$

Taking z-transform, we get

$$H(z)[1+3z^{-1}+2z^{-2}]=2-z^{-1}$$

Therefore, $H(z)=\dfrac{2-z^{-1}}{1+3z^{-1}+2z^{-2}}=\dfrac{2-z^{-1}}{(1+z^{-1})(1+2z^{-1})}=\dfrac{A_1}{1+z^{-1}}+\dfrac{A_2}{1+2z^{-1}}$

$$A_1=\frac{2-z^{-1}}{1+2z^{-1}}\bigg|_{z^{-1}=-1}=-3$$

$$A_2=\frac{2-z^{-1}}{1+z^{-1}}\bigg|_{z^{-1}=-\frac{1}{2}}=5$$

Therefore, $H(z)=\dfrac{-3}{1+z^{-1}}+\dfrac{5}{1+2z^{-1}}$

Taking inverse z-transform, we obtain

$$h(n)=[-3(-1)^n+5(-2)^n]u(n)$$

(b) The impulse response for the given difference equation is

$$h(n)=\delta(n)+3\delta(n-1)-4\delta(n-2)+2\delta(n-3)$$

Example 5.39 *Determine the impulse response and unit step response of the system described by the difference equation.*

(i) $y(n)=0.6y(n-1)-0.08y(n-2)+x(n)$

(ii) $y(n)=0.7y(n-1)-0.1y(n-2)+2x(n)-x(n-2)$

Solution (i) The given difference equation of the system is

$$y(n)=0.6\,y(n-1)-0.08\,y(n-2)+x(n)$$

i.e., $y(n)-0.6\,y(n-1)+0.08\,y(n-2)=x(n)$

To find the impulse response h(n)

For an impulse input, $x(n)=\delta(n)$. Hence, $y(n)=h(n)$

Therefore, $h(n)-0.6\,h(n-1)+0.08\,h(n-2)=\delta(n)$

Taking z-transform, we get $H(z)[1-0.6z^{-1}+0.08z^{-2}]=1$

Hence

$$H(z)=\frac{1}{(1-0.6z^{-1}+0.08z^{-2})}=\frac{1}{(1-0.4z^{-1}+1-0.02z^{-1})}$$

$$=\frac{A_1}{(1-0.4z^{-1})}+\frac{A_2}{(1-0.2z^{-1})}=\frac{2}{(1-0.4z^{-1})}-\frac{1}{(1-0.2z^{-1})}$$

Taking inverse z-transform, we get the impulse response as

$$h(n) = 2(0.4)^n u(n) - (0.2)^n u(n)$$

To find the unit step response s(n)

For the unit step input, $x(n)=u(n)$

Therefore, the difference equation becomes

$$y(n) - 0.6\, y(n-1) + 0.08\, y(n-2) = x(n)$$

Taking z-transform, we get

$$Y(z)\left[1-0.6z^{-1}+0.08z^{-2}\right]=\frac{1}{1-z^{-1}}$$

$$Y(z)=\frac{1}{\left(1-z^{-1}\right)\left(1-0.6z^{-1}+0.08z^{-2}\right)}$$

$$=\frac{1}{\left(1-z^{-1}\right)\left(1-0.4z^{-1}\right)\left(1-0.2z^{-1}\right)}$$

$$=\frac{A_1}{\left(1-z^{-1}\right)}+\frac{A_2}{\left(1-0.4z^{-1}\right)}+\frac{A_3}{\left(1-0.2z^{-1}\right)}$$

$$=\frac{25}{12}\frac{1}{\left(1-z^{-1}\right)}-\frac{4}{3}\frac{1}{\left(1-0.4z^{-1}\right)}+\frac{1}{4}\frac{1}{\left(1-0.2z^{-1}\right)}$$

Taking inverse z-transform, we get the unit step response as

$$s(n)=y(n)=\frac{25}{12}u(n)-\frac{4}{3}(0.4)^n u(n)+\frac{1}{4}(0.2)^n u(n)$$

Alternate Method

To find step response from the impulse response using convolution

We know that the step response $s(n)$ is the running sum of the impulse response $h(n)$. That is,

$$s(n)=h(n)*u(n)=\sum_{k=-\infty}^{\infty}h(k)u(n-k)$$

Here, $u(n-k)=0$ for $k>n$ and $u(n-k)=1$ for $k\le n$.

Hence, $s(n)=\sum_{k=0}^{n}h(k)$

For the given difference equation.

$$h(n) = 2(0.4)^n u(n) - (0.2)^n u(n)$$

i.e., $$h(k) = 2(0.4)^k u(k) - (0.2)^k u(n)$$

$$s(n)=2\sum_{k=0}^{n}(0.4)^k-\sum_{k=0}^{n}(0.2)^k$$

$$=2\frac{1-(0.4)^{n+1}}{1-0.4}-\frac{1-(0.2)^{n+1}}{1-0.2}\left(\text{since }\sum_{k=0}^{n}\alpha^k=\frac{1-\alpha^{n+1}}{1-\alpha},|\alpha|<1\right)$$

$$=\left[\frac{10}{3}-\frac{4}{3}(0.4)^n\right]-\left[\frac{5}{4}-\frac{1}{4}(0.2)^n\right]=\left[\frac{25}{13}-\frac{4}{3}(0.4)^n+\frac{1}{4}(0.2)^n\right]u(n)$$

(iii) The given difference equation of the system is

$$y(n) = 0.7\, y(n-1) - 0.1\, y(n-2) + 2x(n) - x(n-2)$$

i.e., $y(n) - 0.7\, y(n-1) + 0.1\, y(n-2) = 2x(n) - x(n-2)$

To find the impulse response h(n)

For the impulse input, $x(n) = u(n)$. Hence, $y(n) = h\,(n)$

Taking z-transform, we get

$$H\,(z)[1 - 0.7z^{-1} + 0.1z^{-2}] = (2 - z^{-2})$$

$$\frac{H(z)}{z} = \frac{2z^2 - 1}{z(z-0.5)(z-0.2)} = \frac{A_1}{z} + \frac{A_2}{(z-0.5)} + \frac{A_3}{(z-0.2)}$$

$$= -\frac{10}{z} - \frac{10}{3}\frac{1}{(z-0.5)} + \frac{46}{3}\frac{1}{(z-0.2)}$$

$$H(z) = -10 - \frac{10}{3}\frac{z}{(z-0.5)} + \frac{46}{3}\frac{z}{(z-0.2)}$$

Taking inverse z-transform, we get

$$h(z) = -10\delta(n) - \frac{10}{3}(0.5)^n u(n) + \frac{46}{3}(0.2)^n u(n)$$

To find the unit step response s(n)

We know that $s(n) = \sum_{k=0}^{n} h(k)$

$$s(n) = -10\sum_{k=0}^{n}\delta(k) - \frac{10}{3}\sum_{k=0}^{n}(0.5)^k + \frac{46}{3}\sum_{k=0}^{n}(0.2)^k$$

$$= -10 - \frac{10}{3}\left[\frac{1-(0.5)^{n+1}}{1-0.5}\right] + \frac{46}{3}\left[\frac{1-(0.2)^{n+1}}{1-0.2}\right]$$

$$= -10 - \frac{20}{3} + \frac{10}{3}(0.5)^n + \frac{115}{6} - \frac{23}{6}(0.2)^n$$

$$= \left[\frac{5}{2} + \frac{10}{3}(0.5)^n - \frac{23}{6}(0.2)^n\right]u(n)$$

Example 5.40 *A second order discrete time system is characterised by the difference equation $y(n) - 0.1\, y(n-1) - 0.02\, y(n-2) = 2x(n) - x(n-1)$. Determine $y(n)$ for $n \geq 0$ when $x(n) = u(n)$ and the initial conditions are $y(-1) = -10$ and $y(-2) = 5$.*

Solution Using Eqs. (5.14(a)) and (5.14 (b)), we can compute the z-transform of the given difference equation as

$$Y(z) - 0.1[z^{-1}Y(z) + y(-1)] - 0.02[z^{-2}\, Y(z) + z^{-1}y(-1) + y(-2)]$$
$$= 2X(z) - z^{-1}X(z) = (2 - z^{-1})\, X(z)$$

Here, $x(-1) = 0$, $y(-1) = 10$ and $y(-2) = 5$. Since $x(n) = u(n)$,

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

Substituting these values in the above equation, we get

$$Y(z)\,z^{-1}0.1\,[z^{-1}Y(z) - 10] - 0.02\,[z^{-2}Y(z) - 10z^{-1} + 5]$$
$$= (2 - z^{-1})/(1 - z^{-1})$$

Solving this equation for $Y(z)$, we get

$$(1 - 0.1\,z^{-1} - 0.02\,z^{-2})\,Y(z) = (2 - z^{-1})/(1 - z^{-1}) - 0.2\,z^{-1} - 0.9$$

$$Y(z) = \frac{1.1 - 0.3z^{-1} + 0.2z^{-2}}{(1 - z^{-1})(1 - 0.1z^{-1} - 0.02z^{-2})}$$

$$= \frac{1.1 - 0.3z^{-1} + 0.2z^{-2}}{(1 - z^{-1})(1 - 0.2z^{-1})(1 + 0.1z^{-1})} = \frac{1.1z^3 - 0.3z^2 + 0.2z}{(z-1)(z-0.2)(z+0.1)}$$

Therefore,

$$F(z) = \frac{Y(z)}{z} = \frac{1.1z^2 - 0.3z + 0.2}{(z-1)(z-0.2)(z+0.1)} = \frac{A_1}{z-1} + \frac{A_2}{z-0.2} + \frac{A_3}{z+0.1}$$

where

$$A_1 = (z-1)F(z)|_{z=1} = \frac{1.1(1)^2 - 0.3(1) + 0.2}{(1-0.2)(1+0.1)} = 1.136$$

$$A_2 = (z-0.2)F(z)|_{z=0.2} = \frac{1.1(0.2)^2 - 0.3(0.2) + 0.2}{(0.2-1)(0.2+0.1)} = -0.7667$$

and

$$A_3 = (z+0.1)F(z)|_{z=-0.1} = \frac{1.1 - (0.1)^2 - 0.3(-0.1) + 0.2}{(0.1-1)(-0.1-0.2)} = 0.727$$

Therefore, $$Y(z) = 1.136\left[\frac{z}{z-1}\right] - 0.7667\left[\frac{z}{z-0.2}\right] + 0.727\left[\frac{z}{z+0.1}\right]$$

Taking inverse z-transform, we obtain the output signal $y(n)$ given by

$$y(n) = 1.136\,u(n) - 0.7667(0.2)^n\,u(n) + 0.727(-0.1)^n\,u(n)$$

Example 5.41 *Compute the response of the system*

$$y(n) = 0.7y(n-1) - 0.12y(n-2) + x(n-1) + x(n-2)$$

to the input $x(n) = n.u(n)$ using z-transforms.

Solution The given difference equation is

$$Y(z) = 0.7z^{-1}Y(z) - 0.12z^{-2}Y(z) + z^{-1}X(z) + z^{-2}X(z)$$

Therefore, $$H(z) = \frac{Y(z)}{X(z)} = \frac{z^{-1}(1 + z^{-1})}{1 - 0.7z^{-1} + 0.12z^{-2}}$$

The poles of the system function are $z_1 = 0.4$, $z_2 = 0.3$ and the region of convergence is $|z| > 0.4$. Since the poles lie inside the unit circle, the system is stable.

For the input, $x(n) = nu(n)$

$$X(z) = \frac{z}{(z-1)^2}$$

$$\frac{Y(z)}{X(z)} = \frac{z+1}{(z-0.4)(z-0.3)}$$

$$Y(z) = \frac{z(z+1)}{(z-0.3)(z-0.4)(z-1)^2}$$

$$\frac{Y(z)}{z} = \frac{(z+1)}{(z-1)^2(z-0.4)(z-0.3)}$$

$$= \frac{A_1}{z-0.3} + \frac{A_2}{z-0.4} + \frac{A_3}{z-1} + \frac{A_4}{(z-1)^2}$$

$$= -\frac{26.53}{z-0.3} + \frac{38.89}{z-0.4} - \frac{12.36}{z-1} + \frac{4.76}{(z-1)^2}$$

$$Y(z) = -\frac{26.53z}{z-0.3} + \frac{38.89z}{z-0.4} - \frac{12.36z}{z-1} + \frac{4.76z}{(z-1)^2}$$

Taking inverse z-transform, we obtain

$$y(n) = -26.53(0.3)^n u(n) - 38.89(0.4)^n u(n) + 12.36u(n) + 4.7nu(n)$$

FREQUENCY RESPONSE 5.7

Frequency response is a complex function that describes the magnitude and phase shift of a filter over a range of frequencies.

5.7.1 Properties of Frequency Response

If $h(n)$ is a real sequence, the frequency response has the following properties

(a) $H(e^{j\omega})$ takes on values for all ω, i.e. on a continuum of ω.

(b) $H(e^{j\omega})$ is periodic in ω with period 2π.

(c) The magnitude response $|H(e^{j\omega})|$ is an even function of ω and symmetrical about π.

(d) The phase response $|H(e^{j\omega})|$ is an odd function of ω and antisymmetrical about π.

5.7.2 Frequency Response of an Interconnection of Systems

Parallel Connection

If there are L number of linear time invariant systems in the time-domain connected in parallel, the impulse response $h(n)$ of the resultant system is given by

$$h(n) = \sum_{k=1}^{L} h_k(n)$$

where $h_k(n)$, $k=1, 2, \ldots, L$ is the impulse response of the individual systems. By using the linearity property of the z-transform, the frequency response of the overall system is

$$H(z) = \sum_{k=1}^{L} H_k(z) = H_1(z) + H_2(z) + \ldots + H_L(z) \tag{5.34}$$

where $z = e^{j\omega}$. Here $H_k(e^{j\omega})$ is the frequency response to the impulse response $h_k(n)$.

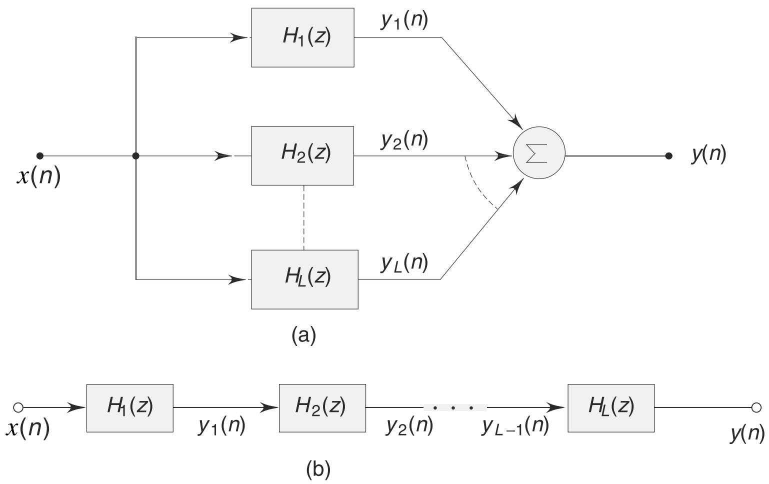


Fig. 5.12 *(a) Parallel, and (b) Cascade Interconnection of Linear Discrete Time Systems*

From the above discussion, it is clear that the parallel interconnection of systems involves additivity in both the time and frequency domain.

Cascade Connection

If the L linear time invariant systems are connected in cascade, the impulse response of the overall system is

$$h(n) = h_1(n) * h_2(n) * \dots h_L(n)$$

Using the convolution property of the z-transform, we get

$$H(z) = H_1(z)\, H_2(z) \cdots H_L(z)$$

Hence, $H(e^{j\omega}) = H_1\left(e^{j\omega}\right) H_2\left(e^{j\omega}\right) \cdots H_L\left(e^{j\omega}\right)$ (5.35)

Here we observe that the cascade connection involves convolution of the impulse responses in the time domain and multiplication of the frequency responses in the frequency domain.

5.7.3 Energy Density Spectrum

In the digital system, the spectrum of the signal at the output of the system is

$$Y(z) = H(z)X(z)\,|_{z=e^{j\omega}} \quad (5.36)$$

Hence, $Y(e^{j\omega}) = H(e^{j\omega})\, X(e^{j\omega})$. This is the desired input-output relation in the frequency domain, which means that the spectrum of the signal at the output of the system is equal to the frequency response of the system multiplied by the spectrum of the signal at the input.

$$|Y(e^{j\omega})|^2 = |H(e^{j\omega})|^2 |\, X(e^{j\omega})|^2 \quad (5.37)$$

As the energy density spectra of $x(n)$ is $S_{xx}(e^{j\omega}) = |X(e^{j\omega})|^2$ and the energy density spectra of $y(n)$ is $S_{yy}(e^{j\omega}) = |Y(e^{j\omega})|^2$, we have

$$S_{yy}(e^{j\omega}) = |H(e^{j\omega})|^2\, S_{xx}(e^{j\omega}) \quad (5.38)$$

5.7.4 Magnitude and Phase Spectrum

The magnitude response is the absolute value of a filter's complex frequency response. The phase response is the angle component of a filter's frequency response. For a linear time invariant system with a real-valued impulse response, the magnitude and phase functions possess symmetry properties which are detailed below. From the definition of z-transform, $H(e^{j\omega})$, a complex function of the real variable ω can be expressed as

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

$$= \sum_{n=-\infty}^{\infty} h(n)\cos\omega n - j\sum_{n=-\infty}^{\infty} h(n)\sin\omega n$$

$$= H_R(e^{j\omega}) + jH_I(e^{j\omega}) = \left|H(e^{j\omega})\right| e^{j\phi(\omega)}$$

$$= \sqrt{H_R^2(e^{j\omega}) + H_I^2(e^{j\omega})}\, e^{j\tan^{-1}[H_I(e^{j\omega})/H_R(e^{j\omega})]}$$

where $H_R(e^{j\omega})$ and $H_I(e^{j\omega})$ denote the real and imaginary components of $H(e^{j\omega})$.

Therefore, $\left|H(e^{j\omega})\right| = \sqrt{H_R^2\left(e^{j\omega}\right) + H_I^2\left(e^{j\omega}\right)}$

$$\Phi(\omega) = \tan^{-1}\left[\frac{H_I\left(e^{j\omega}\right)}{H_R\left(e^{j\omega}\right)}\right]$$

Also, $\Phi(\omega)$ may be expressed as

$$\Phi(\omega) = \frac{1}{2j}\ln\left\{\frac{H(z)}{H\left(z^{-1}\right)}\right\}\Big|_{z=e^{j\omega}}$$

The quantity $|H(e^{j\omega})|$ is called the magnitude function or ***magnitude spectrum*** and the quantity $\Phi(\omega)$ is called the phase function or ***phase spectrum***.

$$\left|H\left(e^{j\omega}\right)\right|^2 = H\left(e^{j\omega}\right)H^*\left(e^{j\omega}\right)$$

$$= H\left(e^{j\omega}\right)H\left(e^{-j\omega}\right) = H(z)H\left(z^{-1}\right)\Big|_{z=e^{j\omega}}$$

Here the function $H(z^{-1})$ has zeros and poles that are the reciprocal of the zeros and poles of $H(z)$. According to the correlation property for the z-transform, the function $H(z)H(z^{-1})$ is the z-transform of the autocorrelation of the unit sample response. Then it follows from the Wiener Khintchine theorem that $|H(z)|^2$ is the Fourier transform of the autocorrelation sequence of $h(n)$.

Example 5.42 *Obtain the frequency response of the first order system with difference equation $y(n) = x(n) + 10y(n-1)$ with initial condition $y(-1) = 0$ and sketch it. Comment about its stability.*

Solution The difference equation of the first order system is $y(n) = x(n) + 10y(n-1)$

Taking z-transform, we get

$$Y(z) = X(z) + 10\,[z^{-1}Y(z) - y(-1)]$$

$$Y(z)[1 - 10z^{-1}] = X(z) \quad [\text{since } y(-1) = 0]$$

Therefore, $H(z) = \dfrac{Y(z)}{X(z)} = \dfrac{1}{1-10z^{-1}} = \dfrac{z}{z-10}$

Frequency Response

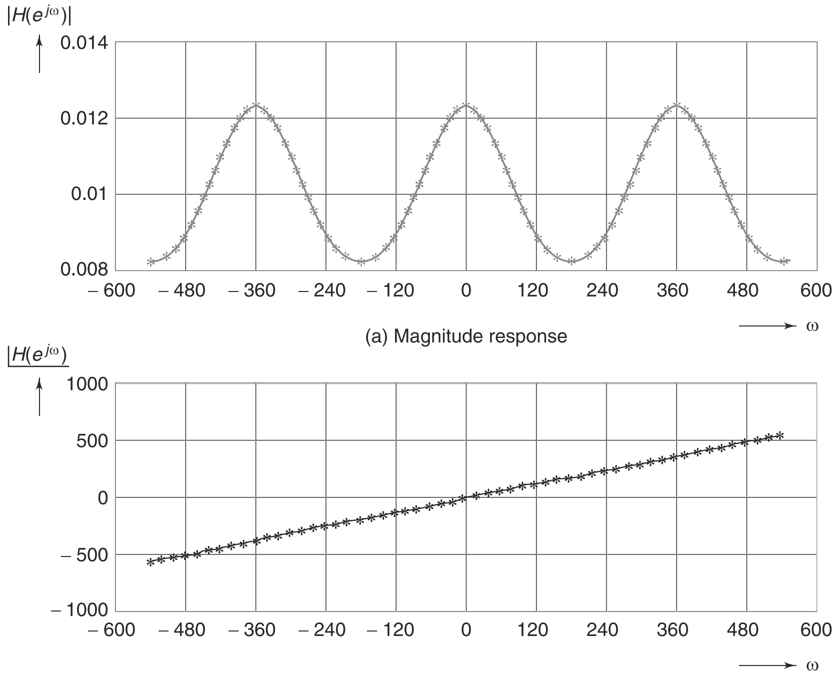
The frequency response is obtained by substituting $z = e^{j\omega}$.

Therefore, $H\left(e^{j\omega}\right) = H(\omega) = \dfrac{e^{j\omega}}{e^{j\omega} - 10}$

Magnitude Response

$$\left|H\left(e^{j\omega}\right)\right| = \frac{\left|e^{j\omega}\right|}{\left|e^{j\omega} - 10\right|} = \frac{1}{\sqrt{(\cos\omega - 10)^2 + \sin^2\omega}} = \frac{1}{101 - 20\cos\omega}$$

The magnitude response is shown in Fig. E5.42(a).



(a) Magnitude response

(b) Phase response

Fig. E5.42 *Magnitude Response and Phase Response*

Phase Response

$$H\left(e^{j\omega}\right) = \frac{e^{j\omega}}{e^{j\omega} - 10} = \frac{\cos\omega + j\sin\omega}{(\cos\omega - 10) + j\sin\omega}$$

$$\Phi(\omega) = \angle H\left(e^{j\omega}\right) = \omega - \tan^{-1}\left(\frac{\sin\omega}{\cos\omega - 10}\right)$$

The phase response is shown in Fig. E5.42(b).

Stability

The transfer function, $H(z) = \dfrac{z}{z - 10}$

Therefore, $h(n) = (10)^n\, u(n)$

$$h\,(n) = 1,\ h\,(1) = 10,\ h\,(2) = 100,\ \ldots\ h(\infty) = \infty$$

$$\sum_{k=0}^{\infty} |n(k)| = \infty$$

Hence the system is constable.

5.7.5 Time Delay

The time delay of a filter is a measure of the average delay of the filter as a function of frequency. It is defined as the negative first derivative of a filter's phase response. If the complex frequency of a filter is $H(e^{j\omega})$, then the group delay is

$$\tau(\omega) = -\frac{d\Phi(\omega)}{d\omega} \tag{5.39}$$

where $\Phi(\omega)$ is the phase angle of $H\,(e^{j\omega})$.

Example 5.43 *Determine the impulse response h (n) for the system described by the second order difference equation.*

$$y(n) - 3y(n-1) - 4y(n-2) = x(n) + 2x(n-1)$$

Solution For the impulse response, $x(n) = \delta(n)$ and hence $y(n) = h(n)$

Therefore, the given difference equation becomes

$$h(n) - 3h(n-1) - 4h(n-2) = \delta(n) + 2\delta(n-1)$$

Taking z-transform and rearranging, we get

$$H(z)[1 - 3z^{-1} - 4z^{-2}] = [1 + 2z^{-1}]$$

Therefore, $$H(z) = \frac{1+2z^{-1}}{1-3z^{-1}-4z^{-2}} = \frac{z(z+2)}{z^2-3z-4} = \frac{z(z+2)}{(z+1)(z-4)}$$

$$F(z) = \frac{H(z)}{z} = \frac{(z+2)}{(z+1)(z-4)} = \frac{A_1}{(z+1)} + \frac{A_2}{(z-4)}$$

where $$A_1 = (z+1)F(z)\big|_{z=-1} = \left.\frac{(z+2)}{(z-4)}\right|_{z=-1} = -\frac{1}{5}$$

$$A_2 = (z-4)F(z)\big|_{z=4} = \left.\frac{(z+2)}{(z+1)}\right|_{z=4} = \frac{6}{5}$$

Therefore, $$\frac{H(z)}{z} = \frac{-\frac{1}{5}}{(z+1)} + \frac{\frac{6}{5}}{(z-4)}$$

Hence, $$H(z) = \frac{-\frac{1}{5}z}{(z+1)} + \frac{\frac{6}{5}z}{(z-4)}$$

Taking inverse z-transform, we get the impulse response

$$h(n) = \left[-\frac{1}{5}(-1)^n + \frac{6}{5}(4)^n\right]u(n)$$

Example 5.44 *Find the magnitude and phase responses for the system characterised by the difference equation*

$$y(n)=\frac{1}{6}x(n)+\frac{1}{3}x(n-1)+\frac{1}{6}x(n-2)$$

Solution Taking z-transform of the given difference equation, we get

$$Y(z)=\frac{1}{6}X(z)+\frac{1}{3}z^{-1}X(z)+\frac{1}{6}z^{-2}X(z)$$

Therefore, $$H(z)=\frac{Y(z)}{X(z)}=\frac{1}{6}+\frac{1}{3}z^{-1}+\frac{1}{6}z^{-2}$$

The frequency response is $H\left(e^{j\omega}\right)=\frac{1}{6}\left[1+2e^{-j\omega}+e^{-j2\omega}\right]$

$$=\frac{1}{6}e^{-j\omega}\left[e^{j\omega}+2+e^{-j\omega}\right]=\frac{1}{3}(1+\cos\omega)\,e^{-j\omega}$$

Hence, magnitude response is $|H(\omega)|=\frac{1}{3}(1+\cos\omega)$, and

Phase response is $\Phi(\omega)=-\omega$.

Example 5.45 *Find the impulse response, frequency response, magnitude response and phase response of the second order system*

$$y(n)-y(n-1)+\frac{3}{16}y(n-2)=x(n)-\frac{1}{2}x(n-1)$$

Solution For the impulse response, $x(n)=\delta(n)$ and hence $y(n)=h(n)$.

Therefore, the given function becomes

$$y(n)-y(n-1)+\frac{3}{16}y(n-2)=x(n)-\frac{1}{2}x(n-1)$$

Taking z-transform, we get

$$H(z)-z^{-1}H(z)+\frac{3}{16}z^{-2}H(z)=1-\frac{1}{2}z^{-1}$$

Therefore, $$H(z)=\frac{1-\frac{1}{2}z^{-1}}{1-z^{-1}+\frac{3}{16}z^{-2}}=\frac{z\left(z-\frac{1}{2}\right)}{z^2-z+\frac{3}{16}}=\frac{z\left(z-\frac{1}{2}\right)}{\left(z-\frac{1}{4}\right)\left(z-\frac{3}{4}\right)}$$

$$\frac{H(z)}{z}=\frac{A_1}{\left(z-\frac{1}{4}\right)}+\frac{A_2}{\left(z-\frac{3}{4}\right)}$$

Evaluating, we get $A_1=\frac{1}{2}$ and $A_2=\frac{1}{2}$

Hence $$\frac{H(z)}{z}=\frac{\frac{1}{2}}{\left(z-\frac{1}{4}\right)}+\frac{\frac{1}{2}}{\left(z-\frac{3}{4}\right)}$$

$$H(z)=\frac{\frac{1}{2}z}{\left(z-\frac{1}{4}\right)}+\frac{\frac{1}{2}z}{\left(z-\frac{3}{4}\right)}$$

Taking inverse z-transform, we have the impulse response

$$h(n)=\left[\frac{1}{2}\left(\frac{1}{4}\right)^n+\frac{1}{2}\left(\frac{3}{4}\right)^n\right]u(n)$$

Since the roots of the characteristic equation have magnitudes less than unity, we know that the impulse response is bounded.

Frequency Response

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

Therefore, $H(z)\big|_{z=e^{j\omega}}=H\left(e^{j\omega}\right)=\frac{1}{2}\left[\frac{1}{1-\frac{1}{4}e^{-j\omega}}+\frac{1}{1-\frac{3}{4}e^{-j\omega}}\right]$

Magnitude and Phase Responses

$$H(z)=\frac{z\left(z-\frac{1}{2}\right)}{\left(z-\frac{1}{4}\right)\left(z-\frac{3}{4}\right)}$$

$$H(e^{j\omega})=\frac{e^{j\omega}\left(e^{j\omega}-\frac{1}{2}\right)}{\left(e^{j\omega}-\frac{1}{4}\right)\left(e^{j\omega}-\frac{3}{4}\right)}$$

Therefore $\quad \left|H\left(e^{j\omega}\right)\right|=\dfrac{\left|\left(e^{j\omega}-\frac{1}{2}\right)\right|}{\left|\left(e^{j\omega}-\frac{1}{4}\right)\right|\left|\left(e^{j\omega}-\frac{3}{4}\right)\right|}$

$$\Phi(\omega)=\omega+\arg\left(e^{j\omega}-\frac{1}{2}\right)-\arg\left(e^{j\omega}-\frac{1}{4}\right)-\arg\left(e^{j\omega}-\frac{3}{4}\right)$$

Example 5.46 *The output y(n) for an LTI system to the input x(n) is*

$$y(n)=x(n)-2x(n-1)+x(n-2)$$

Compute and sketch the magnitude and phase of the frequency response of the system for $|\omega|\geq\pi$.

Solution $\quad y(n)=x(n)-2x(n-1)+x(n-2)$

Taking z-transform on both sides, we get

$$Y(z)=X(z)-2z^{-1}X(z)+z^{-2}X(z)$$

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

Substituting $z=e^{j\omega}$, we get frequency response given by

ω	π	$7\pi/6$	$8\pi/6$	$9\pi/6$	$10\pi/6$	$11\pi/6$	$12\pi/6$	$13\pi/6$	$14\pi/6$	$15\pi/6$	$16\pi/6$	$17\pi/6$	$18\pi/6$
$\lvert H(e^{j\omega})\rvert$	4	3.73	3	2	1	0.268	0	0.268	1	2	3	3.73	4

$$H\left(e^{j\omega}\right)=1-2e^{-j\omega}+e^{-2j\omega}$$

$$=e^{-j\omega}\left[e^{j\omega}-2+e^{-j\omega}\right]=2e^{-j\omega}\left[\cos\omega-1\right]$$

Therefore, the magnitude response is

$$|H(e^{j\omega})|=|2[\cos\omega-1]|$$

The phase response is $\phi(\omega)=\angle H(e^{j\omega})=-\omega$

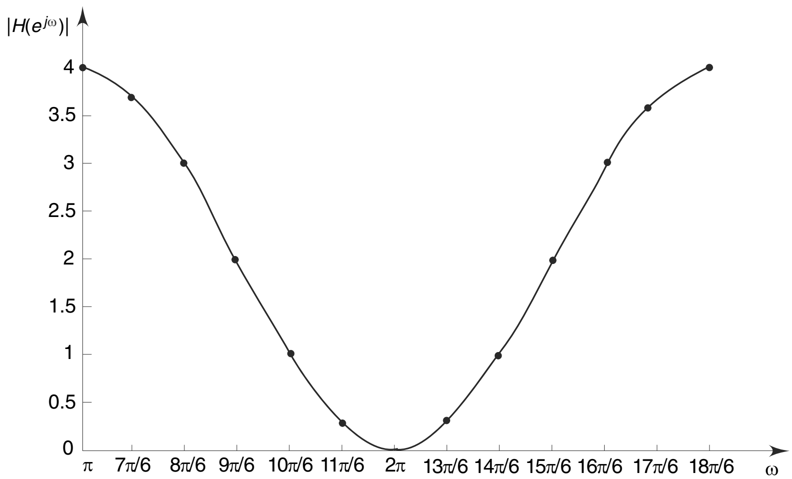


Fig. E5.46(a)

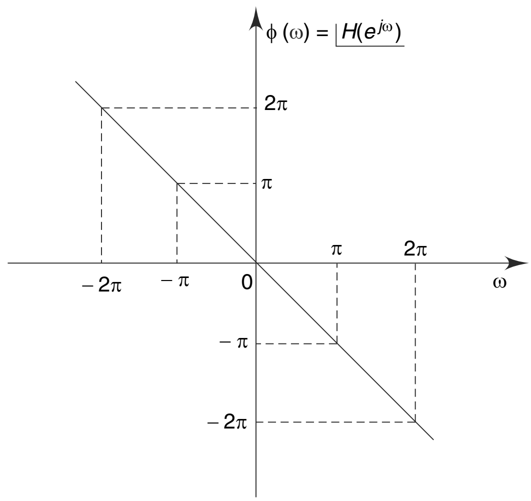


Fig. E5.46(b)

Example 5.47 *Determine the frequency response, magnitude response, phase response and time delay of the system given by*

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

Solution

To find the frequency response H ($e^{j\omega}$)

Given, $$y(n)+\frac{1}{2}y(n-1)=x(n)-x(n-1)$$

Taking z-transform, we get

$$Y(z)+\frac{1}{2}z^{-1}Y(z)=X(z)-z^{-1}X(z)$$

$$Y(z)\left(1+\frac{1}{2}z^{-1}\right)=X(z)[1-z^{-1}]$$

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

Therefore, the frequency response is, $H\left(e^{j\omega}\right)=\dfrac{1-e^{-j\omega}}{1+\frac{1}{2}e^{-j\omega}}$

To find the magnitude response |H ($e^{j\omega}$)|

Here,

$$H\left(e^{j\omega}\right)=\frac{1-e^{-j\omega}}{1+\frac{1}{2}e^{-j\omega}}=\frac{1-\cos\omega+j\sin\omega}{1+\frac{1}{2}\cos\omega-\frac{1}{2}j\sin\omega}$$

$$\left|H\left(e^{j\omega}\right)\right|^2=\frac{(1-\cos\omega)^2+\sin^2\omega}{\left(1+\frac{1}{2}\cos\omega\right)^2+\left(\frac{1}{2}\sin\omega\right)^2}=\frac{2(1-\cos\omega)}{\frac{5}{4}+\cos\omega}=\frac{8(1-\cos\omega)}{5+4\cos\omega}$$

Therefore, $\left|H\left(e^{j\omega}\right)\right|^2=2\sqrt{2}\sqrt{\dfrac{1-\cos\omega}{5+4\cos\omega}}$

To find the phase response Φ(ω)

$$H\left(e^{j\omega}\right)=\frac{1-\cos\omega+j\sin\omega}{1+\frac{1}{2}\cos\omega-\frac{1}{2}j\cos\omega}$$

$$\Phi(\omega)=\tan^{-1}\left(\frac{\sin\omega}{1-\cos\omega}\right)-\tan^{-1}\left(\frac{\frac{1}{2}\sin\omega}{1+\frac{1}{2}\cos\omega}\right)$$

$$=\tan^{-1}\left(\frac{\sin\omega}{1-\cos\omega}\right)+\tan^{-1}\left(\frac{\frac{1}{2}\sin\omega}{1+\frac{1}{2}\cos\omega}\right)=\Phi_1(\omega)+\Phi_2(\omega)$$

***To find the time delay* [τ(ω)]**

$$\tau(\omega) = \tau_1(\omega) + \tau_2(\omega) = -\frac{d\Phi_1}{d\omega} - \frac{d\Phi_2}{d\omega}$$

$$\tau_1(\omega) = -\frac{d\phi_1}{d\omega} = -\frac{(1-\cos\omega)\cos\omega - \sin\omega(\sin\omega)}{(1-\cos\omega)^2 + \sin^2\omega}$$

$$= -\frac{\cos\omega - 1}{2 - 2\cos\omega} = \frac{1}{2}$$

$$\tau_2(\omega) = -\frac{d\phi_2}{d\omega} = -\frac{\left(1+\frac{1}{2}\cos\omega\right)\frac{1}{2}\cos\omega + \frac{1}{2}\sin\omega\left(\frac{1}{2}\sin\omega\right)}{\left(1+\frac{1}{2}\cos\omega\right)^2 + \frac{1}{4}\sin^2\omega}$$

$$= -\frac{\frac{1}{2}\cos\omega + \frac{1}{4}}{\cos\omega + \frac{5}{4}} = -\frac{1+2\cos\omega}{5+4\cos\omega}$$

Hence, $\tau(\omega) = \tau_1(\omega) + \tau_2(\omega)$

$$= \frac{1}{2} - \frac{1+2\cos\omega}{5+4\cos\omega} = \frac{3}{2(5+4\cos\omega)}$$

Note: If $\Phi(\omega) = \tan^{-1}\left[\frac{u(\omega)}{v(\omega)}\right]$, then

$$\tau(\omega) = -\frac{d\Phi(\omega)}{d\omega} = -\frac{(v\,du - u\,dv)}{v^2 + u^2}$$

Example 5.48 *The frequency response of a system is given by*

$$H\left(e^{j\omega}\right) = \frac{e^{j\omega} - a}{e^{j\omega} - b}$$

where a and b are real with a ≠ b. Show that $|H(e^{j\omega})|^2$ *is constant if ab = 1 and determine its value. Also, find the phase response and time delay.*

Solution Given $H\left(e^{j\omega}\right) = \frac{e^{j\omega} - a}{e^{j\omega} - b} = \frac{(\cos\omega - a) + j\sin\omega}{(\cos\omega - b) + j\sin\omega}$

$$\left|H\left(e^{j\omega}\right)\right|^2 = \frac{(\cos\omega - a^2) + \sin^2\omega}{(\cos\omega - b^2) + \sin^2\omega} = \frac{1 + a^2 - 2a\cos\omega}{1 + b^2 - 2b\cos\omega}$$

Substituting $b = \frac{1}{a}$ [since $ab = 1$], we have

$$\left|H\left(e^{j\omega}\right)\right|^2 = \frac{1 + a^2 - 2a\cos\omega}{1 + \frac{1}{a^2} - \frac{2}{a}\cos\omega} = a^2 . \frac{1 + a^2 - 2a\cos\omega}{1 + a^2 - 2a\cos\omega} = a^2$$

To find phase response Φ(ω)

$$H\left(e^{j\omega}\right)=\frac{e^{j\omega}-a}{ae^{j\omega}-\dfrac{1}{a}}=a.\frac{e^{j\omega}-a}{a\,e^{j\omega}-1}=-a\,e^{j\omega}\,\frac{1-ae^{-j\omega}}{1-ae^{j\omega}}$$

$$=-ae^{j\omega}\,\frac{1-a\cos\omega+ja\sin\omega}{1-a\cos\omega-ja\sin\omega}$$

$$\Phi(\omega)=\omega-\pi+\tan^{-1}\left[\frac{a\sin\omega}{1-a\cos\omega}\right]-\tan^{-1}\left[\frac{-a\sin\omega}{1-a\cos\omega}\right]$$

$$=\omega-\pi+2\tan^{-1}\left[\frac{a\sin\omega}{1-a\cos\omega}\right]$$

To find time delay τ(ω)

i.e. $$\tau(\omega)=-\frac{d\phi(\omega)}{d\omega}=-1-2\,\frac{(1-a\cos\omega)a\cos\omega-(a\sin\omega)a\sin\omega}{(1-a\cos\omega)^2+(a\sin\omega)^2}$$

$$=-1-\frac{a\cos\omega-a^2}{1-2\cos\omega+a^2}=\frac{-1+a\cos\omega}{1-2a\cos\omega+a^2}$$

5.7.6 Geometrical Construction Method to Determine Magnitude and Phase of the Frequency Response of a Digital System

Like an analog filter, a digital filter can be represented in the frequency domain by a magnitude response and a phase response. The difference between analog and digital-filters is that the transfer function of the analog filter is calculated on the imaginary axis of the s-plane whereas the transfer function of the digital filter is evaluated on the unit circle $|z|=1$ of the z-plane.

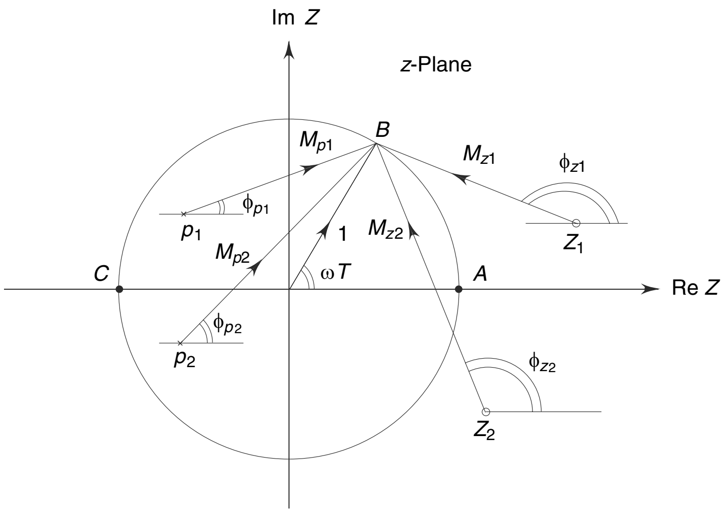


Fig. 5.13 *Geometric Representation for a Second Order Recursive Filter*

The geometric evaluation of the magnitude and phase responses of the system with a greater number of poles and zeros can be done by determining the magnitude and phase angles of the vectors at a given frequency on the unit circle $|z|=|e^{j\omega T}|=1$. The magnitude $M(\omega)$ and phase $\Phi(\omega)$ of the frequency response of a digital system can be determined using the geometrical construction method as explained below.

Points A and C in Fig. 5.13 correspond to frequencies 0 and $\omega_s/2$, the Nyquist frequency or folding frequency, which is equal to πf_s or π/T, where f_s is the sampling frequency and T is the sampling period. One complete revolution of the phase $e^{j\omega T}$ about the origin corresponds to a frequency increment of $\omega_s = \dfrac{2\pi}{T}$. Here, $H(e^{j\omega T})$ is a periodic function of frequency with a period ω_s. The frequency response has the property $H(e^{j\omega T}) = H(e^{-j\omega T})$. Therefore, the magnitude function $M(\omega)$ is an even function of ω, and the phase function $\Phi(\omega)$ is an odd function of ω. Vectors are drawn from each pole and zero to $e^{j\omega T}$ point on the unit circle.

The transfer function of a digital system may be expressed in term of its poles and zeros as

$$H\left(e^{j\omega T}\right) = M(\omega)e^{j\Phi(\omega)} = \frac{H_0 \prod_{i=1}^{p} (e^{j\omega T} - z_i)}{\prod_{i=1}^{q} (e^{j\omega T} - p_i)} \tag{5.40}$$

By substituting, $\left(e^{j\omega T} - z_i\right) = M_{zi} e^{j\Phi z_i}$

and $\left(e^{j\omega T} - p_i\right) = M_{pi} e^{j\Phi p_i}$

We have, Magnitude as

$$M(\omega) = \left|H\left(e^{j\omega}\right)\right| = \frac{H_0 \prod_{i=1}^{p} M_{zi}}{\prod_{i=1}^{q} M_{pi}}$$

$$= \frac{H_0 \prod_{i=1}^{P} \{\text{Vector magnitude from the } i^{\text{th}} \text{ zero to the frequency point on the circumference of the unit circle}\}}{\prod_{i=1}^{q} \{\text{Vector magnitude from the } i^{\text{th}} \text{ pole to the frequency point on the circumference of the unit circle}\}}$$

That is, the magnitude of the system function $M(\omega)$ equals the product of all zero vector lengths, divided by the product of all pole vector lengths.

Phase Shift

$$\Phi(\omega) = \left|H\left(e^{j\omega}\right)\right| = \sum_{i=1}^{p} \Phi_{z_i} - \sum_{i=1}^{q} \Phi_{p_i}$$

$$= \sum_{i=1}^{p} \{\text{Angle form the } i^{\text{th}} \text{ zero to the frequency point on the circumference of the unit circle}\}$$

$$- \sum_{i=1}^{q} \{\text{Angle form the } i^{\text{th}} \text{ pole to the frequency point on the circumference of the unit circle}\}$$

That is, the phase $\Phi(\omega)$ equals the sum of all zero vector phases, minus the sum of all pole vector phases.

Example 5.49 *Determine the frequency response, magnitude response and phase response for the system given by*

$$y(n)-\frac{3}{4}y(n-1)+\frac{1}{8}y(n-2)=x(n)-x(n-1)$$

for $T = 1$ msec and $f = 0$ Hz, 10 Hz and 100 Hz and 1 kHz. Also, give a geometric interpretation of $H(e^{j\omega T})$ for finding magnitude and phase responses.

Solution For the impulse response, $x(n)=\delta(n)$ and hence $y(n)=h(n)$

Therefore, $h(n)-\frac{3}{4}h(n-1)+\frac{1}{8}h(n-2)=\delta(n)-\delta(n-1)$

Taking z-transform, we get

$$H(z)-\frac{3}{4}z^{-1}H(z)+\frac{1}{8}z^{-2}H(z)=1-z^{-1}$$

Therefore, $$H(z)=\frac{1-z^{-1}}{1-\frac{3}{4}z^{-1}+\frac{1}{8}z^{-2}}=\frac{1-z^{-1}}{\left(1-\frac{1}{4}z^{-1}\right)\left(1-\frac{1}{2}z^{-1}\right)}$$

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

Therefore, the frequency response is

$$H\left(e^{j\omega T}\right)=\frac{e^{j\omega T}\left(e^{j\omega T}-1\right)}{\left(e^{j\omega T}-\frac{1}{4}\right)\left(e^{j\omega T}-\frac{1}{2}\right)}$$

$$M(\omega)=\left|H\left(e^{j\omega T}\right)\right|=\frac{\left|e^{j\omega T}-1\right|}{\left|e^{j\omega T}-\frac{1}{4}\right|\left|e^{j\omega T}-\frac{1}{2}\right|}$$

$$\Phi(\omega)=\left|H\left(e^{j\omega T}\right)\right|=\omega T+\arg\left(e^{j\omega T}-1\right)-\arg\left(e^{j\omega T}-\frac{1}{4}\right)-\arg\left(e^{j\omega T}-\frac{1}{2}\right)$$

Using the above equations, the magnitude response $M(\omega)$ and the phase response $\Phi(\omega)$ for $T=1$ msec and $f=0$ Hz, 10 Hz, 100 Hz and 1 kHz are calculated and tabulated below. The geometric interpretation for $M(\omega)$ and $\Phi(\omega)$ is shown in Fig. E5.49.

Table 5.2 *Magnitude and phase response for different values of frequency*

	0 Hz	10 Hz	100 Hz	1000 Hz
$M(\omega)$	0	1.667	1.14737	0
$\phi(\omega)$	0	−96.5°	−144.7°	0 (or) 360°

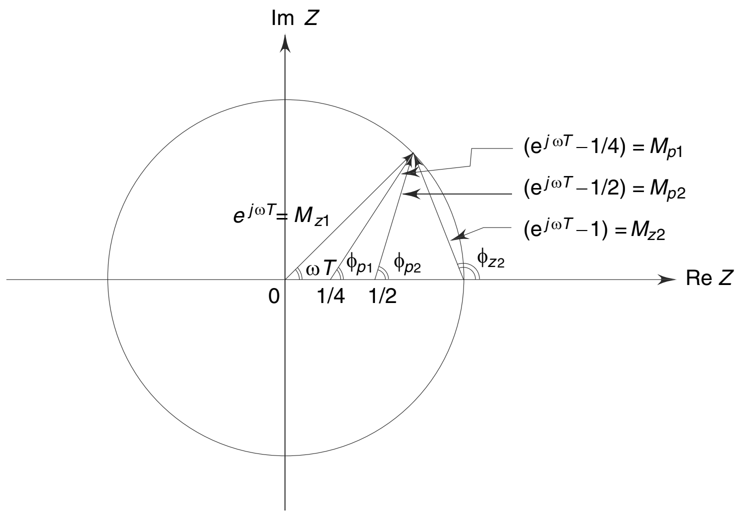


Fig. E5.49 *Geometric Interpretation for M(ω) and Φ(ω) [not to scale]*

$$M(\omega) = \frac{Mz_1 Mz_2}{Mp_1 Mp_2} \quad \text{and} \quad \Phi(\omega) = \omega T + \Phi z_2 - (\Phi_{p1} + \Phi_{p2})$$

REVIEW QUESTIONS

5.1 When is a system said to be linear?

5.2 Define the terms (i) linearity (ii) time invariance and (iii) causality as applied to a discrete time system.

5.3 Explain shift-invariant systems in detail.

5.4 With an example, discuss in detail time invariance and causality for a discrete-time system.

5.5 Explain briefly the Paley-Wiener Criterion.

5.6 Explain (i) Causality and (ii) stability of a linear time invariant system.

5.7 State the conditions for a digital filter to be causal and stable.

5.8 What is the necessary and sufficient condition for stability?

5.9 What is the condition for a system to be BIBO stable?

5.10 Show that the necessary and sufficient condition for a LTI system to be stable is

$$\sum_{n=-\infty}^{\infty} |h(n)| < \infty \text{ where } h(n) \text{ is the impulse response.}$$

5.11 How will you test the stability of a digital filter?

5.12 How will you obtain the power spectral density of a digital filter?

5.13 Write the mathematical and graphical representation of a unit sample sequence.

5.14 Give the mathematical and graphical representation of a unit step sequence

5.15 State and explain the transfer function of an LTI system.

5.16 What is the impulse response of two LTI systems connected in cascade?

5.17 Explain in detail the system transfer function.

5.18 Discuss briefly the frequency response of LTI systems.

5.19 Explain the method of obtaining the frequency response function of linear shift-invariant systems.

5.20 Describe the relationship between impulse response and frequency response of a discrete time system.

5.21 Discuss magnitude response, phase response and time delay.

5.22 Distinguish between IIR and FIR systems.

5.23 Determine the impulse response and the unit step response of the systems described by the difference equation

$$y(n) = 0.6\,y(n-1) - 0.08\,y(n-2) + x(n)$$

5.24 A system is characterised by

$$y(n) - by(n-1) = x(n)$$

where $x(n) = 1$ for $n \geq 0$

$= 0$ for $n < 0$, and $y(0) = 1$

Obtain the expression for step response of the system.

5.25 Verify whether the following impulse responses describe causal, stable or LTI systems. Give reasons for your answers.

(a) $h(n) = e^{-0.6n}\,u(n)$

(b) $h(n) = \sin(n)\,e^{n}\,u(n)$

(c) $h(n) = 4\,\delta(n-2) + 2\,\delta(n-4)$

(d) $h(n) = u(n-2) - u(n+3)$

(e) $h(n) = \begin{cases} \cos\left(\dfrac{n\pi}{8}\right), & -1 < n < 15 \\ 0, & \text{otherwise} \end{cases}$

Ans: (a) and (c) are causal and stable.
(b) is causal and unstable.
(d) and (e) are non-causal and stable.

5.26 For each of the following discrete-time signals, determine whether or not the system is linear, shift-invariant, causal and stable.

(*a*) $y(n) = x(n+7)$, (*b*) $y(n) = x^3(n)$, (*c*) $y(n) = n\,x(n)$

(*d*) $y(n) = \alpha + \sum_{k=0}^{4} x(n-k), \alpha$ is a non-zero constant.

(e) $y(n) = \alpha + \sum_{k=-4}^{4} x(n-k), \alpha$ is a non-zero constant.

Ans: (a) Linear, time-invariant, non-causal and stable.
(b) Non-linear, time-invariant, causal and BIBO stable.
(c) Linear, not time-invariant, causal and not BIBO stable.
(d) Non-linear, time-invariant, causal and BIBO stable.
(e) Non-linear, time-invariant, non-causal and BIBO stable.

5.27 Determine whether the following systems are linear or non-linear, causal or non-causal, shift invariant or shift-variant .

(i) $y(nT) = x(nT+T) + x(nT-T)$.

(ii) $y(nT) = x^2(nT+T)\, e^{-nT} \sin \omega nT$

(iii) $y(n) = a\, y(n-1) + x(n)$

5.28 Using the Paley-Wiener Criterion, show that $|H(f)| = e^{-\beta f^2}$ is not a suitable amplitude response for a causal LTI system.

5.29 Discuss the stability of the system described by

$$H(z) = \frac{z^{-1}}{1 - z^{-1} - z^{-2}}$$

5.30 Find the stability region for the causal system

$$H(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

by evaluating its poles and restricting them to be inside the unit circle.

5.31 A causal LTI system is described by the difference equation

$$y(n) = y(n-1) + y(n-2) + x(n-1)$$

where $x(n)$ is the input and $y(n)$ is the output.

(i) Find the system function $H(z) = \dfrac{Y(z)}{X(z)}$ for this system, plot the poles and zeros of $H(z)$ and indicate the region of convergence.

(ii) Find the unit sample response of the system.

(iii) Is the system stable or not?

5.32 Find the system function $H(z)$ and frequency response if the system is described by

$$y(nT) + \frac{3}{2} y(nT-T) + \frac{1}{2} y(nT-2T) = x(nT) + \frac{1}{2} x(nT-T)$$

Assume $T = 1$ sec. Sketch the amplitude and phase responses.

5.33 Obtain the condition for the stability of the filter

$$H(z) = \frac{a_1 + a_2 z^{-1}}{1 + b_1 z^{-1} + b_2 z^{-2}}$$

5.34 A system has unit sample response $h(n)$ given by

$$h(n) = -\frac{1}{4}\delta(n+1) + \frac{1}{2}\delta(n) - \frac{1}{4}\delta(n-1)$$

(i) Is the system BIBO stable?

(ii) Is the filter causal?

(iii) Find the frequency response.

Ans: (i) BIBO stable.

(ii) Non-causal

(iii) $H(e^{j\omega}) = 0.5(1 - \cos \omega)$

5.35 Determine the linear convolution of the following sequences

$$x(n) = (2, 5, -1, 4) \text{ and } y(n) = (-4, 2, 1, 0, -1)$$

5.36 Given $y(n)$, the linear convolution of $x(n)$, the input sequence and $h(n)$, the impulse response. Find the input sequence $x(n)$ if

$$y(n) = (2, 1, 2, 6, 0, 17, 12) \text{ and } h(n) = (2, -3, 2, 3)$$

5.37 Determine the range of the parameter k for which the system defined by the following difference equation is stable.

$$y(nT) - 2k\,y(nT - T) + k2\,y(nT - 2T) = x(nT).$$

Ans: $0 < k < 1$.

5.38 Determine the step response of an LTI system whose impulse response $h(n)$ is given by $h(n) = a^{-n}u(-n)$, $0 < a < 1$.

5.39 Determine the response of the system defined by the equation

$$y(n) = \frac{5}{6}y(n-1) - \frac{1}{6}y(n-2) + x(n)$$

to the input signal $x(n) = \delta(n) - \frac{1}{3}\delta(n-1)$ assuming zero initial condition.

Ans: $h(n) = \left(\frac{1}{2}\right)^n u(n)$.

5.40 Find the general solution of the first order difference equation

$$y(n) - 0.9\,y(n-1) = 0.5\,x(n) + 0.9\,x(n-1);\ n \geq 0 \text{ with } y(-1) = 5.$$

5.41 Find the impulse response for the system given by

$$y(n) + 4y(n-1) + 4y(n-2) = x(n)$$

Ans: $h(n) = (n+1)(-2)^n u(n)$

5.42 Find the frequency response of the system

(i) $y(n) - \frac{1}{2}y(n-1) = x(n)$

Ans: $H\left(e^{j\omega}\right) = \dfrac{1}{\left(1 - \frac{1}{2}e^{-j\omega}\right)}$

(ii) $y(n) + \frac{1}{2}y(n-1) = x(n) - x(n-1)$

Ans: $H\left(e^{j\omega}\right) = \dfrac{1 - e^{-j\omega}}{1 + \frac{1}{2}e^{-j\omega}}$

5.43 Compute the spectrum of the finite sequence $x(n)$ for $N = 6$ if

$$x(0) = x(4) = 3,\ x(1) = x(3) = 2 \text{ and } x(2) = x(5) = 1.$$

Ans: $X(e^{j\omega}) = 3 + 2e^{-j\omega} + e^{-j2\omega} + 2e^{-j3\omega} + 3e^{-j4\omega} + e^{-j5\omega}$

5.44 A causal system is described by the following difference equation

$$y(n)+\frac{1}{4}y(n-1)=x(n)+\frac{1}{2}x(n-1)$$

Find the system transfer function $H(z)$ and give the corresponding region of convergence. Also, obtain the corresponding unit sample response of the above system.

Ans: $H(z)=\dfrac{1+\frac{1}{2}z^{-1}}{1+\frac{1}{4}z^{-1}}$.

5.45 The transfer functions of two first order systems are given by

$$y_1(n) = x_1(n) - 0.5\, y_1(n-1)$$

$$y_2(n) = x_2(n)+y_2(n-1)\ ;\ n\geq 0$$

(i) Find $H_1(z)$ and $H_2(z)$

(ii) Evaluate the output of $y(n)$ if $H_1(z)$ and $H_2(z)$ are connected in parallel when the input $x(n)=(-1)^n,\ n\geq 0$

5.46 Find the solution to the following linear constant coefficient difference equation

$$y(n)-\frac{3}{2}y(n-1)+\frac{1}{2}y(n-2)=\left(\frac{1}{4}\right)^n \quad \text{for } \geq 0$$

with initial conditions $y(-1)=4$ and $y(-2)=10$.

Ans: $y(n)=\left[\frac{1}{3}\left(\frac{1}{4}\right)^n+\left(\frac{1}{2}\right)^n+\frac{2}{3}\right]u(n)$

5.47 Compute the spectrum of the sequence $f(n)=(6, -10, 0, 4, -2, 1)$ with 1 msec sampling time.

5.48 Determine the magnitude and phase response of the following systems.

(a) $y(n)=\frac{1}{2}[x(n)+x(n-1)]$ **Ans:** $M(\omega)=\cos\frac{\omega}{2},\ \phi(\omega)=-\frac{\omega}{2}$

(b) $y(n)=\frac{1}{2}[x(n)-x(n-1)]$ **Ans:** $M(\omega)=\sin\frac{\omega}{2},\ \phi(\omega)=\frac{\pi}{2}-\frac{\omega}{2}$

(c) $y(n)=\frac{1}{2}[x(n+1)-x(n-1)]$ **Ans:** $M(\omega)=\sin\omega,\ \phi(\omega)=\frac{\pi}{2}$

(d) $y(n)=\frac{1}{2}[x(n+1)+x(n-1)]$ **Ans:** $M(\omega)=\cos\omega,\ \phi(\omega)=0$

(e) $y(n)=\frac{1}{2}[x(n)+x(n-2)]$ **Ans:** $M(\omega)=\cos\omega,\ \phi(\omega)=-\omega$

(f) $y(n)=\frac{1}{2}[x(n)-x(n-2)]$ **Ans:** $M(\omega)=\sin\omega,\ \phi(\omega)=\frac{\pi}{2}-\omega$

5.49 Determine the frequency response, magnitude and phase responses and time delay of the systems given by

(i) $y(n) = x(n) - x(n-1) + x(n-2)$ and

(ii) $y(n)-\frac{1}{2}y(n-1)=x(n)$

Ans: (i) $H\left(e^{j\omega}\right)=e^{-j\omega}(2\cos\omega-1)$

$$\left|H\left(e^{j\omega}\right)\right|=|(2\cos\omega-1)|$$

$$\Phi(\omega)=-\omega,\ 0\le\omega<\frac{\pi}{3}$$

$$=-\omega\pm\pi,\ \frac{\pi}{3}<\omega<\pi$$

$$\tau(\omega)=-\frac{\mathrm{d}\Phi}{\mathrm{d}\omega}=1,\ \omega\ne\frac{\pi}{3}$$

(ii) $H\left(e^{j\omega}\right)=1-e^{j\omega}+e^{-j2\omega}$

$$\left|H\left(e^{j\omega}\right)\right|=\frac{1}{\sqrt{\frac{5}{4}-\cos\omega}}$$

$$\Phi(\omega)=-\tan^{-1}\left[\frac{\frac{1}{2}\sin\omega}{1-\frac{1}{2}\cos\omega}\right]$$

$$\tau(\omega)=\frac{2\cos\omega-1}{5-4\cos\omega}$$

5.50 Determine frequency, magnitude and phase responses and time delay for the system

$$y(n)+\frac{1}{4}y(n-1)=x(n)-x(n-1)$$

Ans: $H\left(e^{j\omega}\right)=\dfrac{1-\cos\omega+j\sin\omega}{1+\frac{1}{4}\cos\omega-j\frac{1}{4}\sin\omega}$

$$\left|H\left(e^{j\omega}\right)\right|=4\sqrt{\frac{2(1-\cos\omega)}{17+8\cos\omega}}$$

$$\Phi(\omega)=\tan^{-1}\left[\frac{\left(\frac{1}{4}\right)\sin\omega}{1+\frac{1}{4}\cos\omega}\right]+\tan^{-1}\left[\frac{\sin\omega}{1-\cos\omega}\right]$$

$$\tau(\omega)=\frac{1}{2}+\frac{1+4\cos\omega}{17+8\cos\omega}$$

5.51 Calculate the amplitude and phase responses of the filter whose transfer function is

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

and plot approximately, assuming $T=1$ msec.

5.52 Calculate the amplitude and phase response of

(i) $H(z)=\dfrac{(z+2)(z-3)}{\left(z+\frac{1}{2}\right)\left(z-\frac{1}{3}\right)}$ (ii) $H(z)=\dfrac{(5+z^{-1})(3+z^{-1})}{\left(2+z^{-1}\right)\left(4+z^{-1}\right)}$

for T = 1 msec and f = 0 Hz, 10 Hz, 100 Hz and 1 kHz.

Chapter 6

Discrete and Fast Fourier Transforms

INTRODUCTION 6.1

Chapters 2 and 3 have established an idea that a continuous-time signal can be represented in the frequency domain. In Chapter 4, the fact that a discrete-time signal can have a z-transform representation was introduced. These two ideas have provided a gateway to the study of discrete-time signals in the frequency domain. Direct computation of Discrete Fourier Transform (DFT) requires about N^2 complex multiplications, which can be reduced by using symmetry properties of the trigonometric functions. In 1965, Cooley and Tukey gave a method of computing DFTs requiring a number of operations proportional to $N \log_2 N$, which is referred to as Fast Fourier Transform (FFT). The FFT is claimed to be an $N \log (N)$ algorithm. Therefore, the speed up is in the order of $(\log_2 N)/N$ where N is the time-series sample length.

In the design of a DSP system, two fundamental tasks are involved, viz. (i) analysis of the input signal, and (ii) design of a processing system to give the desired output. The Discrete Fourier Transform (DFT) and Fast Fourier Transform (FFT) are very important mathematical tools for carrying out these tasks. They can be used to analyse a two-dimensional signal. The FFT algorithms eliminate the redundant calculation and enable to analyse the spectral properties of a signal. They offer rapid frequency-domain analysis and processing of digital signals, and investigation of digital systems. The FFT also allows time domain signal processing operations to be performed equivalently in the frequency-domain. In both domains FFT has considerable reduction in computation time. FFT algorithms are mainly useful in computing the DFT and IDFT and also find applications in linear filtering, digital spectral analysis and correlation analysis.

DISCRETE FOURIER SERIES 6.2

Any periodic function can be expressed in a Fourier series representation. Consider a discrete-time signal $x(n)$ that is periodic with period N defined by $x(n) = x(n + kN)$ for any integer value of k.

The periodic function $x(n)$ can be synthesised as the sum of sine and cosine sequences or equivalently a linear combination of complex exponentials whose frequencies are multiples of the fundamental frequency $2\pi/N$. This is done by constructing a periodic sequence for which each period is identical to the finite-length sequence.

Exponential Form of Discrete Fourier Series

A real periodic discrete-time signal $x(n)$ of period N can be expressed as a weighted sum of complex exponential sequences. As the sinusoidal sequences are unique only for digital frequencies from 0 to 2π, the expansion has only a finite number of complex exponentials.

The exponential form of the Fourier series for a periodic discrete-time signal is given by

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{jk\omega_0 n}, \quad \text{for all } n \tag{6.1}$$

where the coefficients $X(k)$ and the fundamental digital frequency ω_0 are expressed as

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-jk\omega_0 n}, \quad \text{for all } k \tag{6.2}$$

where $\omega_0 = 2\pi/N$

Equations (6.1) and (6.2) are called the Discrete Fourier Series (DFS) synthesis and analysis pair. Here both $X(k)$ and $x(n)$ are periodic sequences.

The equivalent form of Eq. (6.2) for $X(k)$ is

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn} \tag{6.3a}$$

where W_N is defined by

$$W_N = e^{-j2\pi/N} \tag{6.3b}$$

Trigonometric Form of Discrete Fourier Series

The trigonometric Fourier series representation of a continuous-time periodic signal $f(t)$ is given by

$$f(t) = a_0 + \sum_{n=1}^{\infty}(a_n \cos n\omega_0 t + b_n \sin n\omega_0 t)$$

where the constants a_0, a_n and b_n can be determined as

$$a_0 = \frac{1}{T}\int_0^T f(t)\,\mathrm{d}t$$

$$a_n = \frac{2}{T}\int_0^T f(t)\cos n\omega_0 t\,\mathrm{d}t, \qquad n = 1, 2, \ldots$$

$$b_n = \frac{2}{T}\int_0^T f(t)\sin n\omega_0 t\,\mathrm{d}t, \qquad n = 1, 2, \ldots$$

From the above equation, we can find an alternative form of the trigonometric Discrete Fourier Series for odd N and even N as given below:

For even *N*:

$$x(n) = A(0) + \sum_{k=1}^{N/2-1} A(k)\cos\left(k\frac{2\pi}{N}n\right) + \sum_{k=1}^{N/2-1} B(k)\cos\left(k\frac{2\pi}{N}n\right) + A\left(\frac{N}{2}\right)\cos\pi n \tag{6.4}$$

Here, the last term contains $\cos \pi n$, which is equal to $(-1)^n$, the highest frequency sequence possible. The constants $A(0)$, $A(k)$ and $B(k)$ will be as follows:

$$A(0) = \frac{1}{N}\sum_{n=0}^{N-1} x(n) \tag{6.5a}$$

$$A(k) = \frac{2}{N}\sum_{n=0}^{N-1} x(n)\cos\left(k\frac{2\pi}{N}n\right), k = 1,2,\ldots\frac{N}{2}-1 \tag{6.5b}$$

$$B(k) = \frac{2}{N}\sum_{n=0}^{N-1} x(n)\sin\left(k\frac{2\pi}{N}n\right), k = 1,2,\ldots\frac{N}{2}-1 \tag{6.5c}$$

$$A(N/2) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\cos\pi n \tag{6.5d}$$

For odd N:

$$x(n) = A(0) + \sum_{k=1}^{(N-1)/2} A(k)\cos\left(k\frac{2\pi}{N}n\right) + \sum_{k=1}^{(N-1)/2} B(k)\sin\left(k\frac{2\pi}{N}n\right) \tag{6.6}$$

The constants $A(0)$, $A(k)$ and $B(k)$ will be

$$A(0) = \frac{1}{N}\sum_{n=0}^{N-1} x(n) \tag{6.7a}$$

$$A(k) = \frac{2}{N}\sum_{n=0}^{N-1} x(n)\cos\left(k\frac{2\pi}{N}n\right), k = 1,2,\ldots(N-1)/2 \tag{6.7b}$$

$$B(k) = \frac{2}{N}\sum_{n=0}^{N-1} x(n)\sin\left(k\frac{2\pi}{N}n\right), k = 1,2,\ldots(N-1)/2 \tag{6.7c}$$

Relationships between the Exponential and Trigonometric Forms of DFS

For even N, the relationships between $X(k)$ of the exponential form and $A(k)$ and $B(k)$ of the trigonometric form for a real $x(n)$ are expressed by

$$A(0) = X(0)/N \tag{6.8a}$$

$$A(k) = [X(k) + X(N-k)]/N, \quad k = 1,2,\ldots\frac{N}{2}-1 \tag{6.8b}$$

$$B(k) = j[X(k) - X(N-k)]/N, \quad k = 1,2,\ldots\frac{N}{2}-1 \tag{6.8c}$$

$$A(N/2) = X(N/2)/N \tag{6.8d}$$

For odd N, the relationships between $A(k)$ and $B(k)$ of the trigonometric form, and $X(k)$ of the exponential form for a real $x(n)$ are expressed by

$$A(0) = X(0)/N \tag{6.9a}$$

$$A(k) = [X(k) + X(N-k)]/N, \quad k = 1,2,\ldots(N-1)/2 \tag{6.9b}$$

$$B(k) = j[X(k) - X(N-k)]/N, \quad k = 1,2,\ldots(N-1)/2 \tag{6.9c}$$

Example 6.1 *Find both the exponential and trigonometric forms of the Discrete Fourier Series representation of $x(n)$ shown in Fig. E6.1.*

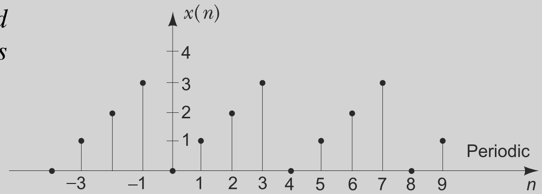


Fig. E6.1

Solution To determine the exponential form of the Discrete Fourier Series:

We know that $W_N^k = e^{-j(2\pi/N)k}$.

Given $N = 4$, hence, $W_4^1 = e^{-j(2\pi/4)} = \cos\frac{\pi}{2} - j\sin\frac{\pi}{2} = -j$

$$W_4^2 = W_4^1 \cdot W_4^1 = (-j)(-j) = -1$$

$$W_4^3 = W_4^2 \cdot W_4^1 = (-1)(-j) = j$$

$$W_4^4 = W_4^3 \cdot W_4^1 = (j)(-j) = 1$$

For $k = 0, X(0) = \sum_{n=0}^{4-1} x(n)W_4^{0-n} = x(0) + x(1) + x(2) + x(3) = 0 + 1 + 2 + 3 = 6$

For $k = 1, X(1) = \sum_{n=0}^{4-1} x(n)W_4^{1-n} = x(0)W_4^0 + x(1)W_4^1 + x(2)W_4^2 + x(3)W_4^3$

$$= 0(1) + 1(-j) + 2(-1) + 3(j) = -2 + j2$$

For $k = 2$, $X(2) = \sum_{n=0}^{4-1} x(n)W_4^{2-n} = x(0)W_4^0 + x(1)W_4^2 + x(2)W_4^4 + x(3)W_4^6$

$$= 0(1) + 1(-1) + 2(1) + 3(-1) = -2$$

For $k = 3$, $X(3) = \sum_{n=0}^{4-1} x(n)W_4^{3-n} = x(0)W_4^0 + x(1)W_4^3 + x(2)W_4^6 + x(3)W_4^9$

$$= x(0)W_4^0 + x(1)W_4^3 + x(2)W_4^2 + x(3)W_4^1 = 0(1) + 1(j) + 2(-1) + 3(-j) = -2 - j2$$

Substituting the above values in Eq. (6.1), we get the complex exponential form of the Discrete Fourier Series as

$$x(n) = \frac{1}{4}\left[6 + [-2 + j2]W_4^{-n} + (-2)W_4^{-2n} + [-2 - j2]W_4^{-3n}\right]$$

$$= \frac{1}{2}\left[3 + (-1 + j)e^{j(\pi/2)n} - e^{j\pi n} - (1 + j)e^{j(3\pi/2)n}\right]$$

To determine the trigonometric form of the Discrete Fourier Series

Using Eq. (6.8), the $A(0)$, $A(1)$, $B(1)$, and $A(2)$ are determined as

$$A(0) = X(0)/N = \frac{6}{4} = \frac{3}{2}$$

$$A(1) = [X(1) + X(4-1)]/4 = [(-2 + j2) + (-2 - j2)]/4 = -1$$

$$B(1) = j[X(1) - X(4-1)]/4 = j[(-2 + j2) - (-2 - j2)]/4 = -1$$

$$A\left(\frac{4}{2}\right) = X\left(\frac{4}{2}\right)/4 = X(2)/4 = -\frac{1}{2}$$

Therefore, the trigonometric form of the Discrete Fourier Series is determined using Eqs. (6.4) and (6.6) as

$$x(n) = \frac{3}{2} - \cos\left(\frac{\pi}{2}n\right) - \sin\left(\frac{\pi}{2}n\right) - \frac{1}{2}\cos\pi n$$

6.2.1 Properties of Discrete Fourier Series

Linearity

Consider two periodic sequences $x_1(n)$ and $x_2(n)$ both with period N, such that

$$\text{DFS}\,[x_1(n)] = X_1(k) \text{ and}$$

$$\text{DFS}\,[x_2(n)] = X_2(k)$$

then, $$\text{DFS}\,[a_1x_1(n) + a_2x_2(n)] = a_1X_1(k) + a_2X_2(k)$$

Time Shifting

If $x(n)$ is a periodic sequence with N samples and

$$\text{DFS}\,[x(n)] = X(k),$$

then, $$\text{DFS}\,[x(n-m)] = e^{-j(2\pi/N)\,mk}X(k)$$

where $x(n - m)$ is a shifted version of $x(n)$.

Symmetry Property

We know that

$$\text{DFS}[x^*(n)] = X^*(-k) \text{ and DFS } [x^*(-n)] = X^*(k)$$

Therefore, $$\text{DFS}\{\text{Re}[x(n)]\} = \text{DFS}\left[\frac{x(n)+x^*(n)}{2}\right]$$

$$= \frac{1}{2}[X(k)+X^*(-k)] = X_e(k)$$

and $$\text{DFS}\{\text{Im}[x(n)]\} = \text{DFS}\left[\frac{x(n)-x^*(n)}{2}\right]$$

$$= \frac{1}{2}[X(k)-X^*(-k)] = X_0(k)$$

We know that, $x(n) = x_e(n) + x_0(n)$

where $$x_e(n) = \frac{1}{2}[x(n)+x^*(-n)]$$

and $$x_0(n) = \frac{1}{2}[x(n)-x^*(-n)]$$

Then,

$$\text{DFS}[x_e(n)] = \text{DFS}\left\{\frac{1}{2}[x(n)+x^*(-n)]\right\}$$

$$= \frac{1}{2}[X(k)+X^*(k)] = \text{Re}\,\{X(k)\}$$

and

$$\text{DFS}[x_o(n)] = \text{DFS}\left\{\frac{1}{2}[x(n) - x^*(-n)]\right\}$$

$$= \frac{1}{2}[X(k) - X^*(k)] = \text{Im}\{X(k)\}$$

Periodic Convolution

Let $x_1(n)$ and $x_2(n)$ be two periodic sequences with period N with

DFS $[x_1(n)] = X_1(k)$ and

DFS $[x_2(n)] = X_2(k)$

If $X_3(k) = X_1(k)\,X_2(k)$, then the periodic sequence $x_3(n)$ with Fourier series coefficients $X_3(k)$ is

$$x_3(n) = \sum_{m=0}^{N-1} x_1(m)x_2(n-m)$$

Hence, $\text{DFS}\left[\sum_{m=0}^{N-1} x_1(m)x_2(n-m)\right] = X_1(k)X_2(k)$

The properties of the Discrete Fourier Series are given in Table 6.1.

Table 6.1 *Properties of Discrete Fourier Series*

Property	*Periodic sequence (period N)*	*DFS coefficients*
Signal	$x(n)$	$X(k)$
Signal	$x_1(n), x_2(n)$	$X_1(k), X_2(k)$
Linearity	$ax_1(n) + bx_2(n)$	$aX_1(k) + bX_2(k)$
Time Shifting	$x(n+m)$	$W_N^{-km}X(k)$
Frequency shifting	$W_N^{ln}x(n)$	$X(k+l)$
Periodic convolution	$\sum_{m=0}^{N-1} x_1(m)x_2(n-m)$	$X_1(k)X_2(k)$
Multiplication	$x_1(n)x_2(n)$	$\frac{1}{N}\sum_{l=0}^{N-1} X_1(l)X_2(k-l)$
Symmetry Property	$x^*(n)$	$X^*(-k)$
	$x^*(-n)$	$X^*(k)$
	Re $\{x(n)\}$	$X_e(k) = \frac{1}{2}[X(k) + X^*(-k)]$
	Im $\{x(n)\}$	$X_o(k) = \frac{1}{2}[X(k) - X^*(-k)]$
	$x_e(n) = \frac{1}{2}[x(n) + x^*(-n)]$	Re $[X(k)]$
	$x_0(n) = \frac{1}{2}[x(n) - x^*(-n)]$	Im $\{X(k)\}$

(Contd.)

(Contd.)

	If $x(n)$ is real	$X(k) = X^*(-k)$ $\text{Re}\,[X(k)] = \text{Re}\,[X(-k)]$ $\text{Im}\,[X(k)] = -\text{Im}\,[X(-k)]$ $\lvert X(k)\rvert = \lvert X(-k)\rvert$ $\angle X(k) = \angle X(-k)$

DISCRETE-TIME FOURIER TRANSFORM (DTFT) 6.3

The **Discrete-Time Fourier Transform** (DTFT) or, simply, the Fourier transform of a discrete time sequence $x(n)$ is represented by the complex exponential sequence $[e^{-j\omega n}]$ where ω is the real frequency variable. This transform is useful to map the time-domain sequence into a continuous function of a frequency variable. The DTFT and the z-transform are applicable to any arbitrary sequences, whereas the DFT can be applied only to finite length sequences.

Definition

The discrete-time Fourier transform $X(e^{j\omega})$ of a sequence $x(n)$ is defined by

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \tag{6.10}$$

This equation represents the Fourier series representation of the periodic function $X(e^{j\omega})$. Hence, the Fourier coefficients $x(n)$ can be determined from $X(e^{j\omega})$ using the Fourier integral expressed by

$$x(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n}d\omega \tag{6.11}$$

called the Inverse Discrete-Time Fourier Transform (IDTFT).

Equations (6.10) and (6.11) are known as the Discrete-Time Fourier Transform pair for the sequence $x(n)$, which relate the time and frequency domain.

Discrete Fourier Transform (DFT)

The Discrete Fourier Transform (DFT) computes the values of the z-transform for evenly spaced points around the unit circle for a given sequence.

If the sequence to be represented is of finite duration, i.e. has only a finite number of non-zero values, the transform used is Discrete Fourier Transform (DFT). DFT finds its applications in digital signal processing including linear filtering, correlation analysis and spectrum analysis.

Definition

Let $x(n)$ be a finite duration sequence. The N-point DFT of the sequence $x(n)$ is expressed by

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi nk/N}, \qquad k = 0,1,\ldots N-1. \tag{6.12}$$

and the corresponding IDFT is

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

6.3.1 Relationship between the DFT and Other Transforms

Relationship to the Fourier Series Coefficients of a Periodic Sequence

The Fourier Series of a periodic sequence $x_p(n)$ with fundamental period N is given by

$$x_p(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi nk/N}, \quad -\infty < n < \infty$$

where the Fourier series coefficients are given by

$$c_k = \sum_{k=0}^{N-1} x_p(n) e^{-j2\pi nk/N}, \quad k = 0,\ldots,N-1.$$

By comparing the above equations with that of DFT pair and defining a sequence $x(n)$ which is identical to $x_p(n)$ over a single period, we get

$$X(k) = Nc_k$$

If a periodic sequence $x_p(n)$ is formed by periodically repeating $x(n)$ every N samples, i.e.

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n - lN)$$

The discrete frequency-domain representation is given by

$$X(k) = \sum_{n=0}^{N-1} x_p(n) e^{-j2\pi nk/N} = NC_k, \quad k = 0,1,\ldots,N-1.$$

and the IDFT is

$$x_p(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k) e^{j2\pi nk/N}, \quad -\infty < n < \infty$$

Relationship to the Spectrum of an Infinite Duration (Aperiodic) Signal

Let $x(n)$ be an aperiodic finite energy sequence. The Fourier transform is given by

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n}$$

If $X(e^{j\omega})$ is sampled at N equally spaced frequencies,

$$\omega_k = \frac{2\pi k}{N} = 0,1,2,\ldots,N-1$$

then $$X(k) = X(e^{j\omega})\Big|_{\omega = \frac{2\pi k}{N}} = \sum_{n=-\infty}^{\infty} x(n) e^{-j2\pi nk/N} \quad k = 0,1,\ldots,N-1.$$

The spectral components $\{X(k)\}$ correspond to the spectrum of a periodic sequence of period N, given by

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n - lN)$$

If $\hat{x}(n)$ is a finite duration sequence given by

$$\hat{x}(n) = \begin{cases} x_p(n), & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases}$$

then $\hat{x}(n)$ resembles the original sequence $x(n)$ only when the duration of the sequence $x(n)$, $L \le N$. In this case,

$$x(n) = \hat{x}(n), 0 \le n \le N-1$$

Otherwise, if $L > N$, then there will be no resemblance.

Relationship of the DFT to the z-transform

Let $X(z)$ be the z-transform for a sequence $x(n)$ which is given by $X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}$ with an ROC that includes the unit circle. If $X(z)$ is sampled at the N equally spaced points on the unit circle,

$z_k = e^{j2\pi k/N}$, $k = 0, 1, 2, \ldots N-1$, then

$$X(k) = X(z)\big|_{z=e^{j2\pi nk/N}}, \quad k = 0,1,\ldots N-1$$

$$= \sum_{n=-\infty}^{\infty} x(n)e^{-j2\pi nk/N}$$

This is identical to the Fourier transform $X(e^{j\omega})$ evaluated at the N equally spaced frequencies $\omega_k = 2\pi k/N, \quad k = 0, 1, \ldots, N-1.$

If the sequence $x(n)$ has a finite duration of length N, then the z-transform is given by

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

Substituting the IDFT relation for $x(n)$, we have

$$X(z) = \sum_{n=0}^{N-1}\left[\frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j2\pi nk/N}\right]z^{-n}$$

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

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

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

The above equation is identical to that of frequency sampling form. When this is evaluated over a unit circle, then

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

6.3.2 Properties of the DFT

The properties of the DFT are useful in the practical techniques for processing signals. The various properties are given below.

1. Periodicity

If $X(k)$ is an N-point DFT of $x(n)$, then

$x(n+N) = x(n)$ for all n

$X(k+N) = X(k)$ for all k

2. Linearity

If $X_1(k)$ and $X_2(k)$ are the N-point DFTs of $x_1(n)$ and $x_2(n)$ respectively, and a and b are arbitrary constants either real or complex-valued, then

$$ax_1(n) + bx_2(n) \xleftrightarrow[N]{DFT} aX_1(k) + bX_2(k)$$

3. Shifting Property

Let $x_p(n)$ is a periodic sequence with period N, which is obtained by extending $x(n)$ periodically, i.e.

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n - lN)$$

Now, shift the sequence $x_p(n)$ by k units to the right. Let the resultant signal be expressed as

$$X'_p(n) = x_p(n-k) = \sum_{l=-\infty}^{\infty} x(n-k-lN)$$

The finite duration sequence

$$X'(n) = \begin{cases} x'_p(n), & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases}$$

can be obtained from $x(n)$ by a circular shift.

The circular shift of a sequence can be represented by the index modulo N,

$$X'(n) = x(n - k, (\text{mod } N))$$

4. Convolution Theorem

If $\quad x_1(n) \xleftrightarrow{DFT} X_1(\omega)$

and $\quad x_2(n) \xleftrightarrow{DFT} X_2(\omega)$

then $\quad x(n) = x_1(n) * x_2(n) \xleftrightarrow{DFT} X(\omega) = X_1(\omega) X_2(\omega)$

Proof We know that the convolution formula is

$$x(n) = x_1(n) * x_2(n) = \sum_{n=-\infty}^{\omega} x_1(k) x_2(n-k)$$

Multiplying both sides of this equation by $e^{-j\omega n}$ and summing over all n, we get

$$X(\omega) = \sum_{n=-\infty}^{\omega} x(n) e^{-j\omega n} = \sum_{n=-\infty}^{\infty} \left[\sum_{n=-\infty}^{\infty} x_1(k) x_2(n-k) \right] e^{-j\omega n}$$

$$= X_1(\omega)\, X_2(\omega)$$

Hence, if two signals are converted in the time domain, then it is equivalent to multiplying their spectra in the frequency domain.

5. Time Reversal of a Sequence

If $\quad x(n) \xleftrightarrow[N]{DFT} X(k)$, then

$$x(-n, (\text{mod } N)) = x(N-n) \xleftrightarrow[N]{DFT} X(-k, (\text{mod } N)) = X(N-k)$$

Hence, when the N-point sequence in time is reversed, it is equivalent to reversing the DFT values.

6. Circular Time Shift

If $\quad x(n) \xleftrightarrow[N]{DFT} X(k)$, then

$$x(n-l, (\text{mod } N)) \xleftrightarrow[N]{DFT} X(k) e^{-j2\pi kl/N}$$

Shifting of the sequence by l units in the time-domain is equivalent to multiplication of $e^{-j2\pi kl/N}$ in the frequency-domain.

7. Circular Frequency Shift

If $\quad x(n) \xleftrightarrow[N]{DFT} X(k)$, then

$$x(n)e^{j2\pi ln/N} \xleftrightarrow[N]{DFT} X(k-l,(\text{mod } N))$$

Hence, when the sequence $x(n)$ is multiplied by the complex exponential sequence $e^{j2\pi ln/N}$, it is equivalent to circular shift of the DFT by l units in the frequency domain.

8. Complex Conjugate Property

If $x(n) \xleftrightarrow[N]{DFT} X(k)$, then

$$x^*(n) \xleftrightarrow[N]{DFT} X^*(-k,(\text{mod } N)) = X^*(N-k)$$

$$X^*(k) \xrightarrow{IDFT} \frac{1}{N}\sum_{k=0}^{N-1} X^*(k)e^{j2\pi kn/N} = \left[\frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j2\pi k(N-n)/N}\right]^*$$

Hence,

$$x^*(-n,(\text{mod } N)) = x^*(N-n) \xleftrightarrow[N]{DFT} X^*(k)$$

9. Circular Convolution

If $x_1(n) \xleftrightarrow[N]{DFT} X_1(k)$ and $x_2(n) \xleftrightarrow[N]{DFT} X_2(k)$, then

$$x_1(n) \circledN x_2(n) \xleftrightarrow[N]{DFT} X_1(k)\, X_2(k)$$

where $x_1(n) \circledN x_2(n)$ denotes the circular convolution of the sequence $x_1(n)$ and $x_2(n)$ defined as

$$x_3(n) = \sum_{m=0}^{N-1} x_1(m)x_2(n-m,(\text{mod } N))$$

$$= \sum_{m=0}^{N-1} x_2(m)x_1(n-m,(\text{mod } N))$$

10. Circular Correlation

For complex-valued sequences $x(n)$ and $y(n)$,

If $x(n) \xleftrightarrow[N]{DFT} X(k)$ and $y(n) \xleftrightarrow[N]{DFT} Y(k)$, then

$$r_{xy}(l) \xleftrightarrow[N]{DFT} R_{xy}(k) = X(k)Y^*(k)$$

where $r_{xy}(l)$ is the (unnormalised) circular cross-correlation sequence, given as

$$r_{xy}(l) = \sum_{n=0}^{N-1} x(n)y^*(n-l,(\text{mod } N))$$

11. Multiplication of Two Sequences

If $x_1(n) \xleftrightarrow[N]{DFT} X_1(k)$ and $x_2(n) \xleftrightarrow[N]{DFT} X_2(k)$, then

$$x_1(n)x_2(n) \xleftrightarrow[N]{DFT} \frac{1}{N} X_1(k) \circledN X_2(k)$$

12. Parseval's Theorem

For complex-valued sequences $x(n)$ and $y(n)$,

If $x(n) \xleftrightarrow[N]{DFT} X(k)$ and $y(n) \xleftrightarrow[N]{DFT} Y(k)$, then

$$\sum_{n=0}^{N-1} x(n)y^*(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)Y^*(k)$$

If $y(n) = x(n)$, then the above equation reduces to

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

This expression relates the energy in the finite duration sequence $x(n)$ to the power in the frequency components $X(k)$.

Example 6.2 *Find the DTFT of u(n).*

Solution Taking the DTFT of the sequence, we obtain

$$X\left(e^{j\omega}\right) = \sum_{n=0}^{\infty} x(n)e^{-j\omega n}$$

$$= \sum_{n=0}^{\infty} 1.\, e^{-j\omega n} = \frac{1}{1-e^{-j\omega}} \quad \left\{\text{since } \sum_{n=0}^{\infty} a^n = \frac{1}{1-a}\right\}$$

Example 6.3 *(a) Find the DTFT of the following finite duration sequence of length L*

$$x(n) = \begin{cases} A, & \text{for } 0 \le n \le L-1 \\ 0, & \text{otherwise} \end{cases}$$

(b) Also, find the inverse DTFT to verify $x(n)$ for $L = 3$ and $A = 1\,V$.

Solution

(a) Taking the DTFT of the sequence, we obtain

$$X(e^{j\omega}) = \sum_{n=0}^{L-1} x(n)e^{-j\omega n}$$

$$= A\sum_{n=0}^{L-1} e^{-j\omega n} = A\left[\frac{1-e^{-j\omega L}}{1-e^{-j\omega}}\right]$$

$$= A\left[\frac{e^{j\omega L/2}e^{-j\omega L/2} - e^{-j\omega L/2}e^{-j\omega L/2}}{e^{j\omega/2}e^{-j\omega/2} - e^{-j\omega/2}e^{-j\omega/2}}\right]$$

$$= A\left[\frac{(e^{-j\omega L/2})\left[e^{j\omega L/2} - e^{-j\omega L/2}\right]}{(e^{-j\omega/2})\left[e^{j\omega/2} - e^{-j\omega/2}\right]}\right] = Ae^{-j\omega}\left(\frac{L-1}{2}\right)\cdot\frac{\sin\left(\frac{\omega L}{2}\right)}{\sin\left(\frac{\omega}{2}\right)}$$

(b) The DTFT of $x(n)$ with $L = 3$ and $A = 1\,V$ is given by

$$X(e^{j\omega}) = A\sum_{n=0}^{L-1} e^{-j\omega n} = 1 + e^{-j\omega} + e^{-j2\omega}$$

The inverse DTFT of $X(e^{j\omega})$ is

$$x(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)e^{j\omega n}\,d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left\{1 + e^{-j\omega} + e^{-j2\omega}\right\}e^{j\omega n}\,d\omega$$

$$= \frac{1}{2\pi}\left[\frac{e^{jn\omega}}{jn} + \frac{e^{j(n-1)\omega}}{j(n-1)} + \frac{e^{j(n-2)\omega}}{j(n-2)}\right]_{-\pi}^{\pi}$$

$$= \frac{1}{2\pi}\left[\frac{2\sin n\pi}{n} + \frac{2\sin\left((n-1)\pi\right)}{(n-1)} + \frac{2\sin\left((n-2)\pi\right)}{(n-2)}\right]$$

For $n = 0$

$$x(0) = \frac{1}{2\pi}\left[\frac{2\sin n\pi}{n}\right] = \frac{\sin n\pi}{n\pi} = 1 \left[\text{L}'\text{ Hospital's rule gives } \lim_{n\to 0}\frac{\sin n\pi}{n\pi} = 1\right]$$

Similarly,

For $n = 1 : x(1) = 1$
For $n = 2 : x(2) = 1$
For $n \geq 3 : x(n) = 0$

These values are identical to the defined sequence for $L = 3$ and $A = 1\text{V}$.

i.e. $x(n) = \begin{cases} 1, & \text{for } 0 \leq n \leq 2 \\ 0, & \text{otherwise} \end{cases}$

Example 6.4 *Find the Fourier transform of the following signals.*

(a) $x(n) = (\alpha^n \sin \omega_0 n)u(n)$, $|a| < 1$ *(b)* $x(n) = (1/4)^n u(n + 4)$

Solution

(a) Given $x(n) = (\alpha^n \sin \omega_0 n)u(n)$, $|a| < 1$

We know that $Z(\sin \omega_0 n) = \dfrac{z \sin \omega_0 n}{z^2 - 2z\cos\omega_0 + 1} = \dfrac{z^{-1}\sin \omega_0}{1 - 2z^{-1}\cos\omega_0 + z^{-2}}$

Using the scaling property, we obtain

$$X(z) = Z(\alpha^n \sin \omega_0 n) = \frac{\alpha z^{-1}\sin\omega_0}{1 - 2\alpha z^{-1}\cos\omega_0 + \alpha^2 z^{-2}} = \frac{z\alpha\sin\omega_0}{z^2 - 2z\alpha\cos\omega_0 + \alpha^2}$$

Substituting $z = e^{j\omega}$, we get the Discrete-Time Fourier Transform (DTFT) as

$$X(e^{j\omega}) = \frac{e^{j\omega}\alpha\sin\omega_0}{e^{j2\omega} - 2e^{j\omega}\alpha\cos\omega_0 + \alpha^2}$$

Note: Substituting $z = e^{j2\pi k/N}$, we get the Discrete Fourier Transform (DFT) as

$$X(k) = \frac{e^{j2\pi k/N}\alpha\sin\omega_0}{e^{j4\pi k/N} - 2e^{j2\pi k/N}\alpha\cos\omega_0 + \alpha^2}$$

(b) Given $x(n) = (1/4)^n\, u(n + 4)$

$$Z\left[\left(\frac{1}{4}\right)^n\right] = \frac{1}{1 - \frac{1}{4}z^{-1}}$$

$$X(z) = Z\left[\left(\frac{1}{4}\right)^n u(n+4)\right] = \frac{1}{1 - \frac{1}{4}z^{-1}} z^4 = \frac{z^5}{\left(z - \frac{1}{4}\right)}$$

Substituting $z = e^{j\omega}$, we get the Discrete-Time Fourier Transform (DTFT) as

$$X(e^{j\omega}) = \frac{e^{j5\omega}}{e^{j\omega} - \frac{1}{4}}$$

Note: Substituting $z = e^{j2\pi k/N}$, we get the Discrete Fourier Transform (DFT) as

$$X(k) = \frac{e^{j10\pi k/N}}{\left(e^{j2\pi k/N} - \frac{1}{4}\right)}$$

Example 6.5 *Prove (a) Re* $[X(k)] = Re\ [X(-k)]$, *(b)* $|X(k)| = |X(-k)|$ *for a real periodic sequence x(n).*

Solution

(a) To prove that $Re\ [X(k)] = Re\ [X(-k)]$:

The discrete Fourier transform of sequence $x(n)$ is given by

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

$$= \sum_{n=0}^{N-1} x(n) \left\{ \cos\left[\frac{2\pi nk}{N}\right] - j \sin\left[\frac{2\pi nk}{N}\right] \right\}$$

$$= \sum_{n=0}^{N-1} x(n) \cos\left[\frac{2\pi nk}{N}\right] - j \sum_{n=0}^{N-1} x(n) \sin\left[\frac{2\pi nk}{N}\right]$$

Then, the real part of $X(k)$ is

$$\text{Re}\left[X(k)\right] = \sum_{n=0}^{N-1} x(n) \cos\left[\frac{2\pi nk}{N}\right] \tag{2}$$

Now, we change the variable k by $-k$ in Eq. (1)

$$X(-k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi n}{N}(-k)} \tag{3}$$

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

$$= \sum_{n=0}^{N-1} x(n) \left\{ \cos\left(\frac{2\pi nk}{N}\right) + j \sin\left(\frac{2\pi nk}{N}\right) \right\}$$

$$= \sum_{n=0}^{N-1} x(n) \cos\left(\frac{2\pi nk}{N}\right) + j \sum_{n=0}^{N-1} x(n) \sin\left(\frac{2\pi nk}{N}\right)$$

Then, the real part of $X(-k)$ is given by

$$Re\left[X(-k)\right] = \sum_{n=0}^{N-1} x(n) \cos\left(\frac{2\pi nk}{N}\right) \tag{4}$$

Comparing Eqs. (2) and (4), we obtain

$$\text{Re}\,[X(k)] = Re\,[X(-k)]$$

(b) To prove that $|X(k)| = |X(-k)|$

We know that $X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi nk}{N}}$

$$|X(k)| = \left| \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi nk}{N}} \right| \le \sum_{n=0}^{N-1} |x(n)| \left| e^{-j\frac{2\pi nk}{N}} \right|$$

$$\le \sum_{n=0}^{N-1} |x(n)|, \text{ since } \left| e^{-j\frac{2\pi nk}{N}} \right| = 1 \qquad (5)$$

Also,

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

$$|X(-k)| = \left| \sum_{n=0}^{N-1} x(n) e^{j\frac{2\pi nk}{N}} \right| \le \sum_{n=0}^{N-1} |x(n)| \left| e^{j\frac{2\pi nk}{N}} \right|$$

$$\le \sum_{n=0}^{N-1} |x(n)|, \quad \text{since } \left| e^{j\frac{2\pi nk}{N}} \right| = 1 \qquad (6)$$

Comparing Eqs. (5) and (6), we obtain $|X(k)| = |X(-k)|$.

Example 6.6 *Determine the DFT of the sequence*

$$x(n) = \begin{cases} \frac{1}{4}, & \text{for } 0 \le n \le 2 \\ 0, & \text{otherwise} \end{cases}$$

Solution The N-point DFT of the sequence $x(n)$ is defined as

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

$$x(n) = \left\{ \frac{1}{4}, \frac{1}{4}, \frac{1}{4} \right\}$$

Therefore, $X(k) = \frac{1}{4}\left[1 + e^{-j\omega} + e^{-j2\omega}\right]\Big|_{\omega=\frac{2\pi k}{N}}$

$$= \frac{1}{4} e^{-j\omega}[1 + 2\cos\omega], \text{ where } \omega = \frac{2\pi k}{N} \text{ and } N = 3$$

$$= \frac{1}{4} e^{-j2\pi k/3} = \left[1 + 2\cos\frac{2\pi k}{3}\right]$$

Hence, $X(k) = \frac{1}{4} e^{-j2\pi nk/3} = \left[1 + 2\cos\left(\frac{2\pi k}{3}\right)\right]$, where $k = 0, 1, \ldots, N-1$

Example 6.7 *Determine the DFT of the sequence*

$$x(n) = \begin{cases} \frac{1}{5}, & for \quad -1 \le n \le 1 \\ 0, & otherwise \end{cases}$$

Solution $x(n) = \left\{\frac{1}{5}, \underset{\uparrow}{\frac{1}{5}}, \frac{1}{5}\right\}$

We know that, $X(k) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}$ at $\omega = \frac{2\pi k}{N}$, $\quad k = 0, 1, ..., N-1$

$$X(k) = \frac{1}{5}\left[e^{j\omega} + 1 + e^{-j\omega}\right]_{\text{at } \omega = \frac{2\pi k}{N}}$$

$$= \frac{1}{5}[1 + 2\cos\omega] \quad \text{where } \omega = \frac{2\pi k}{N} \text{ and } N = 3 = \frac{1}{5}\left[1 + 2\cos\left(\frac{2\pi k}{3}\right)\right]$$

Hence, $X(k) = \frac{1}{5}\left[1 + 2\cos\left(\frac{2\pi k}{3}\right)\right]$ where $k = 0, 1, \ldots, N-1$.

Example 6.8 *Derive the DFT of the sample data sequence* $x(n) = \{1, 1, 2, 2, 3, 3\}$ *and compute the corresponding amplitude and phase spectrum.*

Solution The N-point DFT of a finite duration sequence $x(n)$ is defined as

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi nk/N}, \quad k = 0, 1, ..., N-1.$$

For $k = 0$

$$X(0) = \sum_{n=0}^{5} x(n)e^{-j2\pi(0)n/6} = \sum_{n=0}^{5} x(n) = 1+1+2+2+3+3 = 12$$

For $k = 1$

$$X(1) = \sum_{n=0}^{5} x(n)e^{-j2\pi(1)n/6}$$

$$= \sum_{n=0}^{5} x(n)e^{-j\pi n/3}$$

$$= 1 + e^{-j\pi/3} + 2\,e^{-j2\pi/3} + 2\,e^{-j\pi} + 3\,e^{-j4\pi/3} + 3\,e^{-j5\pi/3}$$

$$= 1 + 0.5 - j0.866 + 2(-0.5 - j0.866) + 2(-1) + 3(-0.5 + j0.866) + 3(0.5 + j0.866)$$

$$= -1.5 + j2.598$$

For $k = 2$

$$X(2) = \sum_{n=0}^{5} x(n)e^{-j2\pi(2)n/6}$$

$$= \sum_{n=0}^{5} x(n)e^{-2j\pi n/3}$$

$$= 1 + e^{-j2\pi/3} + 2\,e^{-j4\pi/3} + 2e^{-j2\pi} + 3e^{-j8\pi/3} + 3e^{-j10\pi/3}$$
$$= 1 + (-0.5) - j0.866 + 2(-0.5 + j0.866) + 2(1)$$
$$+ 3(-0.5 - j0.866) + 3(-0.5 + j0.866) = -1.5 + j0.866$$

For $k = 3$

$$X(3) = \sum_{n=0}^{5} x(n)e^{-j2\pi(3)n/6}$$
$$= \sum_{n=0}^{5} x(n)e^{-j\pi n}$$
$$= 1 + e^{-j\pi} + 2e^{-j2\pi} + 2e^{-j3\pi} + 3e^{-j4\pi} + 3e^{-j5\pi}$$
$$= 1 - 1 + 2(1) + 2(-1) + 3(1) + 3(-1) = 0$$

For $k = 4$

$$X(4) = \sum_{n=0}^{5} x(n)\,e^{-j2\pi(4)n/6}$$
$$= \sum_{n=0}^{5} x(n)e^{-j4\pi n/3}$$
$$= 1 + e^{-j4\pi/3} + 2e^{-j8\pi/3} + 2e^{-j4\pi} + 3e^{-j16\pi/3} + 3e^{-j20\pi/3}$$
$$= 1 + (-0.5 + j0.866) + 2(-0.5 - j0.866) + 2(1)$$
$$+ 3(-0.5 + j0.866) + 3(-0.5 - j0.866) = -1.5 - j0.866$$

For $k = 5$

$$X(5) = \sum_{n=0}^{5} x(n)\,e^{-j2\pi(5)n/6}$$
$$= \sum_{n=0}^{5} x(n)e^{-j5\pi n/3}$$
$$= 1 + e^{-j5\pi/3} + 2e^{-j10\pi/3} + 2e^{-j5\pi} + 3e^{-j20\pi/3} + 3e^{-j25\pi/3}$$
$$= 1 + (-0.5 + j\,0.866) + 2(-0.5 + j\,0.866) + 2(-1)$$
$$+ 3(-0.5 - j\,0.866) + 3(0.5 - j\,0.866)$$
$$= -1.5 - j2.598$$
$$X(k) = \{12, -1.5 + j2.598, -1.5 + j0.866, 0, -1.5 - j\,0.866, -1.5 - j\,2.598\}$$

The corresponding amplitude spectrum is given by

$$|X(k)| = \Big\{\sqrt{12 \times 12}, \sqrt{(-1.5)^2 + (-2.598)^2}, \sqrt{(-1.5)^2 + (0.866)^2}, 0,$$
$$\sqrt{(-1.5)^2 + (-0.866)^2}, \sqrt{(-1.5)^2 + (-2.598)^2}\Big\}$$
$$= \{\underset{\uparrow}{12},\ 2.999,\ 1.732,\ 0,\ 1.732,\ 2.999\}$$

and the corresponding phase spectrum is given by

$$\angle X(k) = \left\{\underset{\uparrow}{\tan^{-1}}(0),\ \tan^{-1}\left(\frac{2.598}{-1.5}\right),\ \tan^{-1}\left(\frac{0.866}{-1.5}\right),\ \tan^{-1}(0), \tan^{-1}\left(\frac{-0.866}{-1.5}\right),\ \tan^{-1}\left(\frac{-2.598}{-1.5}\right)\right\}$$

$$=\left\{\underset{\uparrow}{0}, -\frac{\pi}{3}, -\frac{\pi}{6}, 0, \frac{\pi}{6}, \frac{\pi}{3}\right\}$$

Example 6.9 *Compute the discrete Fourier transform of each of the following finite length sequences considered to be of length N.*

(a) $x(n) = \delta(n)$ (b) $x(n) = \delta(n - n_0)$ *where* $0 < n_0 < N$

Solution

(a) Given $x(n) = \delta(n)$

From the definition, the discrete Fourier transform of the finite sequence of length N is

$$X(e^{j\omega}) = \sum_{n=0}^{N-1} x(n)e^{-j\omega n}$$

Substituting $x(n) = \delta(n)$, we get

$$X(e^{j\omega}) = \sum_{n=0}^{N-1} \delta(n)e^{-j\omega n} = 1$$

(b) Given $x(n) = \delta(n - n_o)$ where $0 < n_o < N - 1$

The Fourier transform of the finite sequence of length N is

$$X(e^{j\omega}) = \sum_{n=0}^{N-1} \delta(n-n_o)e^{-j2\pi nk/N} = e^{-j\omega n_o}$$

Example 6.10 *Find the N-Point DFT for* $x(n) = a^n$ *for* $0 < a < 1$.

Solution The N-point DFT is defined as

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

$$= \sum_{n=0}^{N-1} a^n e^{-j2\pi nk/N} = \sum_{n=0}^{N-1} (ae^{-j2\pi k/N})^n = \frac{1-(ae^{-j2\pi k/N})N}{1-ae^{-j2\pi k/N}}$$

$$X(k) = \frac{1-a^N}{1-ae^{-j2\pi k/N}}, \quad k = 0,1,\ldots,N-1$$

Example 6.11 *Find the 4-point DFT of the sequence* $x(n) = \cos\frac{n\pi}{4}$.

Solution Given $N = 4$,

$$x(n) = [\cos(0), \cos(\pi/4), \cos(\pi/2), \cos(3\pi/4)] = \{1, 0.707, 0, -0.707\}$$

The N-point DFT of the sequence $x(n)$ is defined as

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

The DFT is

$$X(k) = \sum_{n=0}^{3} x(n)e^{-j2\pi nk/4}, \quad k = 0, 1, 2, 3$$

$$= \sum_{n=0}^{3} x(n)e^{-j\pi nk/2}, \quad k = 0, 1, 2, 3$$

For $k = 0$

$$X(0) = \sum_{n=0}^{3} x(n) = 1$$

For $k = 1$

$$X(1) = \sum_{n=0}^{3} x(n)e^{-j\pi(1)n/2}$$

$$= 1 + 0.707\ e^{-j\,\pi/2} + 0 + (-0.707)e^{-j\,3\pi/2}$$
$$= 1 + (0.707)\ (-j) + 0 - (0.707)\ (j) = 1 - j\ 1.414$$

For $k = 2$

$$X(2) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\pi(2)n/2} = \sum_{n=-\infty}^{\infty} x(n)e^{-j\pi n}$$

$$= 1 + (0.707)\ e^{-j\pi} + 0 + (-0.707)\ e^{-j3\pi}$$
$$= 1 + (0.707)\ (-1) + 0 + (-0.707)\ (-1) = 1$$

For $k = 3$

$$X(3) = \sum_{n=0}^{3} x(n)e^{-j\pi(3)n/2}$$

$$= 1 + (0.707)\ e^{-j3\pi/2} + 0 + (-0.707)\ e^{-j9\pi/2}$$
$$= 1 + (0.707)\ (j) + 0 + (-\ 0.707)\ (-j) = 1 + j\ 1.414$$

$$X(k) = \{\ 1,\ 1 - j\ 1.414,\ 1,\ 1 + j\ 1.414\}$$

Example 6.12 *Find the inverse DFT of X(k)* = {1, 2, 3, 4}.

Solution The inverse DFT is defined as $x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j2\pi nk/N}, \quad n = 0, 1, 2, 3, ..., N-1$

Given $N = 4$, $\quad x(n) = \frac{1}{4}\sum_{k=0}^{3} X(k)e^{j2\pi nk/N}, \quad n = 0,1,2,3$

When $n = 0$

$$x(0) = \frac{1}{4}\sum_{k=0}^{3} X(k)e^{j\pi(0)k/2} = \frac{1}{4}\left(1+2+3+4\right) = \frac{5}{2}$$

When $n = 1$

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

$$= \frac{1}{4}(1 + 2e^{j\pi/2} + 3e^{j\pi} + 4e^{j3\pi/2})$$

$$= \frac{1}{4}(1 + 2(j) + 3(-1) + 4(-j)) = \frac{1}{4}(-2 - j2) = -\frac{1}{2} - j\frac{1}{2}$$

When $n = 2$

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

$$= \frac{1}{4}(1 + 2e^{j\pi} + 3e^{j2\pi} + 4e^{j3\pi}) = \frac{1}{4}(1 + 2(-1) + 3(1) + 4(-1)) = \frac{1}{4}(-2) = -1/2$$

When $n=3$

$$x(3)=\frac{1}{4}\sum_{k=0}^{3}X(k)e^{j3\pi k/2}$$

$$=\frac{1}{4}(1+2^{j3\pi/2}+3e^{j3\pi}+4e^{j9\pi/2})$$

$$=\frac{1}{4}(1+2(-j)+3(-1)+4j)=\frac{1}{4}(-2+2j)=-\frac{1}{2}+j\frac{1}{2}$$

$$x(n)=\left\{\frac{5}{2},-\frac{1}{2}-j\frac{1}{2},-\frac{1}{2},-\frac{1}{2}+j\frac{1}{2}\right\}$$

Example 6.13 *Determine the IDFT of* $X(k)=\{3,(2+j),1,(2-j)\}$.

Solution The IDFT is defined as $x(n)=\frac{1}{N}\sum_{k=0}^{N-1}X(k)e^{j2\pi k/N}, \quad 0\le n\le N-1$

Given $N=4$, $x(n)=\frac{1}{4}\sum_{k=0}^{3}X(k)e^{j\pi nk/2}, \quad 0\le n\le 3$

When $n=0$

$$x(0)=\frac{1}{4}\sum_{k=0}^{3}X(k)e^{0}=\frac{1}{4}\big[3+(2+j)+1+(2-j)\big]=2$$

When $n=1$

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

$$=\frac{1}{4}\left[3+(2+j)e^{j\pi/2}+e^{j\pi}+(2-j)e^{j3\pi/2}\right]=\frac{1}{4}[3+(2+j)j-1+(2-j)(-j)]=0$$

When $n=2$

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

$$=\frac{1}{4}[3+(2+j)e^{j\pi}+e^{j2\pi}+(2-j)e^{j3\pi}]=\frac{1}{4}[3+(2+j)(-1)+1+(2-j)(-1)]=0$$

When $n=3$

$$x(3)=\frac{1}{4}\sum_{k=0}^{3}X(k)e^{j3\pi k/2}$$

$$=\frac{1}{4}[3+(2+j)e^{j3\pi/2}+e^{j3\pi}+(2-j)e^{j9\pi/2}]$$

$$=\frac{1}{4}[3+(2+j)(-j)-1+(2-j)(j)]=1$$

Therefore, the IDFT of the given DFT produces the following original sequence values

$$x(n)=\{2,0,0,1\}$$

6.3.3 Circular Convolution (Periodic Convolution)

Consider the two sequences $x_1(n)$ and $x_2(n)$ which are of finite duration. Let $X_1(k)$ and $X_2(k)$ be the N-point DFTs of the two sequences respectively and they are given by

$$X_1(k) = \sum_{n=0}^{N-1} x_1(n)e^{-j2\pi nk/N}, \quad k = 0,1,\ldots,N-1.$$

$$X_2(k) = \sum_{n=0}^{N-1} x_2(n)e^{-j2\pi nk/N}, \quad k = 0,1,\ldots,N-1. \tag{6.13}$$

Let $x_3(n)$ be another sequence of length N and its N-point DFT be $X_3(k)$ which is a product of $X_1(k)$ and $X_2(k)$, i.e.

$$X_3(k) = X_1(k)\, X_2(k),\ k = 0, 1,..., N-1.$$

The sequence $x_3(n)$ can be obtained by taking the inverse DFT of $X_3(k)$, i.e.

$$\begin{aligned} x_3(m) &= IDFT[X_3(k)] \\ &= \frac{1}{N}\sum_{k=0}^{N-1} X_3(k)e^{j2\pi mk/N} = \frac{1}{N}\sum_{k=0}^{N-1} X_1(k)X_2(k)e^{j2\pi mk/N} \\ &= \frac{1}{N}\sum_{k=0}^{N-1}\left[\sum_{n=0}^{N-1} x_1(n)e^{-j2\pi nk/N}\right]\left[\sum_{l=0}^{N-1} x_2(l)e^{-j2\pi lk/N}\right]e^{j2\pi mk/N} \\ &= \frac{1}{N}\sum_{n=0}^{N-1} x_1(n)\sum_{l=0}^{N-1} x_2(l)\left[\sum_{k=0}^{N-1} e^{j2\pi k(m-n-l)/N}\right] \end{aligned}$$

Consider the term within the brackets. It has the form

$$\sum_{k=0}^{N-1} a^k = \begin{cases} N, & \text{for } a = 1 \\ \dfrac{1-a^N}{1-a}, & \text{for } a \neq 1 \end{cases}$$

where, $a = e^{j2\pi(m-n-l)/N}$. When $(m - n - l)$ is a multiple of N, then $a = 1$, otherwise $a^N = 1$ for any value of $a \neq 0$.

Therefore, $$\sum_{k=0}^{N-1} a^k = \begin{cases} N, & l = m-n+pN, N = (m-n)(\text{mod } N), p: \text{integer}) \\ 0, & \text{otherwise} \end{cases}$$

$x_3(m)$ becomes

$$x_3(m) = \sum_{n=0}^{N-1} x_1(n)\, x_2(m-n, (\text{mod } N)),\ m = 0,1,\ldots,N-1 \tag{6.14}$$

where $x_2((m - n) \bmod N)$ is the reflected and circularly shifted version of $x_2(m)$ and n represents the number of indices that the sequence $x(n)$ is shifted to the right.

The above equation has a form similar to the convolution sum and it is called circular convolution. The circularly shifted versions of $x(m)$ are given below:

$$x(m) = [x(0), x(1)... x(N-3), x(N-2), x(N-1)]$$

$$x(m-1, (\text{mod } N)) = [x(N-1), x(0), x(1)... x(N-3), x(N-2)]$$

$$x(m-2, (\text{mod } N)) = [x(N-2), x(N-1), x(0), x(1)... x(N-3)]$$

$$\vdots$$

$$x(m-n, (\text{mod } N)) = [x(N-n), x(N-n+1),..., x(N-n-1)]$$

$$\vdots$$

$$x(m-N, (\text{mod } N)) = [x(0), x(1)...x(N-3), x(N-2), x(N-1)]$$

It is noted that a shift of $n = N$ results in the original signal $x(m)$.

Matrix Multiplication Method

The circular convolution of two sequences $x(n)$ and $h(n)$ can be obtained by representing these sequences in the matrix form as given below:

$$\begin{bmatrix} x(0) & x(N-1) & x(N-2)\ldots x(2) & x(1) \\ x(1) & x(0) & x(N-1)\ldots x(3) & x(2) \\ x(2) & x(1) & x(0)\ \ldots x(4) & x(3) \\ \cdot & \cdot & \cdot\ \ \ldots\ \cdot & \cdot \\ \cdot & \cdot & \cdot\ \ \ldots\ \cdot & \cdot \\ \cdot & \cdot & \cdot\ \ \ldots\ \cdot & \cdot \\ x(N-2) & x(N-3) & x(N-4)\ldots x(0) & x(N-1) \\ x(N-1) & x(N-2) & x(N-3)\ \ldots\ x(1) & x(0) \end{bmatrix} \begin{bmatrix} h(0) \\ h(1) \\ h(2) \\ \cdot \\ \cdot \\ \cdot \\ h(N-2) \\ h(N-1) \end{bmatrix} = \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ \cdot \\ \cdot \\ \cdot \\ y(N-2) \\ y(N-1) \end{bmatrix}$$

The sequence $x(n)$ is repeated via circular path shift of samples and represented in $N \times N$ matrix form. The sequence $h(n)$ is represented as column matrix. The multiplication of these two matrices gives the sequence $y(n)$.

6.3.4 Linear Convolution vs Circular Convolution

Consider a finite duration sequence $x(n)$ of length N_1 which is given a input to an FIR system with impulse response $h(n)$ of length N_2, then the output is given by

$$y(n) = x(n) * h(n)$$

$$= \sum_{k=0}^{N_1-1} x(k)h(n-k), \quad n = 0,1,...,N_1 + N_2 - 1$$

or
$$y(n) = \sum_{k=0}^{N_2-1} h(k)x(n-k), \quad n = 0,1,...,N_1 + N_2 - 1$$

where, $x(n) = 0,\ n < 0$ and $n \geq N_1$

$h(n) = 0,\ n < 0$ and $n \geq N_2$

In the frequency-domain,

$$Y(\omega) = H(\omega)\, X(\omega)$$

If the sequence $y(n)$ has to be represented uniquely in the frequency-domain by samples of its spectrum $Y(\omega)$, then

$$Y(k) = Y(\omega)|_{\omega = 2\pi k/N},\ k = 0, 1, ..., N-1$$

$$= X(\omega)H(\omega)|_{\omega = 2\pi k/N},\ k = 0, 1, ..., N-1$$

$$Y(k) = X(k)\, H(k) \tag{6.15}$$

where $X(k)$ and $H(k)$ are the N-point DFTs of the sequence $x(n)$ and $h(n)$ respectively.

Since $y(n)$ can be obtained by

$$y(n) = \text{IDFT}\ \{X(k)\ H(k)\} \tag{6.16}$$

the N-point circular convolution of $x(n)$ with $h(n)$ must be equivalent to the linear convolution of $x(n)$ with $h(n)$.

In general, the linear convolution of two sequences $x(n)$ and $h(n)$ with lengths N_1 and N_2 respectively will give a sequence with a length of $N \geq N_1 + N_2 - 1$. When $x(n)$ and $h(n)$ have a duration less than N, for implementing linear convolution using circle convolution, $N_2 - 1$ and $N_1 - 1$ zeros are padded (added) at the end of $x(n)$ and $h(n)$ respectively to increase their length to N.

Example 6.14 *Let* $x_1(n) = x_2(n) = \begin{cases} 1, & 0 \leq n \leq N-1 \\ 0, & \textit{otherwise} \end{cases}$

Find out the circular convolution of this sequence.

Solution Here $x_1(n)$ and $x_2(n)$ are the given N-point sequences. The circular convolution of $x_1(n)$ and $x_2(n)$ yields another N-point sequences $x_3(n)$. In this method, an $(N \times N)$ matrix is formed using one of the sequences. Another sequence is arranged as an $(N \times 1)$ column matrix. The product of these two matrices gives the resultant sequences $x_3(n)$ as shown below.

Using matrix approach, we can formulate circular convolution as

$$\begin{pmatrix} x_1(0) & x_1(N-1) & x_1(N-2) & \ldots & x_1(1) \\ x_1(1) & x_1(0) & x_1(N-1) & \ldots & x_2(1) \\ \vdots & \vdots & \vdots & & \vdots \\ x_1(N-2) & x_1(N-3) & x_1(N-4) & \ldots & x_1(N-1) \\ x_1(N-1) & x_1(N-2) & x_1(N-3) & \ldots & x_1(0) \end{pmatrix} \begin{pmatrix} x_2(0) \\ x_2(1) \\ \vdots \\ x_2(N-2) \\ x_2(N-1) \end{pmatrix} = \begin{pmatrix} x_3(0) \\ x_3(1) \\ \vdots \\ x_3(N-2) \\ x_2(N-1) \end{pmatrix}$$

For the given sequences, the circular convolution $x_3(n)$ is determined by

$$\begin{pmatrix} 1 & 1 & 1 & . & . & . & . & 1 \\ 1 & 1 & 1 & . & . & . & . & 1 \\ . & . & . & & & & & . \\ . & . & . & & & & & . \\ . & . & . & & & & & . \\ 1 & 1 & 1 & & & & & 1 \\ 1 & 1 & 1 & . & . & . & . & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ . \\ . \\ . \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} N \\ N \\ . \\ . \\ . \\ . \\ N \end{pmatrix}$$

Hence, $x_3(n) = (N, N, \ldots N, N)$

Example 6.15 *Compute* (a) *linear and* (b) *circular periodic convolutions of the two sequences* $x_1(n) = \{1, 1, 2, 2\}$ *and* $x_2(n) = \{1, 2, 3, 4\}$. (c) *Also find circular convolution using the DFT and IDFT.*

$x_2(n)$ \ $x_1(n)$	1	1	2	2
1	1	1	2	2
2	2	2	4	4
3	3	3	6	6
4	4	4	8	8

Solution

(a) Using matrix representation given as follows, the linear convolution of the two sequences can be determined.

Hence, $x_3(n) = x_1(n) * x_2(n) = \{1, 3, 7, 13, 14, 14, 8\}$

(b) Similarly, circular convolution of the two sequences is shown in Fig. E6.15.

$$x_3(m) = \sum_{n=0}^{N-1} x_1(n) x_2\ (m-n,\ (\text{mod}\ N)),\ m = 0,\ 1,\ ...,\ N-1$$

For $m = 0$

$$x_3(0) = \sum_{n=0}^{3} x_1(n)\ x_2(-n,\ (\text{mod}\ 4))$$

$x_2(-n$, (mod 4)) is the sequence $x_2(n)$ folded. The folded sequence is obtained by plotting $x_2(n)$ in a clockwise direction.

$$x_2(0, (\text{mod}\ 4)) = x_2(0)$$
$$x_2(-1, (\text{mod}\ 4)) = x_2(3)$$
$$x_2(-2, (\text{mod}\ 4)) = x_2(2)$$
$$x_2(-3, (\text{mod}\ 4)) = x_2(1)$$

$x_3(0)$ is obtained by computing the product sequence, i.e. multiplying the sequences $x_1(n)$ and $x_2(-n$, (mod 4)), point by point and taking the sum, we get $x_3(0) = 15$.

For $m = 1$:

$$x_3(1) = \sum_{n=0}^{3} x_1(n)\ x_2(1 - n,\ (\text{mod}\ 4))$$

$x_2(1 - n$, (mod 4)) is the sequence $x_2(-n$, (mod 4)) rotated counterclockwise by one unit in time. From the product sequence, the sum is $x_3(1) = 17$.

For $m = 2$:

$$x_3(2) = \sum_{n=0}^{3} x_1(n)\ x_2(2 - n,\ (\text{mod}\ 4))$$

$x_2(2 - n$, (mod 4)) is the sequence $x_2(-n$, (mod 4)) rotated counterclockwise by two units in time. From the product sequence, the sum is $x_3(2) = 15$.

For $m = 3$:

$$x_3(3) = \sum_{n=0}^{3} x_1(n)\ x_2(3 - n,\ (\text{mod}\ 4))$$

$x_2(3 - n$, (mod 4)) is the sequence $x_2(-n$, (mod 4)) rotated counterclockwise by three units in time. From the product sequence, the sum is $x_3(3) = 13$.

Hence, the circular convolution of the two sequences $x_1(n)$ and $x_2(n)$ is

$$x_3(n) = \{15, 17, 15, 13\}$$

(c) Here the DFT and IDFT are used for finding circular convolution

$$X_3(k) = X_1(k)\ X_2(k)$$

and

$$x_3(n) = \frac{1}{N} \sum_{k=0}^{N-1} X_3(k) e^{j2\pi k/N},\ \ n = 0, 1, ...,\ N-1$$

(i) When $x_1(n) = [1, 1, 2, 2]$

$$X_1(k) = \sum_{n=0}^{N-1} x_1(n) e^{-j2\pi nk/N},\ k = 0, 1, ...,\ N-1$$

For $N = 4$

$$X_1(k) = \sum_{n=0}^{3} x_1(n)e^{-j2\pi nk/4}, \quad k = 0, 1, 2, 3$$

For $k = 0$

$$X_1\ (0) = \sum_{n=0}^{3} x_1(n)e^{-j2\pi n(0)/4} = 6$$

For $k = 1$

$$X_1(1) = \sum_{n=0}^{3} x_1(n)e^{-j\pi n/2}$$

$$= 1 + e^{-j\pi/2} + 2e^{-j\pi} + 2e^{-j3\pi/2} = 1 - j + 2\ (-1) + 2\ (j) = -1 + j$$

For $k = 2$

$$X_1(2) = \sum_{n=0}^{3} x_1(n)e^{-jn\pi}$$

$$= 1 + e^{-j\pi} + 2e^{-2\pi} + 2e^{-j3\pi} = 1 - 1 + 2\ (1) + 2\ (-1) = 0$$

For $k = 3$

$$X_1(1) = \sum_{n=0}^{3} x_1(n)e^{-j3n\pi/2}$$

$$= 1 + e^{-j3\pi/2} + 2e^{-j3\pi} + 2e^{-j9\pi/2} = 1 + (j\ ) + 2\ (-1) - j\ (2) = -1 - j$$

$$X_1(k) = \{6, -1 + j, 0, -1 - j\}$$

(ii) When $x_2(n) = \{1, 2, 3, 4\}$

Here, $N = 4$. Therefore,

$$X_2(k) = \sum_{n=0}^{3} x_2(n)e^{-j2\pi nk/4}, \quad k = 0,1,2,3.1$$

For $k = 0$

$$X_2(0) = \sum_{n=0}^{3} x_2(n)e^{-j2\pi n(0)/4} = 10$$

For $k = 1$

$$X_2(1) = \sum_{n=0}^{3} x_2(n)e^{-j\pi n/2}$$

$$= 1 + 2e^{-j\pi/2} + 3e^{-j\pi} + 4e^{-j3\pi/2} = 1 + 2\ (-j) + 3\ (-1) + 4\ (\ j\ ) = -2 + j2$$

For $k = 2$

$$X_2(2) = \sum_{n=0}^{3} x_2(n)e^{-j\pi n}$$

$$= 1 + 2e^{-j\pi} + 3e^{-j\pi} + 4e^{-j3\pi} = 1 + 2\ (-1) + 3\ (1) + 4\ (-1) = -2$$

For $k = 3$

$$X_2\ (3) = \sum_{n=0}^{3} x_2(n)e^{-j(3\pi/2)n}$$

$$= 1 + 2e^{-j3\pi/2} + 3e^{-j3\pi} + 4e^{-j9\pi/2} = 1 + 2\ (\ j) + 3\ (-1) + 4\ (-j) = -2 - 2j$$

$$X_2(k) = \{10, -2 + 2j, -2, -2 - 2j\}$$

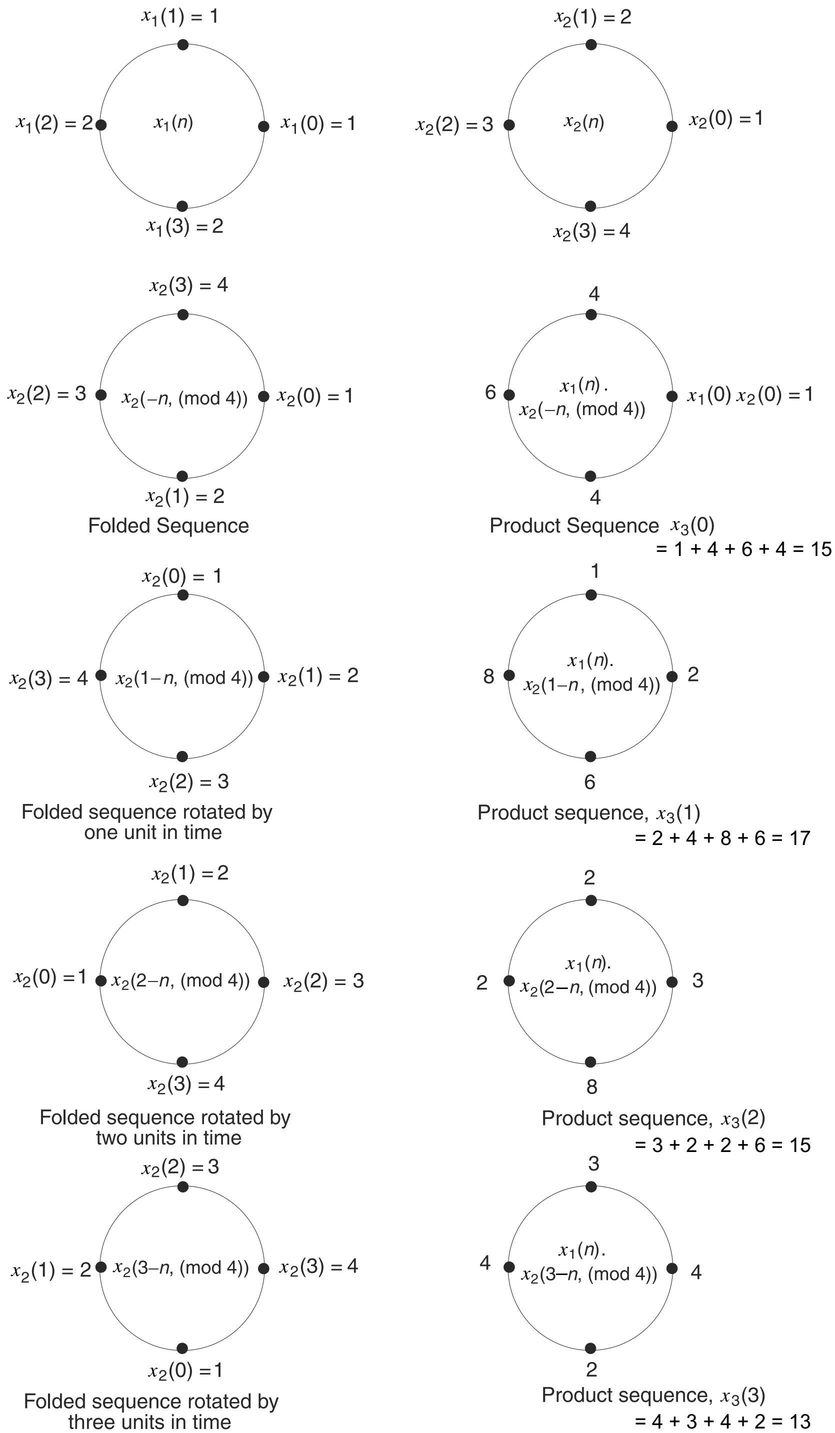

x1(1) = 1
x1(2) = 2
x1(n)
x1(0) = 1
x1(3) = 2
x2(1) = 2
x2(2) = 3
x2(n)
x2(0) = 1
x2(3) = 4
x2(3) = 4
x2(2) = 3
x2(−n, (mod 4))
x2(0) = 1
x2(1) = 2
Folded Sequence
4
6
x1(n) .
x2(−n, (mod 4))
x1(0) x2(0) = 1
4
Product Sequence x3(0)
= 1 + 4 + 6 + 4 = 15
x2(0) = 1
x2(3) = 4
x2(1−n, (mod 4))
x2(1) = 2
x2(2) = 3
Folded sequence rotated by
one unit in time
1
8
x1(n).
x2(1−n, (mod 4))
2
6
Product sequence, x3(1)
= 2 + 4 + 8 + 6 = 17
x2(1) = 2
x2(0) = 1
x2(2−n, (mod 4))
x2(2) = 3
x2(3) = 4
Folded sequence rotated by
two units in time
2
2
x1(n).
x2(2−n, (mod 4))
3
8
Product sequence, x3(2)
= 3 + 2 + 2 + 6 = 15
x2(2) = 3
x2(1) = 2
x2(3−n, (mod 4))
x2(3) = 4
x2(0) = 1
Folded sequence rotated by
three units in time
3
4
x1(n).
x2(3−n, (mod 4))
4
2
Product sequence, x3(3)
= 4 + 3 + 4 + 2 = 13

Fig. E6.15

$$X_3(k) = X_1(k)\,X_2\,(k)$$
$$= \{60, (-1 + j)\,(-2 + 2j), 0, (-1 - j)\,(-2 - 2j)\} = \{60, -4j, 0, 4j\}$$

We know that $x_3(n) = \text{IDFT}\,\{X_3(k)\}$

Hence,

$$x_3(n) = \frac{1}{4}\sum_{k=0}^{3} X_3(k)e^{j2\pi nk/4},\ n\ =\ 0,\ 1,\ 2,\ 3$$

$$= \frac{1}{4}\left[60 + (-4j)e^{j\pi n/2} + (4j)e^{j3\pi n/2}\right]$$

$$x_3(0) = \frac{1}{4}\left[60 + (-4j) + (4j)\right] = 15$$

$$x_3(1) = \frac{1}{4}\left[60 - 4je^{j\pi/2} + 4je^{j3\pi/2}\right] = \frac{1}{4}\left[60 - 4j(+j) + 4j(-j)\right]$$

$$= \frac{1}{4}\left[60 + 4 + 4\right] = 17$$

$$x_3(2) = \frac{1}{4}\left[60 - 4j\,e^{j\pi} + 4j\,e^{j3\pi}\right]$$

$$= \frac{1}{4}\left[60 - 4j(-1) + 4j(-1)\right] = 15$$

$$x_3(3) = \frac{1}{4}\left[60 + (-4j)e^{j3\pi/2} + 4je^{j9\pi/2}\right]$$

$$= \frac{1}{4}\left[60 + (-4j)(-j) + 4j(j)\right] = \frac{1}{4}\left[60 - 4 - 4\right] = 13$$

Therefore, $x_3(n) = \{15, 17, 15, 13\}$

Note: From the above results, we find that the resulting sequences obtained by both linear convolution and circular convolution have different values and length. Linear convolution results in an aperiodic sequence with a length of $(2N - 1)$, i.e. seven in this case, whereas circular convolution results in a periodic sequence with a length of N, i.e.. Four in this case. Circular convolution will produce the same sequence values as those produced by linear convolution if three zeros are padded at the end of the two given sequences $x_1(n)$and $x_2(n)$.

Example 6.16 *Compute* $x_1(n)*x_2(n)$ *if*

$$x_1(n) = \delta\,(n) + \delta\,(n - 1) - \delta\,(n - 2) - \delta\,(n - 3) \text{ and}$$
$$x_2(n) = \delta\,(n) - \delta\,(n - 2) + \delta\,(n - 4)$$

Given $N = 5$.

Solution Given $x_1(n) = \{1, 1, -1, -1\}$

$x_2(n) = \{1, 0, -1, 0, -1\}$

and $N = 5$

Add one zero to the sequence $x_1(n)$ to bring its length to five.

Now, $x_1(n) = \{1, 1, -1, -1, 0\}$

$x_2(n) = \{1, 0, -1, 0, -1\}$

The circular convolution of the two sequences $x_1(n)$ and $x_2(n)$ is shown in Fig. E6.16.

Therefore, $x_3(n) = \{3, 0, -3, -2, 2\}$.

Alternate method

Using matrix approach, we can formulate circular convolution as

$$\begin{bmatrix} x_1(0) & x_1(N-1) & x_1(N-2) & . & . & . & x_1(1) \\ x_1(1) & x_1(0) & x_1(N-1) & & & . & x_1(2) \\ . & . & . & & & . & . \\ . & . & . & & & . & . \\ x_1(N-2) & x_1(N-3) & x_1(N-4) & & & . & x_1(N-1) \\ x_1(N-1) & x_1(N-2) & x_1(N-3) & . & . & . & x_1(0) \end{bmatrix} \begin{bmatrix} x_2(0) \\ x_2(1) \\ . \\ . \\ x_2(N-2) \\ x_2(N-1) \end{bmatrix} = \begin{bmatrix} x_3(0) \\ x_3(1) \\ . \\ . \\ x_3(N-2) \\ x_3(N-1) \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 0 & -1 & 0 \\ 0 & 1 & 1 & 0 & -1 \\ -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & 1 & 1 \\ 1 & 0 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \\ -3 \\ -2 \\ 2 \end{bmatrix}$$

Hence, $x_3(n) = \{3, 0, -3, -2, 2\}$.

Example 6.17 *Find the response of an FIR filter with impulse response* $h(n) = \{1, 2, 4\}$ *to the input sequence* $x(n) = \{1, 2\}$.

Solution

Linear Convolution

Given $h(n) = \{1, 2, 4\}$ and $x(n) = \{1, 2\}$

Here $N_1 = 3$ and $N_2 = 2$. Hence $N = N_1 + N_2 - 1 = 4$

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

Therefore,

$$\begin{aligned} y(0) &= \sum_{k=-\infty}^{\infty} x(k)h(-k) \\ &= \cdots + x(0)\,h(0) + x(1)\,h(-1) + \cdots \\ &= 0 + 1 + 0 \cdots = 1 \\ y(1) &= \sum_{k=-\infty}^{\infty} x(k)h(1-k) \\ &= \cdots + x(0)\,h(1) + x(1)\,h(0) + x(2)\,h(-1) + \cdots \\ &= \cdots + (1)(2) + (2)\,(1) + 0 = 4 \\ y(2) &= \sum_{k=-\infty}^{\infty} x(k)h(2-k) \\ &= \cdots + x(0)\,h(2) + x(1)\,h(1) + x(2)\,h(0) + \cdots \\ &= \cdots\ 0 + (1)(4) + (2)\,(2) + 0 + \cdots = 8 \end{aligned}$$

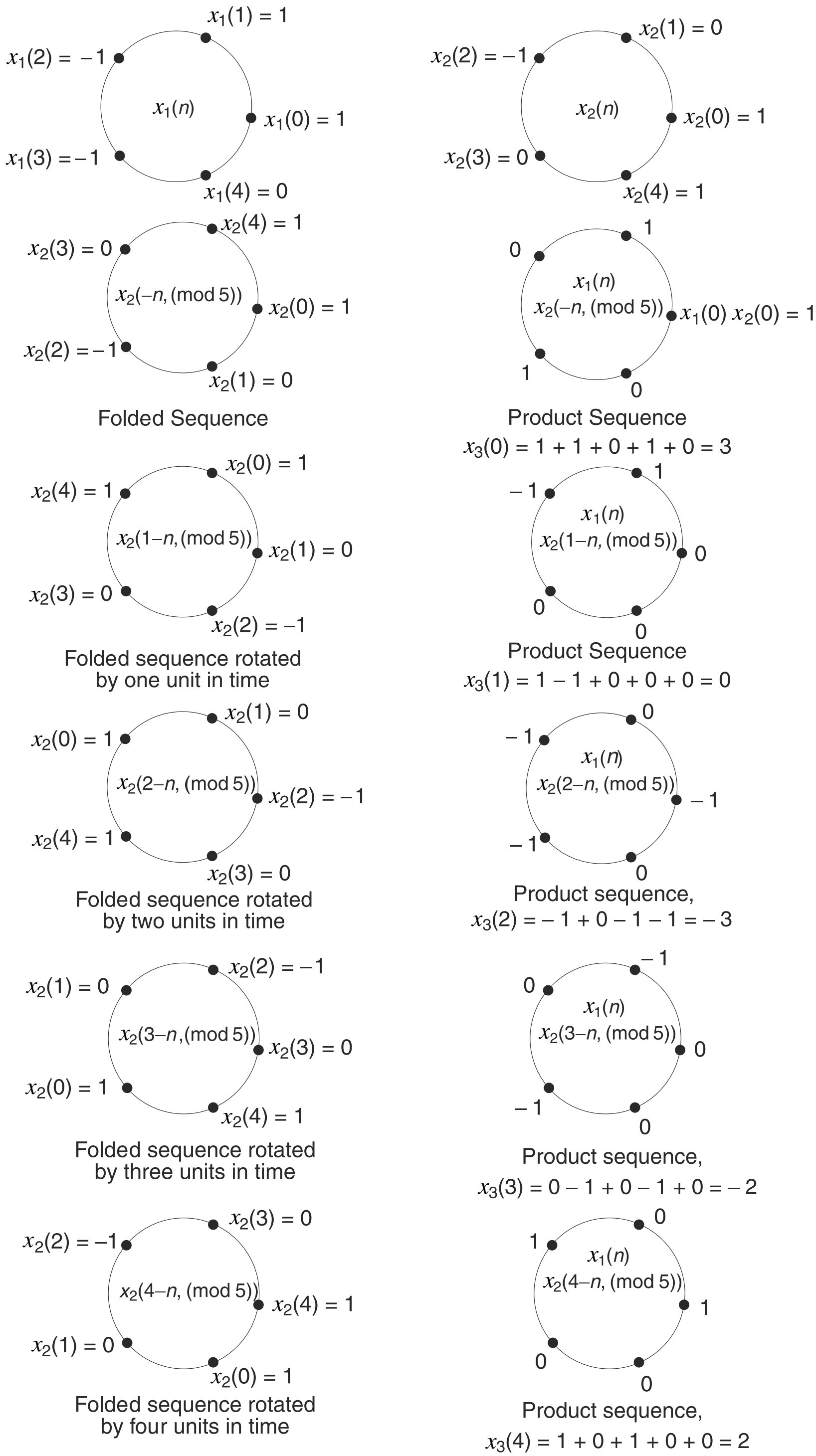

x1(1) = 1
x1(2) = −1
x1(n)
x1(0) = 1
x1(3) = −1
x1(4) = 0
x2(1) = 0
x2(2) = −1
x2(n)
x2(0) = 1
x2(3) = 0
x2(4) = 1
x2(4) = 1
x2(3) = 0
x2(−n, (mod 5))
x2(0) = 1
x2(2) = −1
x2(1) = 0
Folded Sequence
1
0
x1(n)
x2(−n, (mod 5))
x1(0) x2(0) = 1
1
0
Product Sequence
x3(0) = 1 + 1 + 0 + 1 + 0 = 3
x2(0) = 1
x2(4) = 1
x2(1−n, (mod 5))
x2(1) = 0
x2(3) = 0
x2(2) = −1
Folded sequence rotated
by one unit in time
1
− 1
x1(n)
x2(1−n, (mod 5))
0
0
0
Product Sequence
x3(1) = 1 − 1 + 0 + 0 + 0 = 0
x2(1) = 0
x2(0) = 1
x2(2−n, (mod 5))
x2(2) = −1
x2(4) = 1
x2(3) = 0
Folded sequence rotated
by two units in time
0
− 1
x1(n)
x2(2−n, (mod 5))
− 1
− 1
0
Product sequence,
x3(2) = − 1 + 0 − 1 − 1 = − 3
x2(2) = −1
x2(1) = 0
x2(3−n, (mod 5))
x2(3) = 0
x2(0) = 1
x2(4) = 1
Folded sequence rotated
by three units in time
− 1
0
x1(n)
x2(3−n, (mod 5))
0
− 1
0
Product sequence,
x3(3) = 0 − 1 + 0 − 1 + 0 = − 2
x2(3) = 0
x2(2) = −1
x2(4−n, (mod 5))
x2(4) = 1
x2(1) = 0
x2(0) = 1
Folded sequence rotated
by four units in time
0
1
x1(n)
x2(4−n, (mod 5))
1
0
0
Product sequence,
x3(4) = 1 + 0 + 1 + 0 + 0 = 2

Fig. E6.16

$$y(3) = \sum_{k=-\infty}^{\infty} x(k)h(3-k)$$
$$= \cdots + x(0)\ h(3) + x(1)\ h(2) + x(2)\ h(1) + \cdots$$
$$= \cdots 0 + (2)(4) + 0 \ldots = 8$$
$$y(n) = \{1, 4, 8, 8\}$$

Alternate Method

Using matrix representation given below, the linear convolution of the given two sequences can be determined.

$x(n)$ \ $h(n)$	1	2	4
1	1	2	4
2	2	4	8

Hence, $y(n) = \{1, 4, 8, 8\}$

Circular Convolution Using Matrix Multiplication Method

Using matrix approach, we can formulate circular convolution as

$$\begin{bmatrix} 1 & 0 & 0 & 2 \\ 2 & 1 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 4 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 8 \\ 8 \end{bmatrix}$$

Hence, $y(n) = \{1, 4, 8, 8\}$

To find linear convolution through circular convolution with padding of zeros.

Given $h(n) = \{1, 2, 4\}$ and $x(n) = \{1, 2\}$

Therefore, $N_1 = 3$, $N_2 = 2$, $N = N_1 + N_2 - 1 = 4$

Appending zeros to the above sequences, i.e. $(N_2 - 1)$ and $(N_1 - 1)$ zeros are added at the end of $h(n)$ and $x(n)$ respectively. Therefore, $h(n) = \{1, 2, 4, 0\}$ and $x(n) = \{1, 2, 0, 0\}$.

We know that $y(n) = \sum_{n=0}^{N-1} x(n)h(m-n,\ (\text{mod(N)})),\ m = 0,1,\ldots,N-1$

Therefore, from Fig. E6.17, we have

$$y(0) = \sum_{n=0}^{3} x(n)h(-n,(\text{mod}(4))) = 1$$

$$y(1) = \sum_{n=0}^{3} x(n)h(1-n,(\text{mod}(4))) = 4$$

$$y(2) = \sum_{n=0}^{3} x(n)h(2-n,(\text{mod}(4))) = 8$$

$$y(3) = \sum_{n=0}^{3} x(n)h(3-n,(\text{mod}(4))) = 8$$

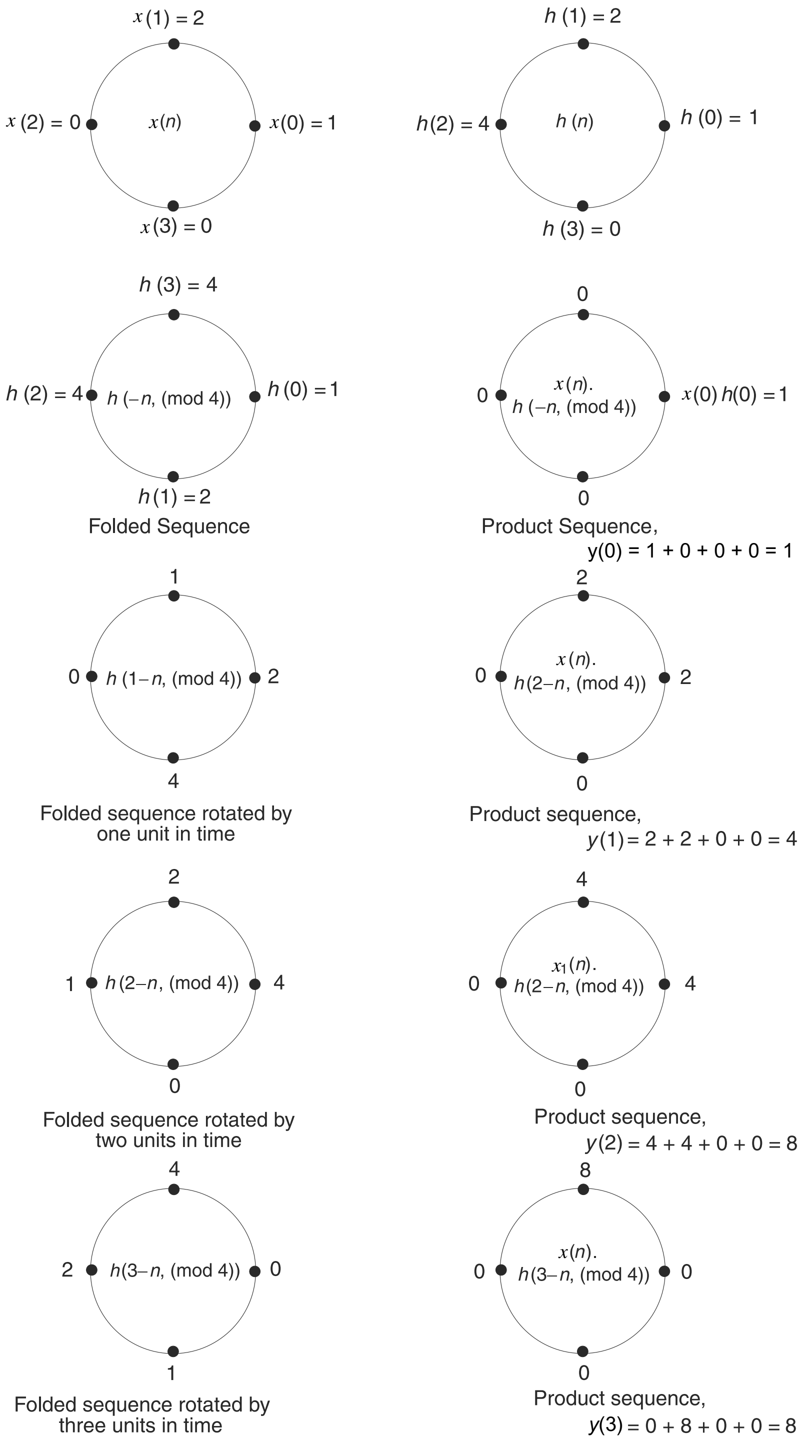

x(1) = 2
x(2) = 0
x(n)
x(0) = 1
x(3) = 0
h(1) = 2
h(2) = 4
h(n)
h(0) = 1
h(3) = 0
h(3) = 4
h(2) = 4
h(−n, (mod 4))
h(0) = 1
h(1) = 2
Folded Sequence
0
0
x(n).
h(−n, (mod 4))
x(0) h(0) = 1
0
Product Sequence,
y(0) = 1 + 0 + 0 + 0 = 1
1
0
h(1−n, (mod 4))
2
4
Folded sequence rotated by
one unit in time
2
0
x(n).
h(2−n, (mod 4))
2
0
Product sequence,
y(1) = 2 + 2 + 0 + 0 = 4
2
1
h(2−n, (mod 4))
4
0
Folded sequence rotated by
two units in time
4
0
x1(n).
h(2−n, (mod 4))
4
0
Product sequence,
y(2) = 4 + 4 + 0 + 0 = 8
4
2
h(3−n, (mod 4))
0
1
Folded sequence rotated by
three units in time
8
0
x(n).
h(3−n, (mod 4))
0
0
Product sequence,
y(3) = 0 + 8 + 0 + 0 = 8

Fig. E6.17

Hence, $y(n) = \{1, 4, 8, 8\}$

Here, after the sequences are padded with zeros, the sequence values obtained by circular convolution is the same as those produced by linear convolution.

Example 6.18 *Consider two periodic sequences x(n) and y(n), x(n) has period N and y(n) has period M. The sequence w(n) is defined as w(n) = x(n) + y(n).*

(a) *Show that w(n) is periodic with period MN.*

(b) *Determine W(k) in terms X(k) and Y(k) where X(k), Y(k) and W(k) are the Discrete Fourier series coefficients with a period of N,M and MN respectively*

Solution Given:

$$w(n) = x(n) + y(n) \tag{1}$$

(a) To determine the relation between L, M and N

We know that $$DFT[w(n)] = W(k) = \sum_{n=0}^{L-1} w(n) e^{-j\left(\frac{2\pi}{L}\right)nk} \tag{2}$$

where L is the discrete frequency index.

Here $W(k)$ is periodic with period L. Therefore,

$$W(0) = W(k)$$
$$W(1) = W(k+1)$$
$$W(2) = W(k+2)$$

and so on.

Substituting Twiddle factor $W_L = e^{-j\frac{2\pi}{L}}$ in Eq. (2), we get

$$W(k) = \sum_{n=0}^{L-1} w(n) W_L^{nk} = \sum_{n=0}^{L-1}[x(n)+y(n)]W_L^{nk} = \sum_{n=0}^{L-1} x(n) W_N^{nk} + \sum_{n=0}^{L-1} y(n) W_M^{nk}$$
$$= \sum_{n=0}^{L-1} x(n) e^{-(j2\pi/N)nk} + \sum_{n=0}^{L-1} y(n) e^{-(j2\pi/M)nk}$$
$$= \sum_{n=0}^{L-1} x(n) e^{-(j2\pi/MN)Mnk} + \sum_{n=0}^{L-1} y(n) e^{-(j2\pi/MN)Nnk}$$

We know that $M \times 2\pi = N \times 2\pi = L \times 2\pi = 2\pi$. Hence $L = MN$

$$W(k) = \sum_{n=0}^{L-1} x(n) e^{-(j2\pi/L)nk} + \sum_{n=0}^{L-1} y(n) e^{-(j2\pi/L)nk}$$
$$W(k) = \sum_{n=0}^{L-1} x(n) W_L^{nk} + \sum_{n=0}^{L-1} y(n) W_L^{nk} \tag{3}$$

Equation (3) shows that if $L = MN$, then only the sum of Eq. $w(n) = x(n) + y(n)$ is possible and relevant, and that LCM of M and N is MN, where $M \neq N$.

(b) To determine W(k) in terms of X(k) and Y(k)

$$W(k) = \sum_{n=0}^{L-1} w(n) W_L^{nk}$$
$$= \sum_{n=0}^{MN-1} x(n) W_{MN}^{nk} + \sum_{n=0}^{MN-1} y(n) W_{MN}^{nk}$$

or $W(k) = X_2(k) + Y_2(k)$ with the period $L = MN$. (4)

Now, $X_2(k)$ with period L can be expressed in terms of $X(k)$, with period N, where

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk} \tag{5}$$

Since $x(n)$ is also periodic with period MN,

$$x(n) = x(n+N) = x(n+2N) = \cdots = x(n+(M-1)N) \tag{6}$$

Using Eqs. (5) and (6), $X(k)$ can be expanded as

$$X(k) = \sum_{n=0}^{N-1} x(n)W_{NM}^{nk} + \sum_{n=0}^{N-1} x(n)W_{NM}^{(n+N)k} + \sum_{n=0}^{N-1} x(n)W_{NM}^{(n+2N)k} + \cdots$$
$$+\sum_{n=0}^{N-1} x(n)W_{NM}^{[(n+(M-1)N)]k} \tag{7}$$

Rearranging Eq. (7) in convenient form, we get

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk/M} + W_N^{nk/M}\sum_{n=0}^{N-1} x(n)W_N^{nk/M} + \cdots$$
$$+W_N^{(M-1)Nk/M}\sum_{n=0}^{N-1} x(n)W_N^{nk/M}$$

Then $X_2(k)$ can be expressed as

$$X_2(k) = \left[1 + W_N^{Nk/M} + W_N^{N(k+1)/M} + \cdots + W_N^{N(M-1)k/M}\right]\sum_{n=0}^{N-1} x(n)W_N^{nk/M}$$
$$= F.X\left(\frac{k}{M}\right) \tag{8}$$

where

$$F = 1 + W_N^{Nk/M} + W_N^{N(k+1)/M} + \cdots + W_N^{N(M-1)k/M}$$
$$= \sum_{p=0}^{M-1} W_N^{PN} = \sum_{p=0}^{M-1} 1 = M \tag{9}$$

where P is any integer. (since $W_N^{PN} = e^{-j(2\pi/N)PN} = 1$)

Therefore, $F = M$ (10)

Substituting $F = M$ in Eq. (8), we get

$$X_2(k) = FX\left(\frac{k}{M}\right) = MX\left(\frac{k}{M}\right) \tag{11}$$

Let $$Y(k) = \sum_{n=0}^{M-1} y(n)W_M^{nk}$$

Since $y(n)$ is also periodic with period NM,

$$y(n) = y(n+M) = y(n+2M) = \cdots = y(n+(N-1)M) \tag{12}$$

$$Y_2(k) = \sum_{n=0}^{M-1} y(n)W_{NM}^{nk} + \sum_{n=0}^{M-1} y(n)W_{NM}^{(n+M)k} + \sum_{n=0}^{M-1} y(n)W_{NM}^{(n+M+1)k} + \cdots + \sum_{n=0}^{M-1} y(n)W_{NM}^{(n+(N-1)M)k}$$
$$= \sum_{n=0}^{M-1} y(n)W_{NM}^{nk} + W_{NM}^{Mk}\sum_{n=0}^{M-1} y(n)W_{NM}^{nk} + \cdots + W_{NM}^{(N-1)Mk}\sum_{n=0}^{M-1} y(n)W_{NM}^{nk}$$

$$= \left[1 + W_{NM}^{Mk} + W_{NM}^{(M+1)k} + \cdots + W_{NM}^{(N-1)Mk}\right] \sum_{n=0}^{M-1} y(n) W_{NM}^{nk}$$

$$= \left[1 + W_{M}^{Mk/N} + W_{M}^{(M+1/N)k} + \cdots + W_{M}^{(N-1)Mk/N}\right] \sum_{n=0}^{M-1} y(n) W_{M}^{nk/N}$$

$$= N \sum_{n=0}^{M-1} y(n) W_{M}^{nk/N} = NY\left(\frac{k}{N}\right) \quad (13)$$

where $\left[1 + W_{M}^{Mk/N} + W_{NM}^{(M+1/N)k} + \cdots + W_{N}^{(N-1/N)Mk} = N\right]$ is already determined

$$W(K)_L = X_2(K) + Y_2(K) \quad (14)$$

$$X_2(k) = MX\left(\frac{k}{M}\right) \quad \text{and} \quad Y_2(k) = NY\left(\frac{k}{N}\right)$$

Substituting the values of $X_2(K)$ and $Y_2(K)$ in Eq. (14), we get

$$W(k)_L = X_2(k)_L + Y_2(k)_L = MX\left(\frac{k}{M}\right) + NY\left(\frac{k}{N}\right)$$

FAST FOURIER TRANSFORM (FFT) 6.4

The fast Fourier transform (FFT) is an algorithm that efficiently computes the discrete Fourier transform (DFT). The DFT of a sequence $\{x(n)\}$ of length N is given by a complex-valued sequence $\{X(k)\}$

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi nk/N}, \quad 0 \le k \le N-1 \quad (6.17)$$

Let W_N be the complex-valued phase factor, which is an Nth root of unity expressed by

$$W_N = e^{-j2\pi/N} \quad (6.18)$$

Hence $X(k)$ becomes

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}, \quad 0 \le k \le N-1 \quad (6.19)$$

Similarly, IDFT becomes

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk} \quad , 0 \le n \le N-1 \quad (6.20)$$

From the above equations, it is evident that for each value of k, the direct computation of $X(k)$ involves N complex multiplications ($4N$ real multiplications) and $N - 1$ complex additions ($4N - 2$ real additions). Hence, to compute all N values of DFT, N^2 complex multiplications and $N(N - 1)$ complex additions are required. The DFT and IDFT involve the same type of computations.

If $x(n)$ is a complex-valued sequence, then the N-point DFT given in Eq. (6.17) can be expressed as

$$X(k) = X_R(k) + j X_1(k) = \sum_{n=0}^{N-1} \left[x_R(n) + j x_1(n)\right] \left[\cos\frac{2\pi nk}{N} - j \sin\frac{2\pi nk}{N}\right]$$

Equating the real and imaginary parts of the above equation, we have

$$X_R(k) = \sum_{n=0}^{N-1} \left[x_R(n)\cos\frac{2\pi nk}{N} + x_1(n)\sin\frac{2\pi nk}{N}\right] \quad (6.21)$$

$$X_I(k)=\sum_{n=0}^{N-1}\left[x_R(n)\sin\frac{2\pi nk}{N}-x_1(n)\cos\frac{2\pi nk}{N}\right] \tag{6.22}$$

The direct computation of the DFT requires $2N^2$ evaluations of trigonometric functions, $4N^2$ real multiplications and $4N(N-1)$ real additions. Also, this is primarily inefficient as it does not exploit the symmetry and periodicity properties of the phase factor W_N, which are given below:

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

Periodicity Property $W_N^{k+N} = W_N^k$ (6.23)

An efficient algorithm for DFT computation is the fast Fourier transform algorithm because FFT algorithms exploit the above two properties.

6.4.1 Radix-2 FFT

By adopting a divide and conquer approach, a computationally efficient algorithm for the DFT can be developed. This approach depends on the decomposition of an N-point DFT into successively smaller size DFTs. If N is factored as $N = r_1\, r_2\, r_3 \cdots r_L$ where $r_1 = r_2 = \cdots = r_L = r$, then $N = r^L$. Hence, the DFT will be of size 'r', where this number 'r' is called the ***radix*** of the FFT algorithm. In this section, the most widely used radix-2 FFT algorithms are described. FFT algorithms take advantage of the periodicity and symmetry of the complex number $W_N^{nk}\left(= e^{-j\left(\frac{2\pi}{N}\right)nk}\right)$.

6.4.2 Decimation-In-Time (DIT) Algorithm

In this case, let us assume that $x(n)$ represents a sequence of N values, where N is an integer power of 2, that is, $N = 2^L$. The given sequence is decimated (broken) into two $\frac{N}{2}$ point sequences consisting of the even numbered values of $x(n)$ and the odd numbered values of $x(n)$.

The N-point DFT of sequence $x(n)$ is given by

$$X(k)=\sum_{n=0}^{N-1}x(n)W_N^{nk},\quad 0\le k\le N-1 \tag{6.24}$$

Breaking $x(n)$ into its even and odd numbered values, we obtain

$$X(k)=\sum_{n=0,n\,\text{even}}^{N-1}x(n)W_N^{nk}+\sum_{n=0,n\,\text{odd}}^{N-1}x(n)W_N^{nk}$$

Substituting $n = 2r$ for n even and $n = 2r + 1$ for n odd, we have

$$\begin{aligned}X(k)&=\sum_{r=0}^{(N/2-1)}x(2r)W_N^{2rk}+\sum_{r=0}^{(N/2-1)}x(2r+1)W_N^{(2r+1)k}\\&=\sum_{r=0}^{(N/2-1)}x(2r)(W_N^2)^{rk}+W_N^k\sum_{r=0}^{(N/2-1)}x(2r+1)(W_N^2)^{rk}\end{aligned} \tag{6.25}$$

Here, $W_N^2=\left[e^{-j(2\pi/N)}\right]^2=e^{-j(2\pi/(N/2))}=W_{N/2}$

Therefore, Eq. (6.25) can be written as

$$X(k)=\sum_{r=0}^{(N/2-1)}x(2r)W_{N/2}^{rk}+W_N^k\sum_{r=0}^{(N/2-1)}x(2r+1)W_{N/2}^{rk}$$

$$= G(k) + W_N^k \cdot H(k), \quad k = 0, 1, \ldots \frac{N}{2} - 1 \tag{6.26}$$

where $G(k)$ and $H(k)$ are the $N/2$-point DFTs of the even and odd numbered sequences respectively. Here, each sum is computed for $0 \le k \le \frac{N}{2} - 1$ since $G(k)$ and $H(k)$ are considered periodic with period $N/2$.

Therefore,

$$X(k) = \begin{cases} G(k) + W_N^k H(k), & 0 \le k \le \frac{N}{2} - 1 \\ G\left(k + \frac{N}{2}\right) + W_N^{(k+N/2)} H\left(k + \frac{N}{2}\right), & \frac{N}{2} \le k \le N - 1 \end{cases} \tag{6.27}$$

Using the symmetry property of $W_N^{k+N/2} = -W_N^k$, Eq. (6.27) becomes

$$X(k) = \begin{cases} G(k) + W_N^k H(k), & 0 \le k \le \frac{N}{2} - 1 \\ G(k + N/2) - W_N^k H(k + N/2), & \frac{N}{2} \le k \le N - 1 \end{cases} \tag{6.28}$$

Figure 6.1 shows the flow graph of the decimation-in-time decomposition of an 8-point ($N = 8$) DFT computation into two 4-point DFT computations. Here the branches entering a node are added to produce the node variable. If no coefficient is indicated, it means that the branch transmittance is equal to one. For other branches, the transmittance is an integer power of W_N.

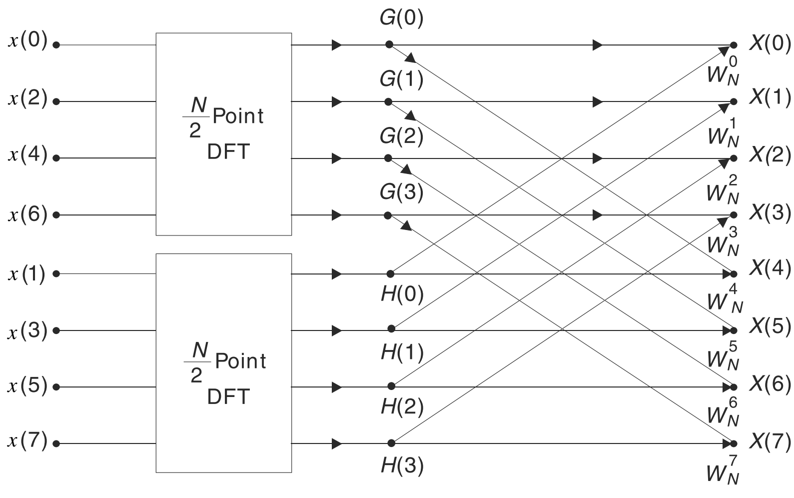


Fig. 6.1 *Flow Graph of the First Stage Decimation-In-Time FFT Algorithm for N = 8*

Then $X(0)$ is obtained by multiplying $H(0)$ by W_N^0 and adding the product to $G(0)$. $X(1)$ is obtained by multiplying $H(1)$ by W_N^1 and adding that result to $G(1)$. For $X(4)$, $H(4)$ is multiplied by W_N^4 and the result is added to $G(4)$. But, since $G(k)$ and $H(k)$ are both periodic in k with period 4, $H(4) = H(0)$ and $G(4) = G(0)$. Therefore, $X(4)$ is obtained by multiplying $H(0)$ by W_N^4 and adding the result to $G(0)$.

For a direct computation of an N-point DFT, without exploiting symmetry, N^2 complex multiplications and $N(N-1) \approx N^2$ complex additions were required. Based on the decomposition of Eq. (6.26), the computation of an N-point DFT using the decomposition requires the computations of two $(N/2)$-point DFTs which requires $2(N/2)^2$ or $N^2/2$ complex multiplications and approximately $2(N/2)^2$ or $N^2/2$ complex additions, which must be combined with N complex multiplications, corresponding to multiplying the second sum by W_N^k and N complex additions, corresponding to adding that product to the first sum. Hence, the computation of Eq. (6.26) for all values of k requires $N + 2(N/2)^2$ or $N + N^2/2$ complex multiplications and $N + N^2/2$ complex additions. It is easy to verify that for $N \geq 3$, $N + N^2/2 < N^2$.

The above process may be continued by expressing each of the two $(N/2)$-point DFTs, $G(k)$ and $H(k)$ as a combination of two $(N/4)$-point DFTs, assuming that $(N/2)$ is even since N is equal to a power of 2. Each of the $(N/2)$-point DFTs in Eq. (6.26) is computed by breaking each of the sums in Eq. (6.26) into two $(N/4)$-point DFTs, which is then combined to give the $(N/2)$-point DFTs. Thus $G(k)$ and $H(k)$ in Eq. (6.26) shall be computed as explained below.

$$
\begin{aligned}
G(k) &= \sum_{r=0}^{(N/2)-1} g(r) W_{N/2}^{rk} \\
&= \sum_{l=0}^{(N/4)-1} g(2l) W_{N/2}^{2lk} + \sum_{l=0}^{(N/4)-1} g(2l+1) W_{N/2}^{(2l+1)k} \\
&= \sum_{l=0}^{(N/4)-1} g(2l) W_{N/4}^{lk} + W_{N/2}^{n} \sum_{l=0}^{(N/4)-1} g(2l+1) W_{N/4}^{lk} \\
G(k) &= A(k) + W_{N/2}^{k} B(k)
\end{aligned}
\tag{6.29}
$$

where $A(k)$ is the $(N/4)$-point DFT of even numbers of the above sequence and

$B(k)$ is the $(N/4)$-point DFT of odd numbers of the above sequence.

Similarly,

$$
\begin{aligned}
H(k) &= \sum_{l=0}^{(N/4)-1} h(2l) W_{N/4}^{lk} + W_{N/2}^{k} \sum_{l=0}^{(N/4)-1} h(2l+1) W_{N/4}^{lk} \\
H(k) &= C(k) + W_{N/2}^{k} D(k)
\end{aligned}
\tag{6.30}
$$

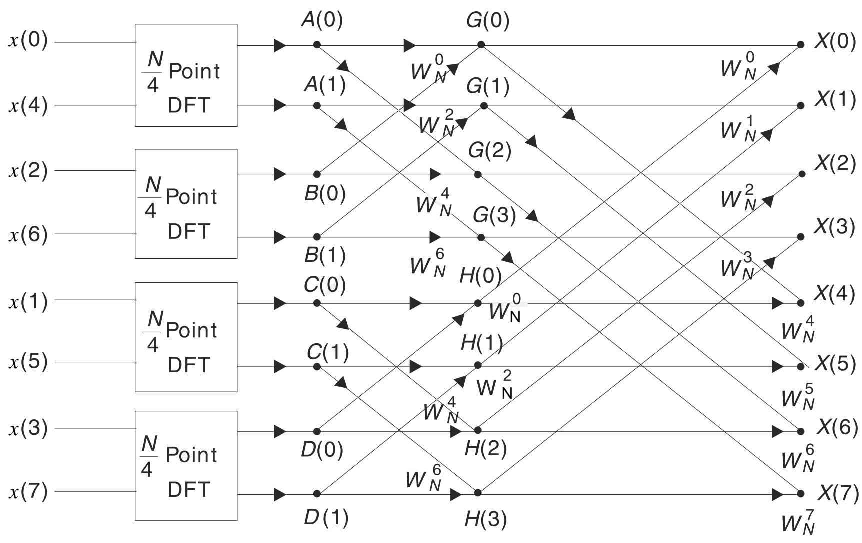


Fig. 6.2 *Flow Graph of the Second Stage Decimation-in-time FFT Algorithm for N* = 8

From Eq. (6.29), we get

$$\text{For } k=0,\ G(0)=A(0)+W_{N/2}^{0}B(0)=A(0)+W_N^0B(0)$$
$$\text{For } k=1,\ G(1)=A(1)+W_{N/2}^{1}B(1)=A(1)+W_N^2B(1)$$
$$\text{For } k=2,\ G(2)=A(2)+W_{N/2}^{2}B(2)=A(0)+W_N^4B(0)$$
$$\text{For } k=3,\ G(3)=A(3)+W_{N/2}^{3}B(3)=A(1)+W_N^6B(1)$$

In the above equations, since $W_{N/2}=W_N^2$ and $W_N^{N/2}=e^{-j\frac{2\pi}{N}\cdot\frac{N}{2}}=e^{-j\pi}=-1$,

$$A(0)=A(2) \text{ and } A(1)=A(3)$$
$$B(0)=B(2) \text{ and } B(1)=B(3)$$

Similarly, from Eq. (6.30), we get

$$H(0)=C(0)+W_N^0D(0)$$
$$H(1)=C(1)+W_N^2D(1)$$
$$H(2)=C(0)+W_N^4D(0)$$
$$H(3)=C(1)+W_N^6D(1)$$

The above process, of reducing an L-point DFT (L is a power of 2) to an $L/2$-point DFTs, can be continued until we are left with 2-point DFTs, or there are L ($=\log_2 N$) stages, to be evaluated.

A 2-point DFT, $F(k)$, $k=0, 1$ may be evaluated as,

$$F(0)=f(0)+f(1)\,W_8^0$$
$$F(1)=f(0)+f(1)\,W_8^4$$

$f(n)$, $n=0, 1$ is a 2-point sequence.

$$W_8^0=1,\ W_8^4=-1$$

Therefore,

$$F(0)=f(0)+W_2^0f(1)=f(0)+f(1)$$
$$F(1)=f(0)+W_2^1f(1)=f(0)-f(1)$$

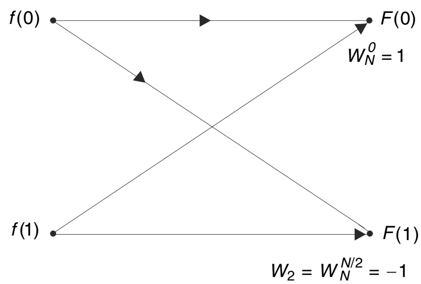


Fig. 6.3 *Flow Graph of a 2-point DFT*

The corresponding flow graph of a 2-point DFT is shown in Fig. 6.3.

The number of complex multiplications and additions required was $N+2(N/2)^2$ or $N+N^2/2$ for the original N-point transform when decomposed into two $(N/2)$- point transforms. When the $(N/2)$-point transforms are decomposed into $N/4$- point transforms, the factor of $(N/2)^2$ is replaced by $N/2+2(N/4)^2$. Hence, the overall computation requires $N+N+4(N/4)^2$ complex multiplications and additions.

Figure 6.4 shows the complete flow graph of the decimation-in-time FFT algorithm for $N=8$, which consists of three stages. The first stage computes the four 2-point DFTs, the second stage computes the two 4-point DFTs and finally the third stage computes the desired 8-point DFT.

From the flow diagram, it is observed that the input data has been shuffled, that is, appears in "bit reversed" order, depicted in Table 6.2 for $N=8$.

The basic computation in the DIT FFT algorithm is illustrated in Fig. 6.5, which is called a butterfly because the shape of its flow graph resembles a butterfly.

The symmetry and periodicity of W_N^r can be exploited to obtain further reductions in computation. The multiplications by $W_N^0=1$, $W_N^{N/2}=-1$, $W_N^{N/4}=j$, and $W_N^{3N/2}=-j$ can be avoided in the DFT computation process in order to save the computational complexity.

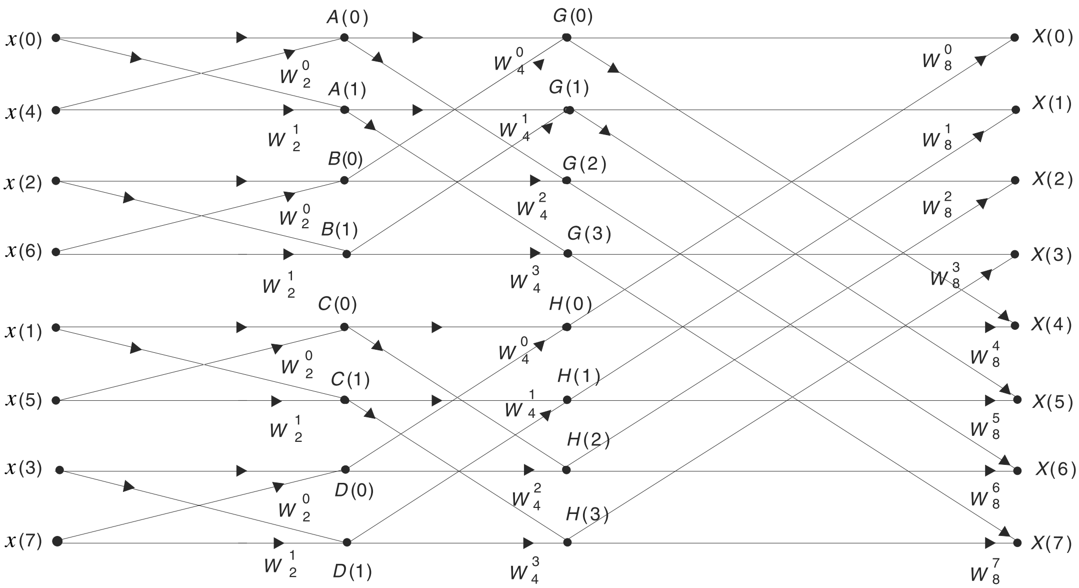


Fig. 6.4 *The Flow-Graph of the Decimation-in-time FFT Algorithm for N = 8*

Table 6.2

Index	*Binary representation*	*Bit reversed binary*	*Bit reversed index*
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7

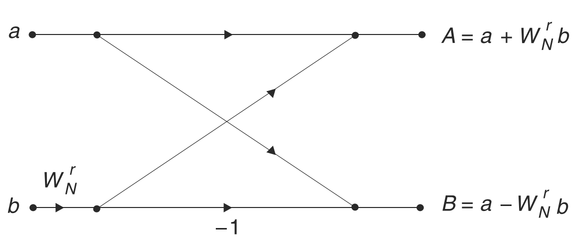


Fig. 6.5 *Basic Butterfly Flow Graph for the Computation in the DIT FFT Algorithm*

In the 8-point DIT FFT flow graph shown in Fig. 6.4, W_2^0, W_2^4, and W_2^8 are equal to 1, and hence these scale factors do not actually represent complex multiplications. Also, since W_2^0, W_2^4, and W_2^8 equal to –1, they do not represent a complex multiplication, where there is just a change in sign. Further, as W_1^4, W_3^4, W_2^8 and W_8^6 are j or $-j$, they need only sign changes and interchanges of real and imaginary parts, even though they represent complex multiplications. When $N = 2^L$, the number of stages of computations is $L = \log_2 N$. Each stage has N complex multiplications and N complex additions. Therefore, the total number of complex multiplications and additions in computing all N-DFT samples is equal to $N \log_2 N$. Hence, the number of complex multiplications is reduced from N^2 to $N \log_2 N$.

In the reduced 8-point DIT FFT flow graph shown in Fig. 6.6, there are actually only four nontrivial complex multiplications corresponding to these scale factors. When the size of the transform is increased, the proportion of nontrivial complex multiplications is reduced and $N \log_2 N$ approximation becomes a little closer.

The reduced flow-graph for 16-point decimation-in-time FFT algorithm is shown in Fig. 6.7.

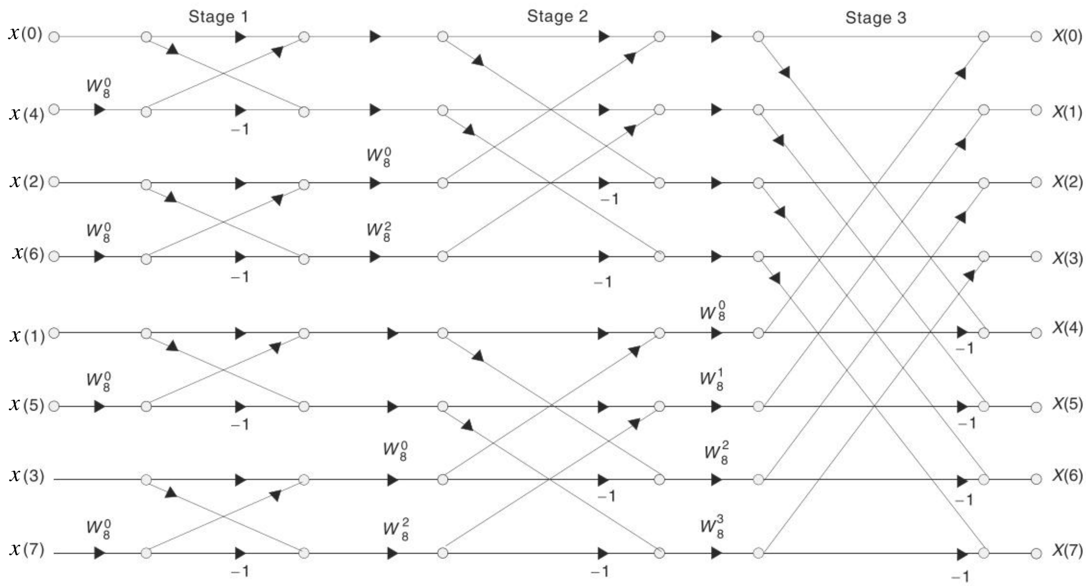


Fig. 6.6 *Reduced Flow-Graph for an 8-Point DIT FFT*

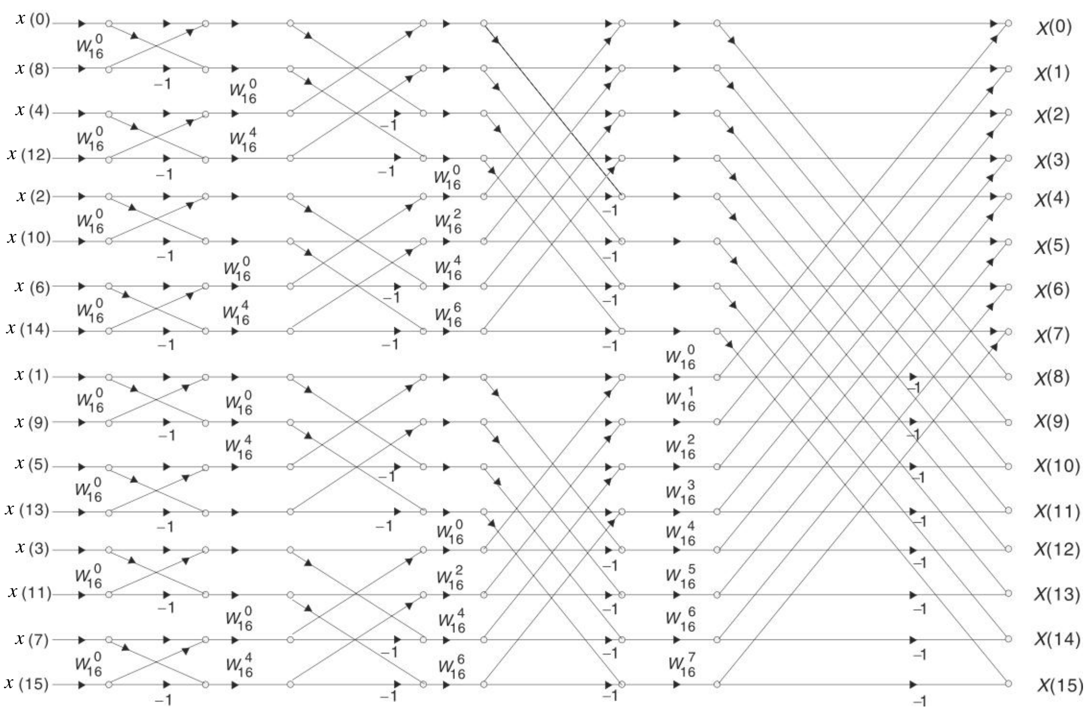


Fig. 6.7 *Flow-Graph of the 16-Point DIT FFT Algorithm*

Example 6.19 *Given x(n) = {1, 2, 3, 4, 4, 3, 2 1}, find X(k) using DIT FFT algorithm.*

Solution

We know that $W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$. Given $N = 8$.

Hence,

$$W_8^0 = e^{-j\left(\frac{2\pi}{8}\right)0} = 1$$

$$W_8^1 = e^{-j\left(\frac{2\pi}{8}\right)} = \cos\ \pi/4 - j\sin\ \pi/4 = 0.707 - j0.707$$

$$W_8^2 = e^{-j\left(\frac{2\pi}{8}\right)2} = \cos\ \pi/2 - j\sin\ \pi/2 = -j$$

$$W_8^3 = e^{-j\left(\frac{2\pi}{8}\right)3} = \cos\ 3\pi/4 - j\sin\ 3\pi/4 = -0.707 - j\,0.707$$

Using DIT FFT algorithm, we can find $X(k)$ from the given sequence $x(n)$ as shown in Fig. E6.19.

Therefore, $X(k) = \{20, -5.828 - j\,2.414, 0, 0.172 - j\,0.414, 0,$
$-0.172 + j\,0.414, 0, -5.828 + j\,2.414\}$

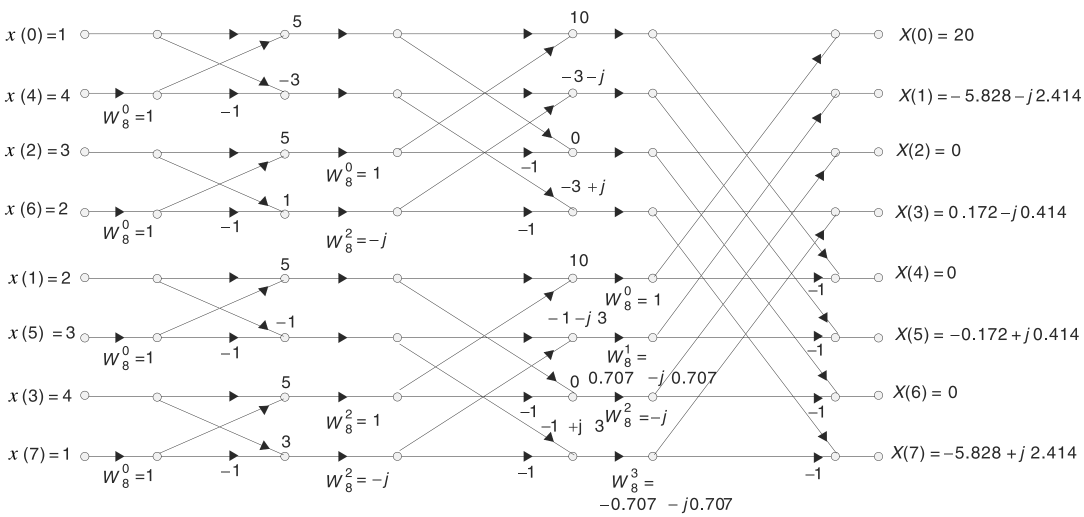


Fig. E6.19

Example 6.20 *Given x(n) = {0, 1, 2, 3, 4, 5, 6, 7}, find X(k) using DIT FFT algorithm.*

Solution Given $N = 8$.

$$W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$$

Hence,

$$W_8^{\,0} = 1$$

$$W_8^{\,1} = 0.707 - j\,0.707$$

$$W_8^{\,2} = -j$$

$$W_8^{\,3} = -0.707 - j\,0.707$$

Using DIT FFT algorithm, we can find $X(k)$ from the given sequence $x(n)$ as shown in Fig. E6.20.

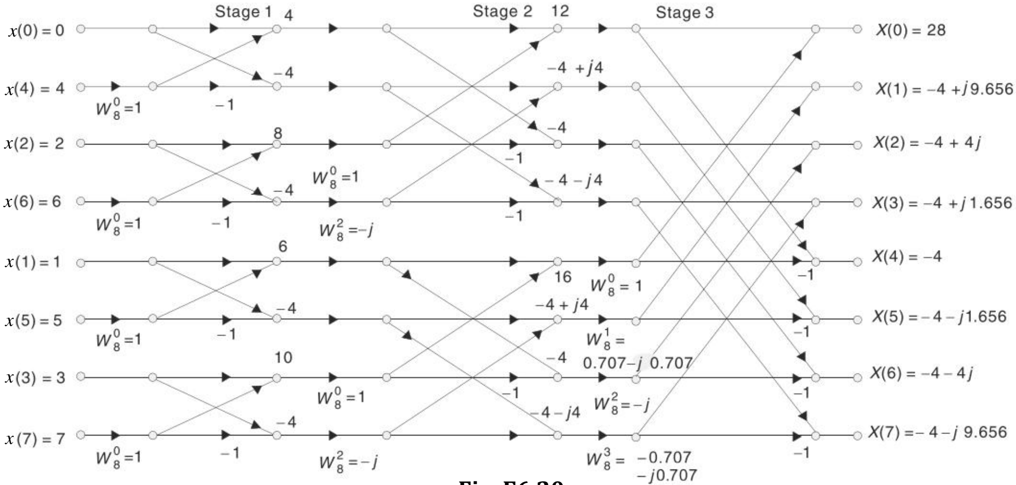


Fig. E6.20

Therefore, $X(k) = \{28, -4 + j\,9.656, -4 + 4j, -4 + j1.656, -4, -4 - j1.656, -4 - 4j, -4 - j\,9.656\}$

Example 6.21 *Given $x(n) = 2^n$ and $N = 8$, find $X(k)$ using DIT FFT algorithm.*

Solution Given

$x(n) = 2^n$ and $N = 8$.

$x(0) = 1, x(1) = 2, x(2) = 4, x(3) = 8$

$x(4) = 16, x(5) = 32, x(6) = 64, x(7) = 128$

$x(n) = \{1, 2, 4, 8, 16, 32, 64, 128\}$

$W_8^0 = 1$

$W_8^1 = 0.707 - j\,0.707$

$W_8^2 = -j$

$W_8^3 = -0.707 - j\,0.707$

Using DIT FFT algorithm, we can find $X(k)$ from the given sequence $x(n)$ as shown in Fig. E6.21.

Therefore, $X(k) = \{255, 48.63 + j166.05, -51 + j102, -78.63 + j46.05, -85,$
$-78.63 - j\,46.05, -51 - j102, 48.63 - j166.05\}$

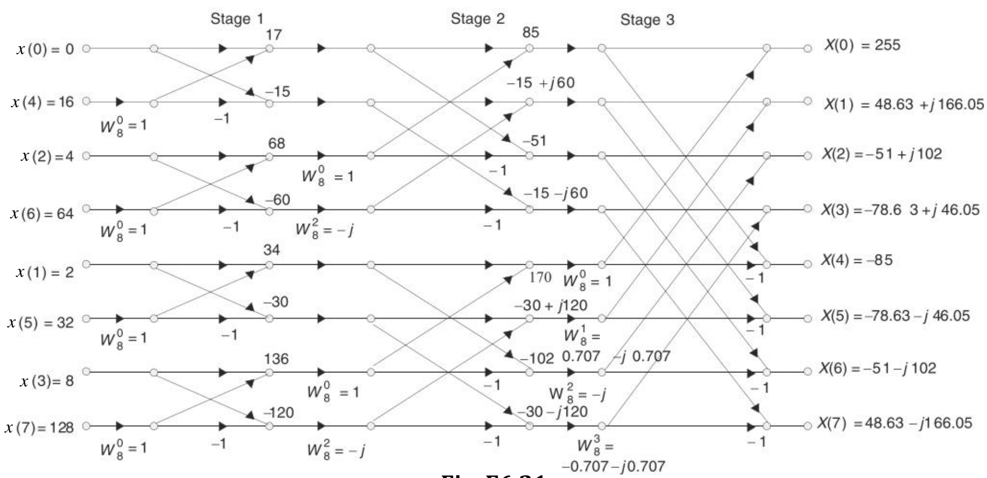


Fig. E6.21

Example 6.22 *Given $x(n) = \{0, 1, 2, 3\}$, find $X(k)$ using DIT FFT algorithm.*

Solution Given $N = 4$

$$W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$$

$$W_4^0 = 1 \text{ and } W_4^1 = e^{-j\pi/2} = -j$$

Using DIT FFT algorithm, we can find $X(k)$ from the given sequence $x(n)$ as shown in Fig. E6.22. Therefore, $X(k) = \{6, -2 + j2, -2, -2 - j2\}$

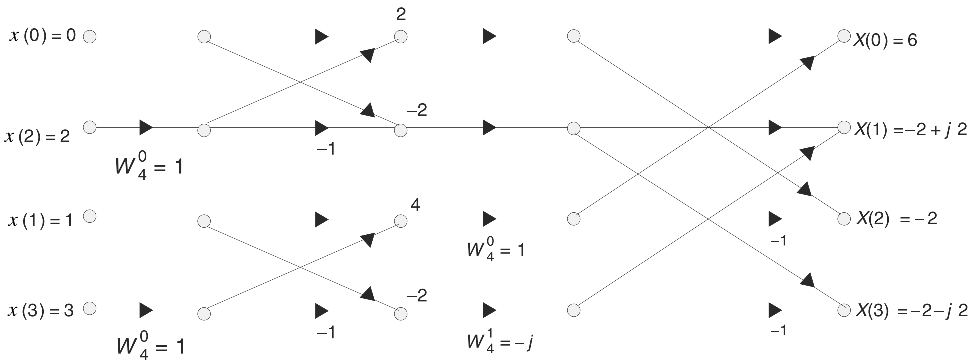


Fig. E6.22

6.4.3 Decimation-In-Frequency (DIF) Algorithms

The decimation-in-time FFT algorithm decomposes the DFT by sequentially splitting input samples $x(n)$ in the time domain into sets of smaller and smaller subsequences and then forms a weighted combination of the DFTs of these subsequences. Another algorithm called decimation-in-frequency FFT decomposes the DFT by recursively splitting the sequence elements $X(k)$ in the frequency domain into sets of smaller and smaller subsequences. To derive the decimation-in-frequency FFT algorithm for N, a power of 2, the input sequence $x(n)$ is divided into the first half and the last half of the points as discussed below.

$$\begin{aligned} x(k) &= \sum_{n=0}^{(N/2)-1} x(n)W_N^{nk} + \sum_{n=N/2}^{N-1} x(n)W_N^{nk} \\ &= \sum_{n=0}^{(N/2)-1} x(n)W_N^{nk} + \sum_{n=0}^{(N/2)-1} x\left(n+\frac{N}{2}\right)W_N^{(n+N/2)k} \\ &= \sum_{n=0}^{(N/2)-1} x(n)W_N^{nk} + W_N^{(N/2)k} + \sum_{n=0}^{(N/2)-1} x(n+N/2)W_N^{nk} \end{aligned} \tag{6.31}$$

Since, $W_N^{(N/2)k} = e^{-j\frac{2\pi}{N}\cdot\frac{N}{2}k} = \cos(\pi k) - j\sin\pi k = (-1)^k$, we obtain

$$\begin{aligned} X(k) &= \sum_{n=0}^{(N/2)-1} x(n)W_N^{nk} + (-1)^k \sum_{n=0}^{(N/2)-1} x(n+N/2)W_N^{nk} \\ &= \sum_{n=0}^{(N/2)-1} \left[x(n) + (-1)^k x\left(n+\frac{N}{2}\right)\right]W_N^{nk} \end{aligned} \tag{6.32}$$

Two different forms of Eq. (6.32) are obtained, depending on whether k is even or odd. Decomposing the sequence in the frequency domain, $X(k)$, into an even numbered subsequence $X(2r)$ and an odd numbered subsequence $X(2r+1)$ where $r = 0, 1, 2, \ldots, (N/2 - 1)$, yields

$$X(2r) = \sum_{n=0}^{(N/2)-1} \left[x(n) + (-1)^{2r} x\left(n + \frac{N}{2}\right) \right] W_N^{2rn}$$

$$= \sum_{n=0}^{(N/2)-1} \left[x(n) + x\left(n + \frac{N}{2}\right) \right] W_{N/2}^{rn}, \quad 0 \le r \le \frac{N}{2} - 1$$

$$[\text{since } W_N^2 = W_{N/2}, W_N^{2rn} = W_{N/2}^{rn}] \tag{6.33}$$

$$X(2r+1) = \sum_{n=0}^{(N/2)-1} \left[x(n) + (-1)^{2r+1} x\left(n + \frac{N}{2}\right) \right] W_N^{(2r+1)n}$$

$$= \sum_{n=0}^{(N/2)-1} \left[x(n) - x\left(n + \frac{N}{2}\right) \right] W_N^n \, W_{N/2}^{rn}, \quad 0 \le r \le \frac{N}{2} - 1 \tag{6.34}$$

Equations (6.33) and (6.34) represent the $N/2$-point DFTs. Equation (6.33) gives the sum of the first half and the last half of the input sequence. Equation (6.34) gives the product of W_N^n with the difference of the first half and the last half of the input sequence.

From Eq. (6.33), $g(n) = x(n) + x\left(n + \frac{N}{2}\right)$ and (6.35)

from Eq. (6.34), $h(n) = x(n) - x\left(n + \frac{N}{2}\right)$ (6.36)

For an 8-point DFT, i.e., $N = 8$

$g(0) = x(0) + x(4)$ $h(0) = x(0) - x(4)$
$g(1) = x(1) + x(5)$ $h(1) = x(1) - x(5)$
$g(2) = x(2) + x(6)$ $h(2) = x(2) - x(6)$
$g(3) = x(3) + x(7)$ $h(3) = x(3) - x(7)$

Here, the computation of the DFT is done by first forming the sequences $g(n)$ and $h(n)$, then calculating $h(n)\, W_N^n$, and finally evaluating the $N/2$-point DFTs of these two sequences to obtain the even numbered output points and the odd numbered output points, respectively. The flow-graph of the first stage of an 8-point DFT computation scheme defined by Eqs. (6.33) and (6.34) is shown in Fig. 6.8.

Equation (6.33) is $X(2r) = \sum_{n=0}^{(N/2)-1} g(n) W_N^{2rn}, \quad 0 \le r \le \frac{N}{2} - 1$

where $g(n) = x(n) + x\left(n + \frac{N}{2}\right)$

Therefore, $X(2r) = \sum_{n=0}^{(N/4)-1} g(n) W_N^{2rn} + \sum_{n=N/4}^{(N/2)-1} g(n) W_N^{2rn}$

$$= \sum_{n=0}^{(N/4)-1} g(n) W_N^{2rn} + \sum_{n=0}^{(N/4)-1} g\left(n + \frac{N}{2}\right) W_N^{2r\left(n + \frac{N}{4}\right)}$$

$$= \sum_{n=0}^{(N/4)-1} g(n) W_N^{2rn} + W_N^{\left(2r \cdot \frac{N}{4}\right)} \sum_{n=0}^{(N/4)-1} g\left(n + \frac{N}{2}\right) W_N^{2rn}$$

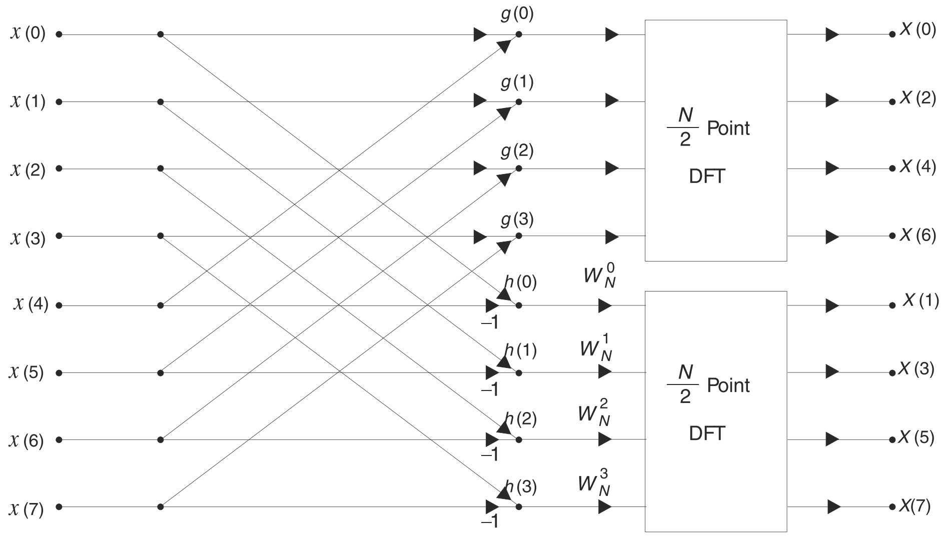


Fig. 6.8 *Flow Graph of the First Stage of Decimation-In-Frequency FFT for N* = 8

Substituting the identity $W_N^{N/2} = -1$ in the above equation, we get

$$X(2r) = \sum_{n=0}^{(N/4)-1}\left[g(n) + (-1)^r g\left(n + \frac{N}{4}\right)\right]W_N^{2rn} \tag{6.37}$$

When $r = 2l$ (even),

$$X(4l) = \sum_{n=0}^{(N/4)-1}\left[g(n) + g\left(n + \frac{N}{4}\right)\right]W_N^{4ln}, \; l = 0, 1, \ldots, \frac{N}{4} - 1$$

$$= \sum_{n=0}^{(N/4)-1}[A(n)]W_N^{4ln}, \text{ where } A(n) = g(n) + g\left(n + \frac{N}{4}\right)$$

Therefore, $A(0) = g(0) + g(2)$

$A(1) = g(1) + g(3)$

When $r = 2l + 1$ (odd),

$$X(4l+2) = \sum_{n=0}^{(N/4)-1}\left[g(n) - g\left(n + \frac{N}{4}\right)\right]W_N^{2n(2l+1)}$$

$$= \sum_{n=0}^{(N/4)-1}[B(n)]W_N^{2n}W_N^{4ln}, \quad 0 \le l \le \left(\frac{N}{4} - 1\right) \tag{6.38}$$

where $B(n) = g(n) - g\left(n + \frac{N}{4}\right)$

$B(0) = g(0) - g(2)$

$B(1) = g(1) - g(3)$

Similarly, Eq. (6.34) becomes

$$X(2r+1) = \sum_{n=0}^{(N/2)-1} h(n)\, W_N^n W_N^{2rn}$$

$$= \sum_{n=0}^{(N/4)-1} h(n)\, W_N^n W_N^{2rn} + \sum_{n=N/4}^{(N/2)-1} h(n) W_N^n W_N^{2rn}$$

$$= \sum_{n=0}^{(N/4)-1} h(n)\, W_N^n W_N^{2rn} + \sum_{n=0}^{(N/4)-1} h\left(n+\frac{N}{4}\right) W_N^{(n+N/4)} W_N^{2r(n+N/4)}$$

$$= \sum_{n=0}^{(N/4)-1} h(n)\, W_N^n W_N^{2rn} + \sum_{n=0}^{(N/4)-1} h\left(n+\frac{N}{4}\right) W_N^n W_N^{2rn} W_N^{(2r+1)N/4}$$

$$= \sum_{n=0}^{(N/4)-1} \left[h(n) + W_N^{(2r+1)N/4} h\left(n+\frac{N}{4}\right)\right] W_N^{(2r+1)n}$$

$$= \sum_{n=0}^{(N/4)-1} \left[h(n) + (-1)^r W_{N/2} h\left(n+\frac{N}{4}\right)\right] W_N^{(2r+1)n} \tag{6.39}$$

Figure 6.9 shows the flow-graph of the second stage of decimation-in-frequency decomposition of an 8-point DFT into four 2-point DFT computations.

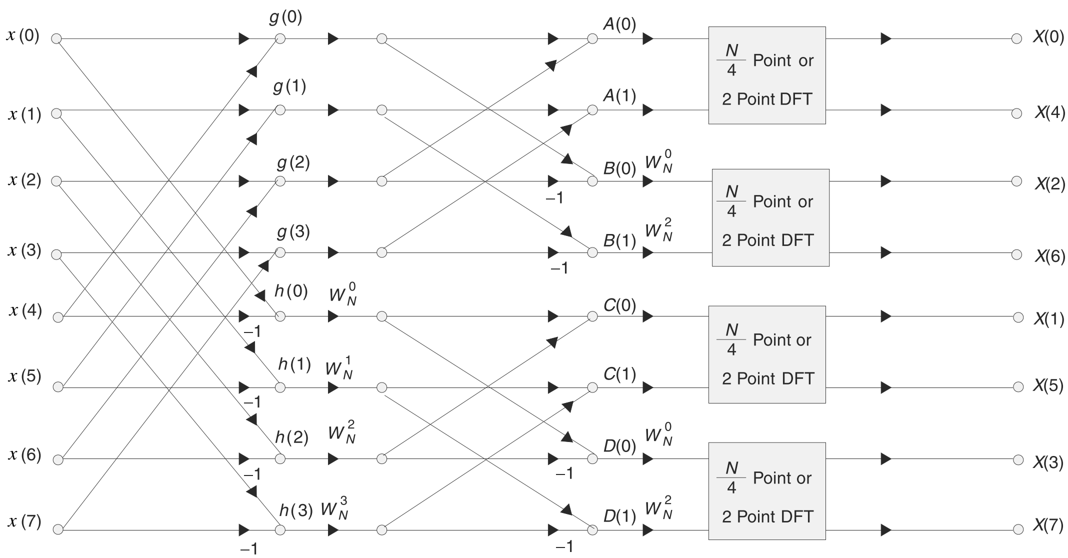


Fig. 6.9 *Flow Graph of the Second Stage of Decimation-In-Frequency FFT for N = 8*

The above decomposition process can be continued through decimation of the $N/2$-point DFTs $X(2r)$ and $X(2r+1)$. The complete process consists of $L = \log_2 N$ stages of decimation, where each stage involves $N/2$ butterflies of the type shown in Fig. 6.10. These butterflies are different from those in the decimation-in-time algorithm. As a result, for computing the N-point DFT down to 2-point transforms, the DIF FFT algorithm requires $(N/2)\log_2 N$ complex multiplications and $N \log_2 N$ complex additions, just as in the case of radix-2 DIT FFT algorithm. The complete flow-graph of an 8-point DIF FFT algorithm is shown in Fig. 6.11.

It is observed from Fig. 6.11 that in the DIF FFT algorithm the input sequence $x(n)$ appears in natural order while the output $X(k)$ appears in the bit-reversed order. The algorithm has in place calculations given below with the butterfly structure shown in Fig. 6.10.

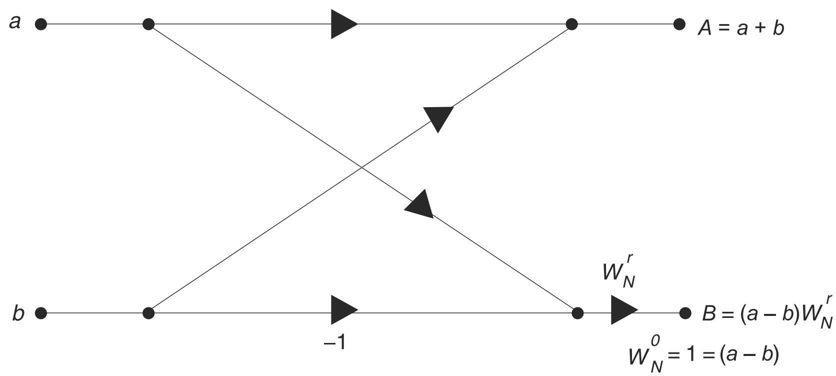


Fig. 6.10 *Basic Butterfly for DIF FFT*

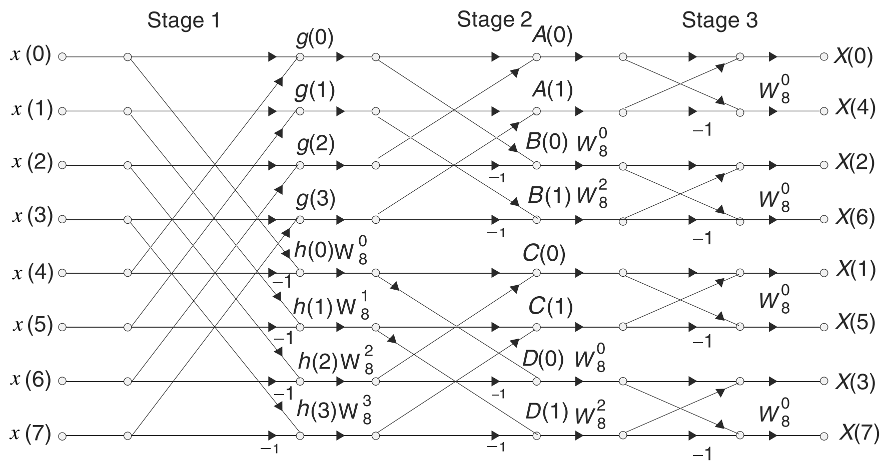


Fig. 6.11 *Reduced Flow Graph of Final Stage DIF FFT for N = 8*

$$A = a + b$$
$$B = (a - b)\, W_N^r \tag{6.40}$$

Example 6.23 *Given x(n) = {1, 2, 3, 4, 4, 3, 2, 1}, find X(k) using DIF FFT algorithm.*

Solution Given $N = 8$.

We know that $W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$

Hence, $W_8^0 = 1$, $W_8^1 = 0.707 - j\,0.707$

$W_8^2 = -j$, $W_8^3 = -0.707 - j\,0.707$

Using DIF FFT algorithm, we can find $X(k)$ from the given sequence $x(n)$ as shown in Fig. E6.23.

Hence, $X(k) = \{20, -5.828 - j\,2.414, 0, -0.172 - j\,0.414, 0,$
$-0.172 + j\,0.414, 0, -5.828 + j\,2.414\}$

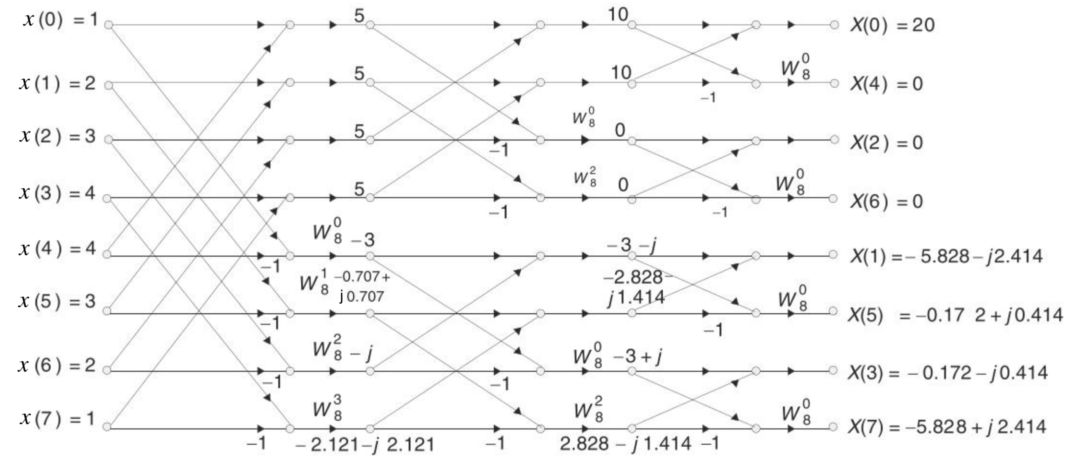


Fig. E6.23

Example 6.24 *Given $x(n) = 2^n$ and $N = 8$, find $X(k)$ using DIF FFT algorithm.*

Solution $x(n) = \{1, 2, 4, 8, 16, 32, 64, 128\}$

Given $N = 8$.

We know that $W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$

Hence, $W_8^0 = 1$

$W_8^1 = 0.707 - j\,0.707$

$W_8^2 = -j$

$W_8^3 = -0.707 - j\,0.707$

Using DIF FFT algorithm, we can find $X(k)$ from the given sequence $x(n)$ as shown in Fig. E6.24.

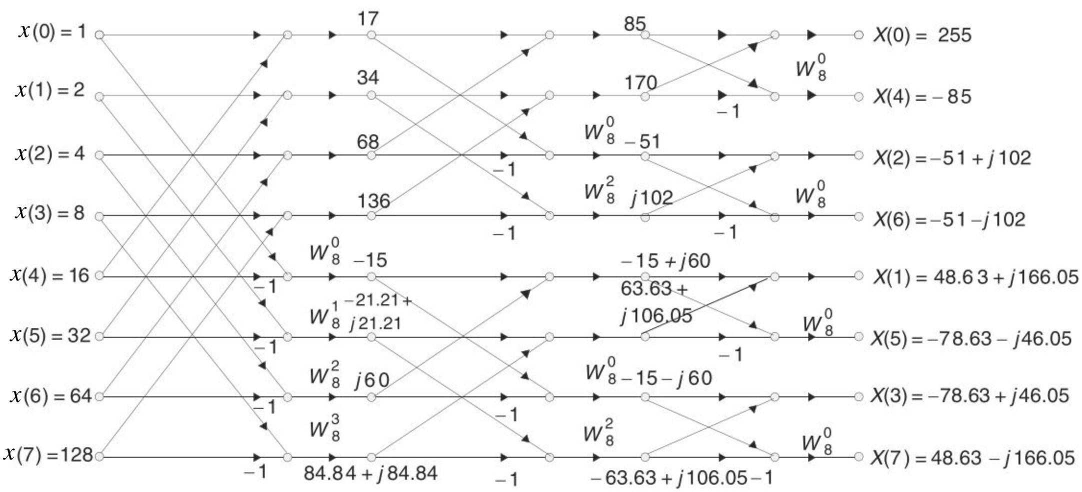


Fig. E6.24

Hence, $X(k) = \{255, 48.63 + j166.05, -51 + j102, -78.63 + j46.05,$
$-85, -78.63 - j46.05, -51 - j102, 48.63 - j166.05\}$

Example 6.25 *Given $x(n) = n + 1$ and $N = 8$, find $X(k)$ using DIF FFT algorithm.*

Solution $x(n) = \{1, 2, 3, 4, 5, 6, 7, 8\}$

Given $N = 8$.

We know that $W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$.

Hence,
$$W_8^0 = 1$$
$$W_8^1 = 0.707 - j\,0.707$$
$$W_8^2 = -j$$
$$W_8^3 = -0.707 - j\,0.707$$

Using DIF FFT algorithm, we can find $X(k)$ from the given sequence $x(n)$ as shown in Fig. E6.25.

Hence, $X(k) = \{36, -4 + j\,9.656, -4 + j\,4, -4 + j\,1.656, -4, -4 - j1.656,$
$-4 - j4, -4 - j\,9.656\}$

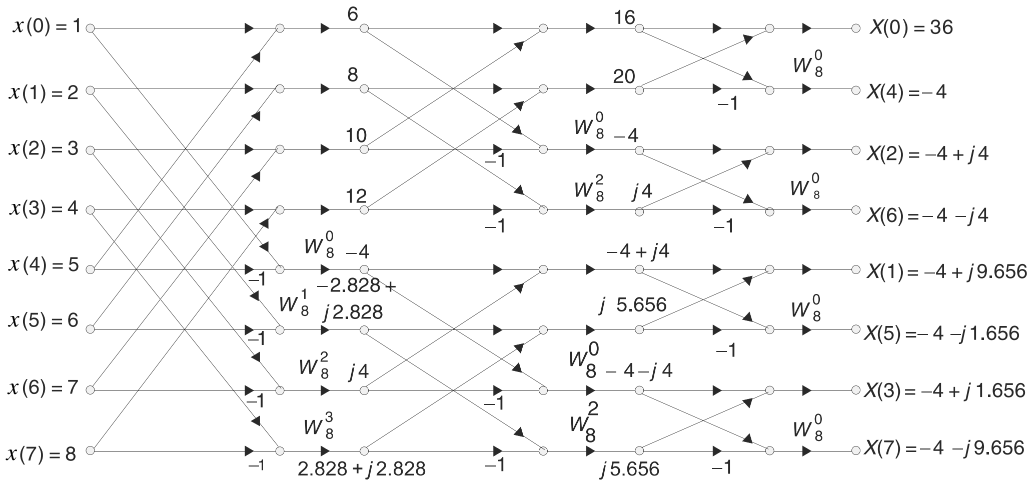


Fig. E6.25

Example 6.26 *Compute the DFTs of the sequence $x(n) = \cos\frac{n\pi}{2}$, where $N = 4$, using DIF FFT algorithm.*

Solution Given $N = 4$ and $x(n) = \{1, 0, -1, 0\}$

$$W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$$

$W_4^0 = 1$ and $W_4^1 = e^{-j\pi/2} = -j$

Using DIF FFT algorithm, we can find $X(k)$ from the given sequence $x(n)$ as shown in Fig. E6.26.

Therefore, $X(k) = \{0, 2, 0, 2\}$

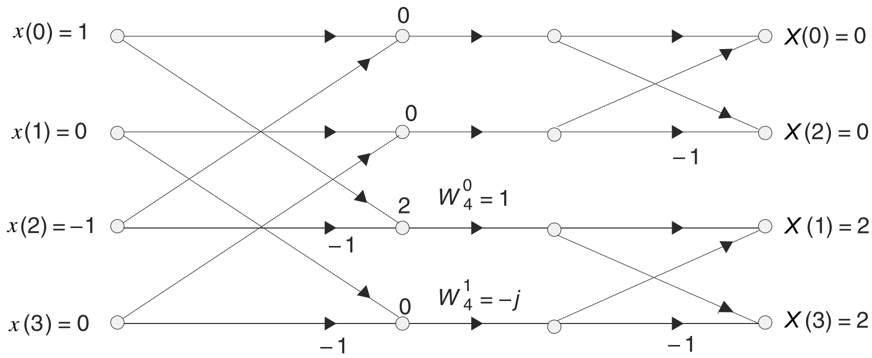


Fig. E6.26

COMPUTING AN INVERSE DFT BY DOING A DIRECT DFT 6.5

An FFT algorithm for calculating the DFT samples can also be used to evaluate efficiently the inverse DFT (IDFT). The inverse DFT is given by

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)W_N^{-nk}, \quad n = 0,1,...N-1 \tag{6.41}$$

Taking complex conjugate of the above expression, we obtain

$$Nx^*(n) = \sum_{k=0}^{N-1} X^*(k)W^{nk} \tag{6.42}$$

The right-hand side is the DFT of the sequence $X^*(k)$. Therefore,

$$x^*(n) = (1/N)\text{ DFT }[X^*(k)] \tag{6.43}$$

Taking the complex conjugate of both sides, we get the desired output sequence $x(n)$ which is given by

$$x(n) = \frac{1}{N}\left[\sum_{k=0}^{N-1} X^*(k)W^{nk}\right]^* \tag{6.44}$$

Hence, $$x(n) = (1/N)\ (\text{FFT}[X^*(k)])^* \tag{6.45}$$

An FFT algorithm can be used to compute the IDFT if the output is divided by N and the "twiddle factors are negative powers of W_N, i.e. powers of W_N^{-1} is used instead of powers of W_N. Therefore, an IFFT flow graph can be obtained from an FFT flow graph by replacing all the $x(n)$ by $X(k)$, dividing the input data by N, or dividing each stage by 2 when N is a power of 2, and changing the exponents of W_N to negative values.

Similarly, an inverse FFT algorithm can be used to calculate the DFT if the output is multiplied by N and powers of W_N are used instead of W_N^{-1}. Hence, the flow graph of Fig. 6.12, corresponding to an inverse FFT algorithm, can be converted to an FFT algorithm by simply converting W_N^{-r} to W_N^{r} and multiplying the output by N. Along with this change and with the input $x(n)$ in normal order while the output $X(k)$ in bit reversed order, Fig. 6.11 can be obtained from Fig. 6.12.

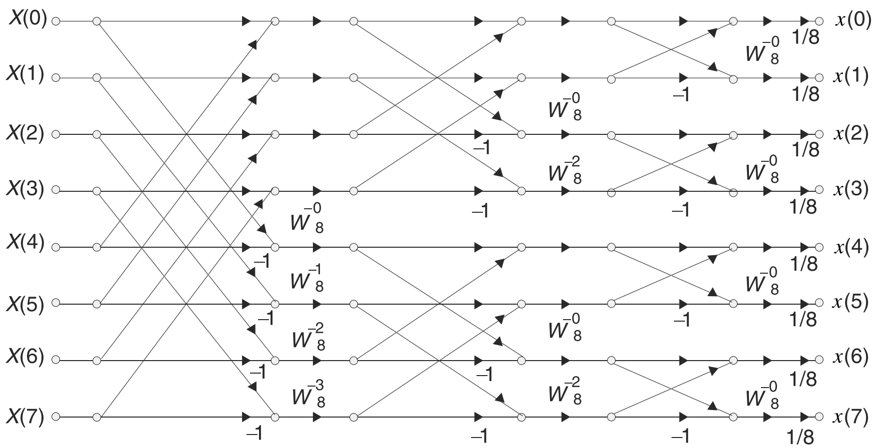


Fig. 6.12

As inverse transform algorithms are related to direct transform algorithms, for each DIT FFT algorithm, there exists a DIF FFT algorithm which corresponds to interchanging the input and output and reversing the direction of all the arrows in the flow diagram.

Example 6.27 *Use the 4-point inverse FFT and verify the DFT results* $\{6, -2+j2, -2, -2-j2\}$ *obtained in Example 6.22 for the given input sequence* $\{0, 1, 2, 3\}$.

Solution

We know that $W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$. Hence,

$W_4^{-0} = 1$ and $W_4^{-1} = e^{j\pi/2} = j$

Using IFFT algorithm, we can find the input sequence $x(n)$ from the given DFT sequence $X(k)$ as shown in Fig. E6.27.

Hence, $x(n) = \{0, 1, 2, 3\}$

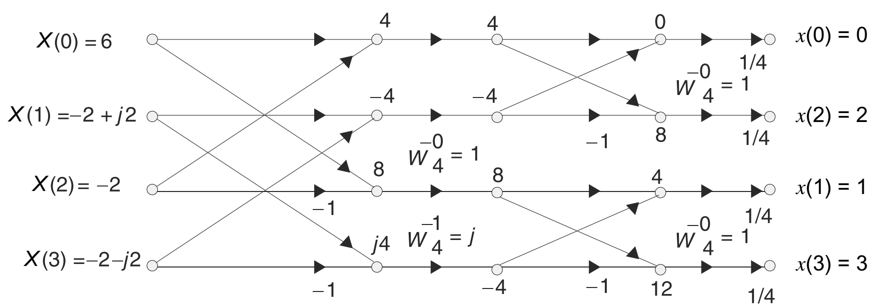


Fig. E6.27

Example 6.28 *Given* $X(k) = \{20, -5.828-j2.414, 0, -0.172-j0.414, 0, -0.172+j0.414, 0, -5.828+j2.414\}$, *find* $x(n)$.

Solution

We know that $W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$. Given $N = 8$. Hence,

$$W_8^{-0} = 1$$
$$W_8^{-1} = 0.707 + j\,0.707$$
$$W_8^{-2} = -j$$
$$W_8^{-3} = -0.707 + j\,0.707$$

Using IFFT algorithm, we can find $x(n)$ from $X(k)$ as shown in Fig. E.6.28.

Hence, $x(n) = \{1, 2, 3, 4, 4, 3, 2, 1\}$

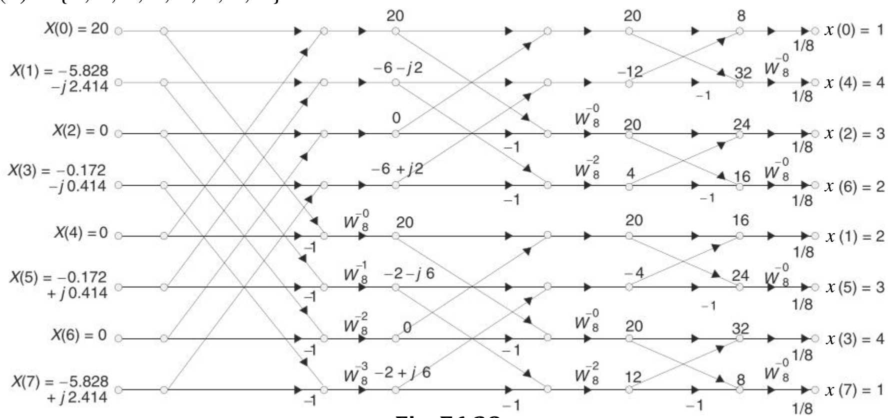


Fig. E6.28

Example 6.29 *Given $X(k)$ = {255, 48.63 + j166.05, –51 + j102, –78.63 + j46.05, –85, –78.63 – j46.05, – 51 – j102, 48.63 – j166.05}, find $x(n)$.*

Solution

We know that $W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$. Given $N = 8$. Hence,

$$W_8^{-0} = 1$$
$$W_8^{-1} = 0.707 + j\,0.707$$
$$W_8^{-2} = -j$$
$$W_8^{-3} = -0.707 + j\,0.707$$

Using IFFT algorithm, we can find $x(n)$ from $X(k)$ as shown in Fig. E6.29.

Hence, $x(n) = \{1, 2, 4, 8, 16, 32, 64, 128\}$

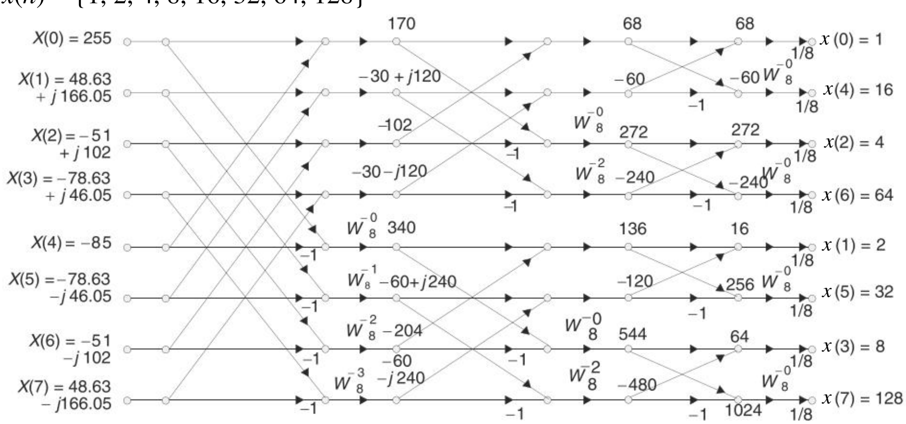


Fig. E6.29

Example 6.30 *Given* $X(k) = \{36, -4 + j9.656, -4 + j4, -4 + j1.656, -4, -4 - j1.656, -4 - j4, -4 - j9.656\}$, *find* $x(n)$.

Solution

We know that $W_N^k = e^{-j\left(\frac{2\pi}{N}\right)k}$. Given $N = 8$. Hence,

$$W_8^{-0} = 1$$
$$W_8^{-1} = 0.707 + j\,0.707$$
$$W_8^{-2} = +j$$
$$W_8^{-3} = -0.707 + j\,0.707$$

Using IFFT algorithm, we can find $x(n)$ from $X(k)$ as shown in Fig. E6.30.

Hence, $x(n) = \{1, 2, 3, 4, 5, 6, 7, 8\}$

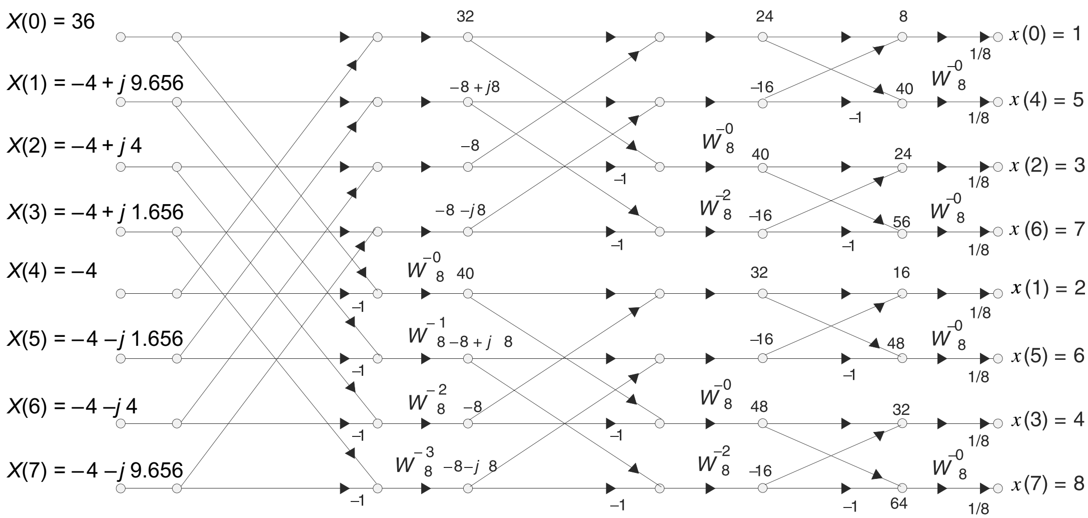


Fig.E6.30

Example 6.31 *Determine DFT (8-point) for a continuous time signal,* $x(t) = \sin(2\pi ft)$ *with* $f = 50$ Hz.

Solution We know that $T = \frac{1}{f} = \frac{1}{50} = 0.02\text{s}$.

$t = 0$,	$x(t) = 0$
$t = 0.0025$ s,	$x(t) = 0.707$
$t = 0.005$ s,	$x(t) = 1$
$t = 0.0075$ s,	$x(t) = 0.707$
$t = 0.01$ s,	$x(t) = 0$
$t = 0.0125$ s,	$x(t) = -0.707$
$t = 0.015$ s,	$x(t) = -1$
$t = 0.0175$ s,	$x(t) = -0.707$

Therefore, $x(n) = \{0, 0.707, 1, 0.707, 0, -0.707, -1, -0.707\}$

These values of the continuous-time signal are plotted in Fig. E6.31(a).

We know that

$$W_8^{-0} = 1$$
$$W_8^1 = 0.707 - j\,0.707$$
$$W_8^2 = -j$$
$$W_8^3 = -0.707 - j\,0.707$$

Using DIF FFT algorithm, we can find $X(k)$ from the sequence $x(n)$ as shown in Fig. E6.31(b).

Therefore, $X(k) = \{0, -j4, 0, 0, 0, 0, 0, j4\}$

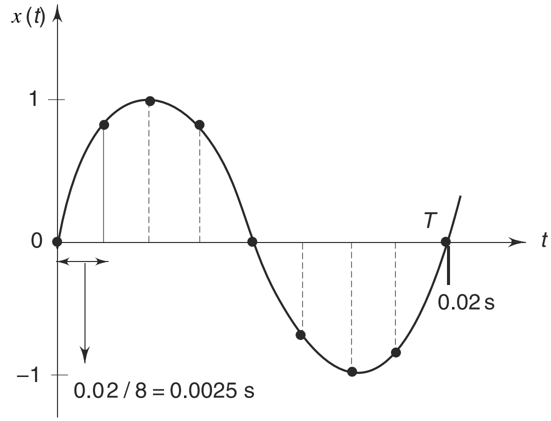


Fig. E6.31(a)

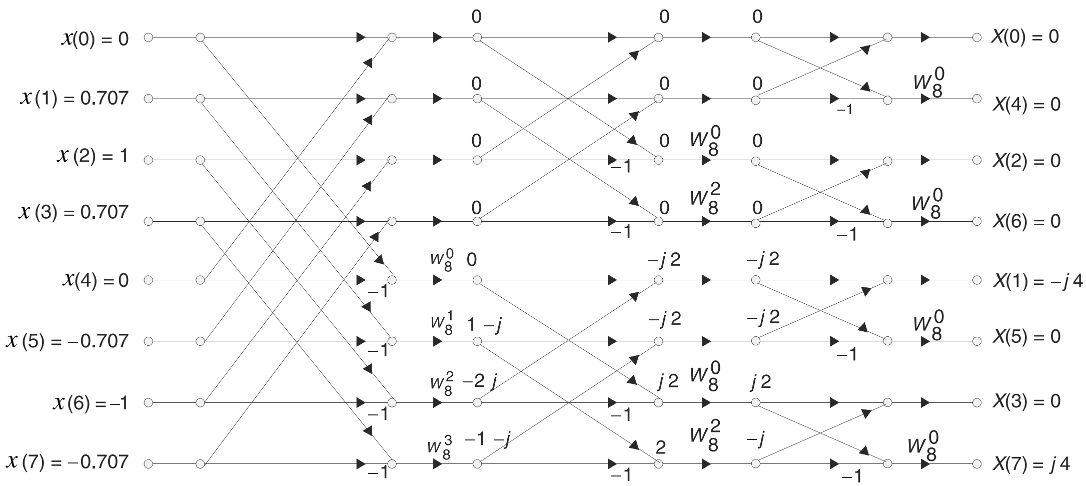


Fig. E6.31(b)

COMPOSITE-RADIX FFT 6.6

A composite or mixed-radix FFT is used when N is a composite number which has more than one prime factor; for example $N = 6$ or 12. For these cases also, efficient DIT and DIF algorithms can be developed. Here we give a DIT FFT decomposition and hence N is written as a product of factors such as $N = m_1\, m_2 \ldots m_r$.

If $N = m_1 N_1$ where $N_1 = m_2 \ldots m_r$, the input sequence $x(n)$ can be separated into m_1 subsequences of N_1 elements each. Then the DFT can be written as

$$X(k) = \sum_{n=0}^{N_1-1} x(nm_1) W_N^{nm_1 k} + \sum_{n=0}^{N_1-1} x(nm_1+1) W_N^{(nm_1+1)k} + \ldots + \sum_{n=0}^{N_1-1} x(nm_1+m_1-1) W_N^{(nm_1+m_1-1)k} \quad (6.46)$$

Radix-3 FFT

In this case, N has been assumed to be a power of 3, that is, $N = 3^L$. For example, when $N = 9 = 3^2$, the given sequence $x(n)$ is decimated into three sequences of length $N/3$ each. For evaluating the DFT for $N = 9$, a radix-3 DIT and DIF FFT algorithms are developed in the Examples 6.32 and 6.37 respectively.

Radix-4 FFT

In this case, N is assumed to be a power of 4, i.e. $N = 4^L$. When $N = 16 = 4^2$, the given sequence $x(n)$ is decimated into four sequences of length $N/4$ each. In the Example 6.38 a radix-4 DIT FFT algorithm is developed for evaluating the DFT for $N = 16$.

Example 6.32 *Develop a radix-3 DIT FFT algorithm for evaluating the DFT for $N = 9$.*

Solution For $N = 9 = 3.3$, Eq. (6.46) becomes

$$X(k) = \sum_{n=0}^{2} x(3n)W_9^{3nk} + \sum_{n=0}^{2} x(3n+1)W_9^{(3n+1)k} + \sum_{n=0}^{2} x(3n+2)W_9^{(3n+2)k}$$
$$= X_1(k) + W_9^k X_2(k) + W_9^{2k} X_3(k)$$

where,
$$X_1(k) = \sum_{n=0}^{2} x(3n)W_9^{3nk} = x(0) + x(3)W_9^{3k} + x(6)W_9^{6k}$$
$$X_2(k) = \sum_{n=0}^{2} x(3n+1)W_9^{3nk} = x(1) + x(4)W_9^{3k} + x(7)W_9^{6k}$$
$$X_3(k) = \sum_{n=0}^{2} x(3n+2)W_9^{3nk} = x(2) + x(5)W_9^{3k} + x(8)W_9^{6k}$$

$$X(0) = X_1(0) + W_9^0 X_2(0) + W_9^0 X_3(0)$$
$$X(1) = X_1(1) + W_9^1 X_2(1) + W_9^2 X_3(1)$$
$$X(2) = X_1(2) + W_9^2 X_2(2) + W_9^4 X_3(2)$$
$$X(3) = X_1(0) + W_9^3 X_2(0) + W_9^6 X_3(0)$$
$$X(4) = X_1(1) + W_9^4 X_2(1) + W_9^8 X_3(1)$$
$$X(5) = X_1(2) + W_9^5 X_2(2) + W_9^{10} X_3(2)$$
$$X(6) = X_1(0) + W_9^6 X_2(0) + W_9^{12} X_3(0)$$
$$X(7) = X_1(1) + W_9^7 X_2(1) + W_9^{14} X_3(1)$$
$$X(8) = X_1(2) + W_9^8 X_2(2) + W_9^{16} X_3(2)$$

Figure E6.32 shows the radix-3 decimation-in-time FFT flow diagram for $N = 9$. Here, we have repeated the 3-point cat's cradle structure as we had repeated butterflies in the radix-2 case. The input sequence appears in digit-reversed order.

Example 6.33 *Develop DIT FFT algorithms for decomposing the DFT for $N = 6$ and draw the flow diagrams for* (a) $N = 2.3$ *and* (b) $N = 3.2$. (c) *Also, by using the FFT algorithm developed in part* (b), *evaluate the DFT values for* $x(n) = \{1, 2, 3, 4, 5, 6\}$.

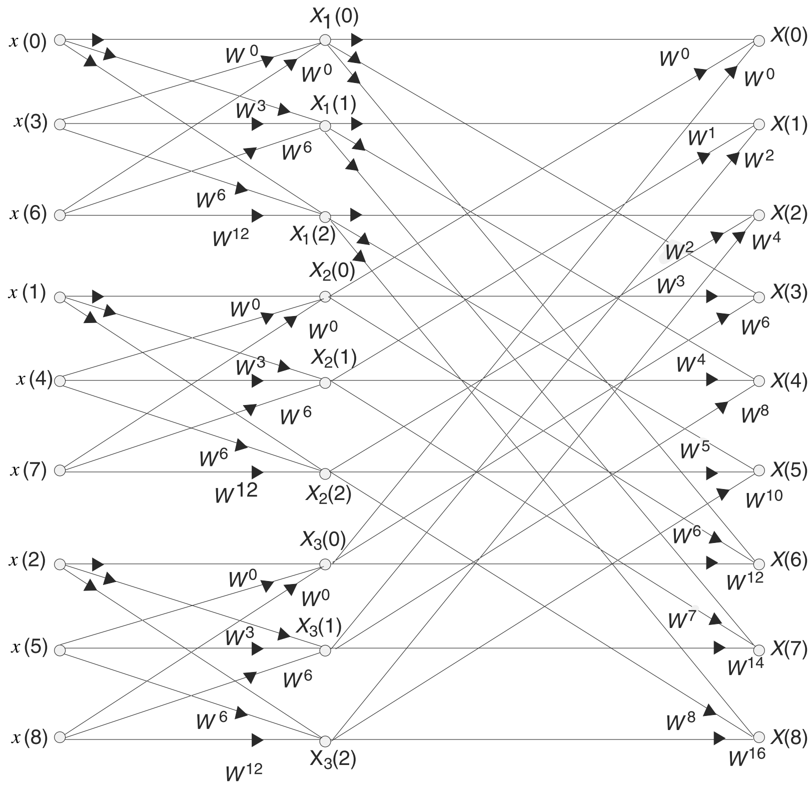


Fig. E6.32 *Radix-3 DIT FFT Flow Diagram for N = 9*

Solution

(a) For $N = 6 = 2.3$, where $m_1 = 2$ and $N_1 = 3$, Eq. (6.46) becomes

$$X(k) = \sum_{n=0}^{2} x(2n)W_6^{2nk} + \sum_{n=0}^{2} x(2n+1)W_6^{(2n+1)k}$$

$$= \sum_{n=0}^{2} x(2n)W_6^{2nk} + W_6^k \sum_{n=0}^{2} x(2n+1)W_6^{2nk}$$

Also, $X_i(k+3) = X_i(k)$

$$X_1(k) = \sum_{n=0}^{2} x(2n)W_6^{2nk}$$

$$= x(0) + x(2)W_6^{2k} + x(4)W_6^{4k}$$

$$X_1(0) = x(0) + x(2) + x(4)$$

$$X_1(1) = x(0)W_6^0 + x(2)W_6^2 + x(4)W_6^4 = x(0) + x(2)W_6^2 + x(4)W_6^4$$

$$X_1(2) = x(0)W_6^0 + x(2)W_6^4 + x(4)W_6^8 = x(0) + x(2)W_6^4 + x(4)W_6^2$$

$$X_2(k) = \sum_{n=0}^{2} x(2n+1)W_6^{2nk} = x(1) + x(3)W_6^{2k} + x(5)W_6^{4k}$$

$$X_2(0) = x(1) + x(3) + x(5)$$
$$X_2(1) = x(1) + x(3)W_6^2 + x(5)W_6^4$$
$$X_2(2) = x(1) + x(3)W_6^4 + x(5)W_6^8 = x(1) + x(3)W_6^4 + x(5)W_6^2$$
$$X(k) = X_1(k) + W_6^k X_2(k)$$
$$X(0) = X_1(0) + X_2(0), \quad X(1) = X_1(1) + W_6^1 X_2(1)$$
$$X(2) = X_1(2) + W_6^2 X_2(2), \quad X(3) = X_1(0) + W_6^3 X_2(0)$$
$$X(4) = X_1(1) + W_6^4 X_2(1), \quad X(5) = X_1(2) + W_6^5 X_2(2)$$

Figure E6.33(a) shows the decimation-in-time FFT flow diagram for $N = 6 = 2.3$.

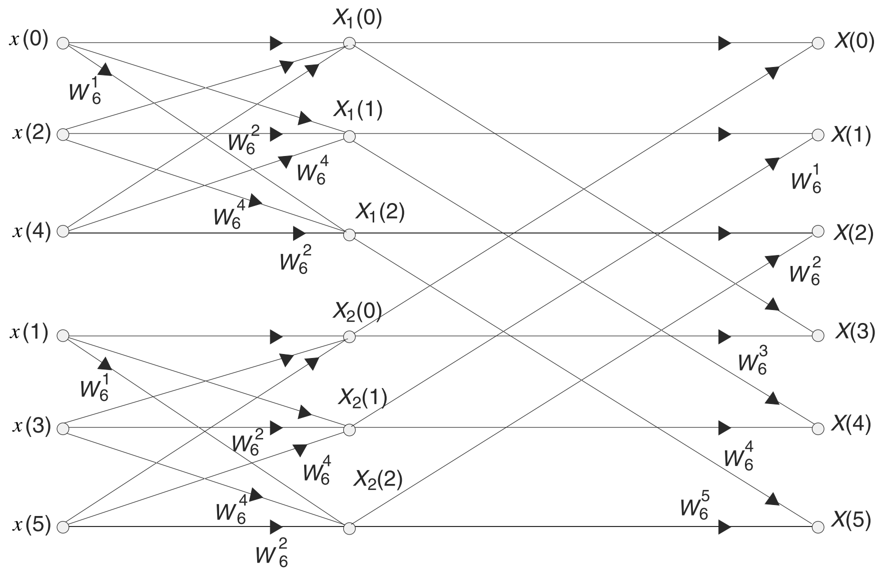


Fig. E6.33(a) *DIT FFT Flow Diagram for N=6=2.3*

(b) For $N = 6 = 3.2$, where $m_1 = 3$ and $N_1 = 2$, Eq. (6.46) becomes

$$X(k) = \sum_{n=0}^{1} x(3n)W_6^{3nk} + \sum_{n=0}^{1} x(3n+1)W_6^{(3n+1)k} + \sum_{n=0}^{1} x(3n+2)W_6^{(3n+2)k}$$
$$= \sum_{n=0}^{1} x(3n)W_6^{3nk} + W_6^k \sum_{n=0}^{1} x(3n+1)W_6^{3nk} + W_6^{2k} \sum_{n=0}^{1} x(3n+2)W_6^{3nk}$$
$$= X_1(k) + W_6^k X_2(k) + W_6^{2k} X_3(k)$$

where, $X_1(k) = \sum_{n=0}^{1} x(3n)W_6^{3nk} = x(0) + x(3)W_6^{3k}$

$$X_2(k) = \sum_{n=0}^{1} x(3n+1)W_6^{3nk} = x(1) + x(4)W_6^{3k}$$
$$X_3(k) = \sum_{n=0}^{1} x(3n+2)W_6^{3nk} = x(2) + x(5)W_6^{3k}$$

Also, $X_i(k+2) = X_i(k)$

$$X_1(0) = x(0) + x(3)W_6^0$$
$$X_1(1) = x(0) + x(3)W_6^3$$
$$X_2(0) = x(1) + x(4)W_6^0$$
$$X_2(1) = x(1) + x(4)W_6^3$$
$$X_3(0) = x(2) + x(5)W_6^0$$
$$X_3(1) = x(2) + x(5)W_6^3$$
$$X(k) = X_1(k) + W_6^k X_2(k) + W_6^{2k} X_3(k)$$
$$X(0) = X_1(0) + W_6^0 X_2(0) + W_6^0 X_3(0)$$
$$X(1) = X_1(1) + W_6^1 X_2(1) + W_6^2 X_3(1)$$
$$X(2) = X_1(2) + W_6^2 X_2(2) + W_6^4 X_3(2)$$
$$= X_1(0) + W_6^2 X_2(0) + W_6^4 X_3(0)$$
$$X(3) = X_1(3) + W_6^3 X_2(3) + W_6^6 X_3(3)$$
$$= X_1(1) + W_6^3 X_2(1) + W_6^6 X_3(1)$$
$$X(4) = X_1(4) + W_6^4 X_2(4) + W_6^8 X_3(4)$$
$$= X_1(0) + W_6^4 X_2(0) + W_6^2 X_3(0)$$
$$X(5) = X_1(5) + W_6^5 X_2(5) + W_6^{10} X_3(5)$$

Figure E6.33(b) show the decimation-in-time FFT flow diagram for $N = 6 = 3.2$.

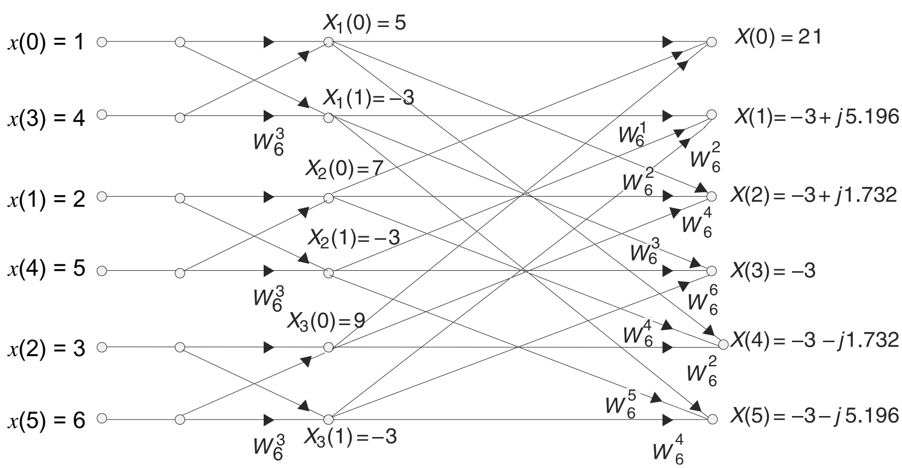


Fig. E6.33(b) *DIT FFT Flow Diagram for $N = 6 = 3.2$*

(c) To evaluate the DFT values for $x(n) = \{1, 2, 3, 4, 5, 6\}$

$$x(n) = \{1, 2, 3, 4, 5, 6\}$$

$$W_6^0 = e^{\frac{-j2\pi}{6}\cdot 0} = 1$$

$$W_6^1 = e^{\frac{-j2\pi}{6}\cdot 1} = \cos\frac{\pi}{3} - j\sin\frac{\pi}{3} = 0.5 - j0.866$$

$$W_6^2 = e^{\frac{-j2\pi}{6}\cdot 2} = -0.5 - j0.866$$

$$W_6^3 = e^{\frac{-j2\pi}{6}\cdot 3} = -1$$

$$W_6^4 = -0.5 + j0.866$$

$$W_6^5 = 0.5 + j0.866$$

$$X_1(0) = x(0) + x(3)W_6^0$$
$$= 1 + 4(1) = 5$$

$$X_1(1) = x(0) + x(3)W_6^3$$
$$= 1 + 4(-1) = -3$$

$$X_2(0) = x(1) + x(4)W_6^0$$
$$= 2 + 5(1) = 7$$

$$X_2(1) = x(1) + x(4)W_6^3$$
$$= 2 + 5(-1) = -3$$

$$X_3(0) = x(2) + x(5)W_6^0$$
$$= 3 + 6 = 9$$

$$X_3(1) = x(2) + x(5)W_6^3$$
$$= 3 + 6(-1) = -3$$

$$X(0) = X(0) + W_6^0 X_2(0) + W_6^0 X_3(0)$$
$$= 5 + 7 + 9 = 21$$

$$X(1) = X_1(1) + W_6^1 X_2(1) + W_6^2 X_3(1)$$
$$= -3 + (0.5 - j0.866)(-3) + (-0.5 - j0.866)(-3)$$
$$= -3\{1 + 0.5 - j0.866 - 0.5 - j0.866\} = -3\{1 - j1.732\} = -3 + j5.196$$

$$X(2) = X_1(2) + W_6^2 X_2(2) + W_6^4 X_3(6)$$
$$= X_1(0) + W_6^2 X_2(0) + W_6^4 X_3(0)$$
$$= 5 + (-0.5 - j0.866)(7) + (-0.5 + j0.866)(9) = -3 + j1.732$$

$$X(3) = X_1(1) + W_6^3 X_2(1) + W_6^6 X_3(1)$$
$$= -3 + (-1)(-3) + (1)(-3) = -3 + 3 - 3 = -3$$

$$X(4) = X_1(0) + W_6^4 X_2(0) + W_6^2 X_3(0)$$
$$= 5 + (-0.5 + j0.866)(7) + (-0.5 - j0.866)(9) = -3 - j1.732$$

$$X(5) = X_1(1) + W_6^5 X_2(1) + W_6^4 X_3(1)$$
$$= -3 + (0.5 + j0.866)(-3) + (-0.5 + j0.866)(-3)$$
$$= -3\{1 + 0.5 + j0.866 - 0.5 + j0.866\} = -3\{1 + j1.732\} = -3 - j5.196$$

The calculated values of DFT are also shown in Fig. E6.33(b).

Example 6.34 *Develop the DFT FFT algorithm for decomposing the DFT for $N = 12$ and draw the flow diagram.*

Solution For $N = 12 = 3.4$, where $m_1 = 3$ and $N_1 = 4$, Eq. (6.46) becomes

$$X(k) = \sum_{n=0}^{3} x(3n)W_{12}^{3nk} + \sum_{n=0}^{3} x(3n+1)W_{12}^{(3n+1)k} + \sum_{n=0}^{3} x(3n+2)W_{12}^{(3n+2)k}$$

$$= \sum_{n=0}^{3} x(3n)W_4^{nk} + W_{12}^{k}\sum_{n=0}^{3} x(3n+1)W_4^{nk} + W_{12}^{2k}\sum_{n=0}^{3} x(3n+2)W_4^{nk}$$

$$= X_1(k) + W_{12}^{k}X_2(k) + W_{12}^{2k}X_3(k)$$

Also, $X_i(k+4) = X_i(k)$

$$X_1(k) = \sum_{n=0}^{3} x(3n)W_{12}^{3nk}$$

$$= x(0) + W_{12}^{3k}x(3) + W_{12}^{6k}x(6) + W_{12}^{9k}x(9)$$

$$X_1(0) = x(0) + x(3) + x(6) + x(9)$$

$$X_1(1) = x(0) + x(3)W_{12}^{3} + W_{12}^{6}x(6) + W_{12}^{9}x(9)$$

$$X_1(2) = x(0) + W_{12}^{6}x(3) + W_{12}^{12}x(6) + W_{12}^{18}x(9)$$

$$X_1(3) = x(0) + W_{12}^{9}x(3) + W_{12}^{18}x(6) + W_{12}^{27}x(9)$$

$$X_2(k) = \sum_{n=0}^{3} x(3n+1)W_{12}^{3nk} = x(1) + x(4)W_{12}^{3k} + x(7)W_{12}^{6k} + x(10)W_{12}^{9k}$$

$$X_2(0) = x(1) + x(4) + x(7) + x(10)$$

$$X_2(1) = x(1) + W_{12}^{3}x(4) + W_{12}^{6}x(7) + W_{12}^{9}x(10)$$

$$X_2(2) = x(1) + W_{12}^{6}x(4) + W_{12}^{12}x(7) + W_{12}^{18}x(10)$$

$$X_2(3) = x(1) + W_{12}^{9}x(4) + W_{12}^{18}x(7) + W_{12}^{27}x(10)$$

$$X_3(k) = \sum_{n=0}^{3} x(3n+2)W_{12}^{3nk}$$

$$= x(2) + W_{12}^{3k}x(5) + W_{12}^{6k}x(8) + W_{12}^{9k}x(11)$$

$$X_3(0) = x(2) + x(5) + x(8) + x(11)$$

$$X_3(1) = x(2) + x(5)W_{12}^{3} + x(8)W_{12}^{6} + x(11)W_{12}^{9}$$

$$X_3(2) = x(2) + x(5)W_{12}^{6} + x(8)W_{12}^{12} + x(11)W_{12}^{18}$$

$$X_3(3) = x(2) + x(5)W_{12}^{9} + x(8)W_{12}^{18} + x(11)W_{12}^{27}$$

$$X(0) = X_1(0) + X_2(0) + X_3(0)$$

$$X(1) = X_1(1) + X_2(1)W_{12}^1 + X_3(1)W_{12}^2$$

$$X(2) = X_1(2) + X_2(2)W_{12}^2 + X_3(1)W_{12}^4$$

$$X(3) = X_1(3) + X_2(3)W_{12}^3 + X_3(3)W_{12}^6$$

$$X(4) = X_1(0) + X_2(0)W_{12}^4 + X_3(0)W_{12}^8$$

$$X(5) = X_1(1) + X_2(1)W_{12}^5 + X_3(1)W_{12}^{10}$$

$$X(6) = X_1(2) + X_2(2)W_{12}^6 + X_3(2)W_{12}^{12}$$

$$X(7) = X_1(3) + X_2(3)W_{12}^7 + X_3(3)W_{12}^{14}$$

$$X(8) = X_1(0) + X_2(0)W_{12}^8 + X_3(0)W_{12}^{16}$$

$$X(9) = X_1(1) + X_2(1)W_{12}^9 + X_3(1)W_{12}^{18}$$

$$X(10) = X_1(2) + X_2(2)W_{12}^{10} + X_3(2)W_{12}^{20}$$

$$X(11) = X_1(3) + X_2(3)W_{12}^{11} + X_3(3)W_{12}^{22}$$

The decimation-in-time FFT flow diagram for $N = 12$ is shown in Fig. E6.34.

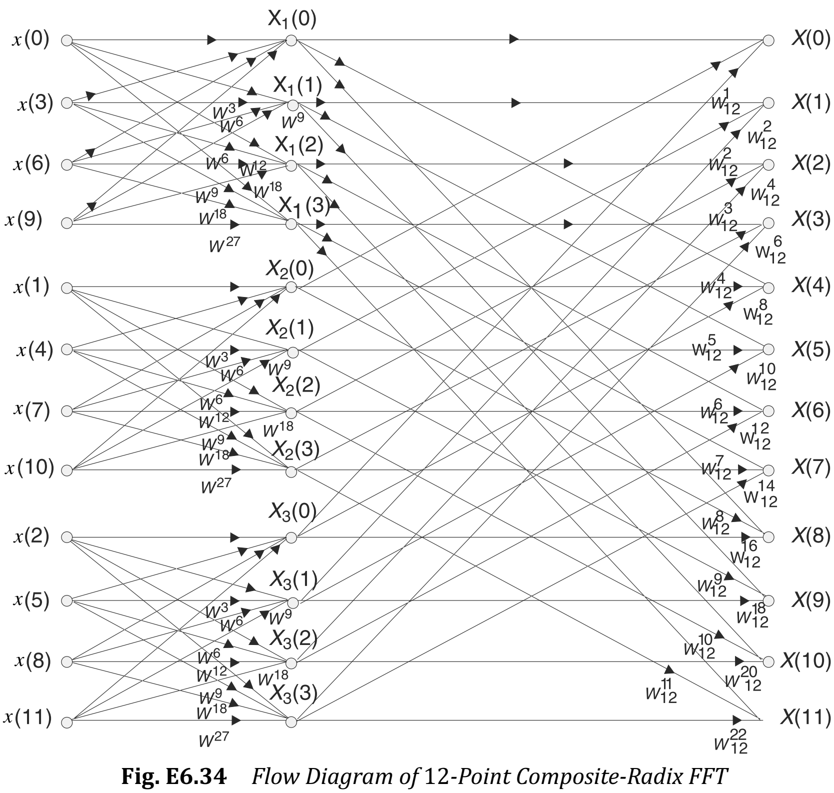


Fig. E6.34 *Flow Diagram of* 12-*Point Composite-Radix FFT*

Example 6.35 *Develop a radix-4 DIT FFT algorithm for evaluating the DFT for N = 16 and hence determine the 16-point DFT of the sequence*

$$x(n) = \{0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1\}$$

Solution

For $N = 16 = 4.4$, the Eq. (6.46) becomes

$$X(k) = \sum_{n=0}^{3} x(4n+2)W_{16}^{4nk} + \sum_{n=0}^{3} x(4n+1)W_{16}^{(4n+1)k}$$
$$+\sum_{n=0}^{3} x(4n+2)W_{16}^{(4n+2)k} + \sum_{n=0}^{3} x(4n+3)W_{N}^{(4n+3)k}$$
$$= X_1(k) + W_{16}^{k} X_2(k) + W_{16}^{2k} X_3(k) + W_{16}^{3k} X_4(k)$$

where

$$X_1(k) = \sum_{n=0}^{1} x(4n)W_9^{4nk} = x(0) + x(4)W_{16}^{4k} + x(8)W_{16}^{8k} + x(12)W_{16}^{12k}$$
$$X_2(k) = \sum_{n=0}^{3} x(4n+1)W_9^{4nk} = x(1) + x(5)W_{16}^{4k} + x(9)W_{16}^{8k} + x(13)W_{16}^{12k}$$
$$X_3(k) = \sum_{n=0}^{3} x(4n+2)W_{16}^{4nk} = x(2) + x(6)W_{16}^{4k} + x(10)W_{16}^{8k} + x(14)W_{16}^{12k}$$
$$X_4(k) = \sum_{n=0}^{3} x(4n+3)W_{16}^{4nk} = x(3) + x(7)W_{16}^{4k} + x(11)W_{16}^{8k} + x(15)W_{16}^{12k}$$

Also, $X_i(k+4) = X_i(k)$

Therefore,

$$X(0) = X_1(0) + X_2(0) + X_3(0) + X_4(0)$$
$$X(1) = X_1(1) + W_{16}^{1} X_2(1) + W_{16}^{2} X_3(1) W_{16}^{3} X_4(1)$$
$$X(2) = X_1(2) + W_{16}^{2} X_2(2) + W_{16}^{4} X_3(2) W_{16}^{6} X_4(2)$$
$$X(3) = X_1(3) + W_{16}^{3} X_2(3) + W_{16}^{6} X_3(3) W_{16}^{9} X_4(3)$$
$$X(4) = X_1(0) + W_{16}^{4} X_2(0) + W_{16}^{8} X_3(0) W_{16}^{12} X_4(0)$$
$$X(5) = X_1(1) + W_{16}^{5} X_2(1) + W_{16}^{10} X_3(1) W_{16}^{15} X_4(1)$$
$$X(6) = X_1(2) + W_{16}^{6} X_2(2) + W_{16}^{12} X_3(2) W_{16}^{18} X_4(2)$$
$$X(7) = X_1(3) + W_{16}^{7} X_2(3) + W_{16}^{14} X_3(3) W_{16}^{21} X_4(3)$$
$$X(8) = X_1(0) + W_{16}^{8} X_2(0) + W_{16}^{16} X_3(0) W_{16}^{24} X_4(0)$$
$$X(9) = X_1(1) + W_{16}^{9} X_2(1) + W_{16}^{18} X_3(1) + W_{16}^{27} X_4(1)$$
$$X(10) = X_1(2) + W_{16}^{10} X_2(2) + W_{16}^{20} X_3(2) W_{16}^{30} X_4(2)$$
$$X(11) = X_1(3) + W_{16}^{11} X_2(3) + W_{16}^{22} X_3(3) W_{16}^{33} X_4(3)$$
$$X(12) = X_1(0) + W_{16}^{12} X_2(0) + W_{16}^{24} X_3(0) W_{16}^{36} X_4(0)$$
$$X(13) = X_1(1) + W_{16}^{13} X_2(1) + W_{16}^{26} X_3(1) W_{16}^{39} X_4(1)$$
$$X(14) = X_1(2) + W_{16}^{14} X_2(2) + W_{16}^{28} X_3(2) W_{16}^{42} X_4(2)$$
$$X(15) = X_1(3) + W_{16}^{15} X_2(3) + W_{16}^{30} X_3(3) W_{16}^{45} X_4(3)$$

Figure E6.35 shows the radix-4 decimation-in-time FFT flow diagram for $N = 16$.

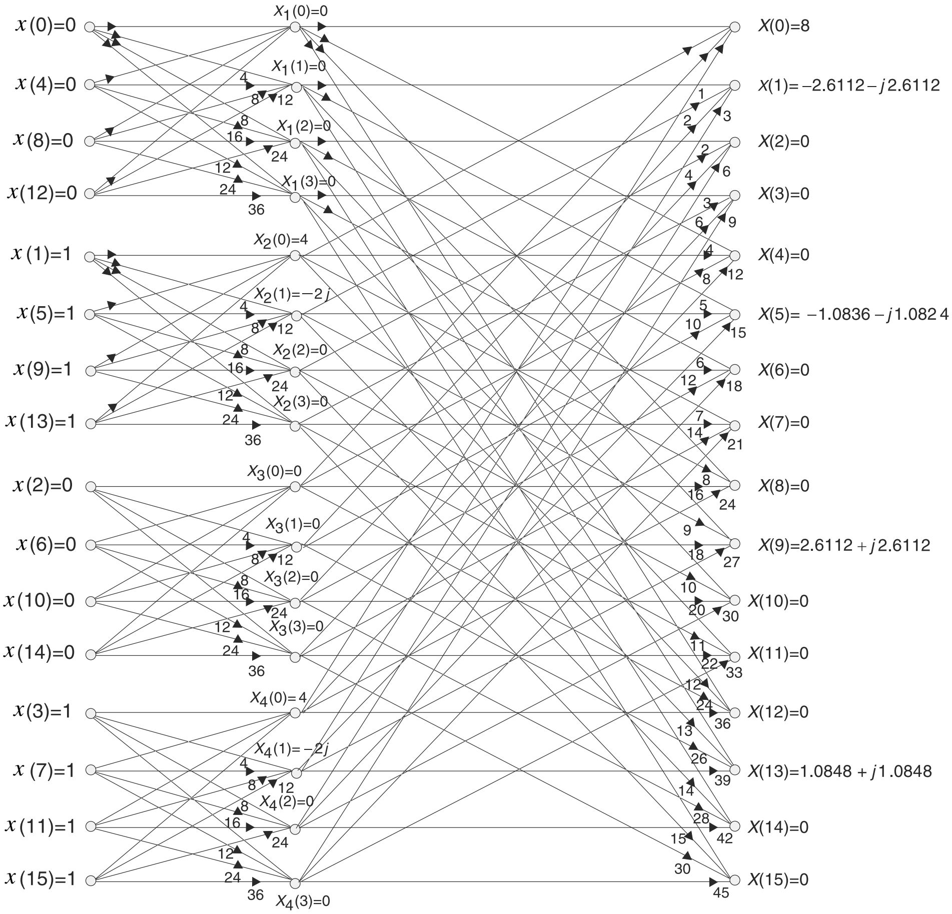


Fig. E6.35 *Radix-4 DIT FFT Flow Diagram for N=16*

To determine the DFT of the given 16-point sequence

$$x(n) = \{0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1\}$$

$$X_1(k) = x(0) + x(4)W_{16}^{4k} + x(8)W_{16}^{8k} + x(12)W_{16}^{12k} = 0 + 0 + 0 + 0 = 0$$

$$X_2(k) = x(1) + x(5)W_{16}^{4k} + x(9)W_{16}^{8k} + x(13)W_{16}^{12k}$$
$$= 1 + 1(-j) + 1(-1) + 1(-j) = -2j$$

$$X_3(k) = x(2) + x(6)W_{16}^{4k} + x(10)W_{16}^{8k} + x(14)W_{16}^{12k} = 0$$

$$X_1(0) = X_1(1) = X_1(2) = X_1(3) = 0$$

$$X_2(0) = 1 + 1 + 1 + 1 = 4$$

$$X_2(1) = x(1) + x(5)W_{16}^{4} + x(9)W_{16}^{8} + x(13)W_{16}^{12} = -j2$$

$$X_2(2) = x(1) + x(5)W_{16}^{8} + x(9)W_{16}^{16} + x(13)W_{16}^{24}$$
$$= 1 + 1(-1) + 1(1) + 1(-1) = 0$$

$$X_2(3) = x(1) + x(5)W_{16}^{12} + x(9)W_{16}^{24} + x(13)W_{16}^{36}$$
$$= 1 + 1(-j) + 1(-1) + 1(j) = 0$$

$$X_3(0) = X_3(1) = X_3(2) = X_3(3) = 0$$

$$X_4(k) = x(3) + x(7)W_{16}^{4k} + x(11)W_{16}^{8k} + x(15)W_{16}^{12k} = 1 + 1(-j) + 1(-1) + 1(-j) = -j2$$

$$X_4(0) = 4$$

$$X_4(1) = -j2$$

$$X_4(2) = 1 + 1(-1) + 1(1) + 1(-1) = 0$$

$$X_4(3) = 1 + 1(-j) + 1(-1) + 1(j) = 0$$

$$X(0) = X_1(0) + X_2(0) + X_3(0) + X_4(0) = 0 + 4 + 0 + 4 = 8$$

$$\begin{aligned} X(1) &= X_1(1) + W_{16}^1 X_2(1) + W_{16}^2 X_3(1) + W_{16}^3 X_4(1) \\ &= 0 + (0.923 - j0.382)(-j2) + (0.707 - j0.707)(0) + (0.3826 - j0.923)(-j2) \\ &= -j1.846 - 0.764 - j0.7652 - 1.846 = -2.61 - j2.61 \end{aligned}$$

$$\begin{aligned} X(2) &= X_1(2) + W_{16}^2 X_2(2) + W_{16}^4 X_3(2) + W_{16}^6 X_4(2) \\ &= 0 + 0 + 0 + 0 = 0 \end{aligned}$$

$$\begin{aligned} X(3) &= X_1(3) + W_{16}^3 X_2(3) + W_{16}^6 X_3(3) + W_{16}^9 X_4(3) \\ &= 0 + 0 + 0 + 0 = 0 \end{aligned}$$

$$\begin{aligned} X(4) &= X_1(0) + W_{16}^4 X_2(0) + W_{16}^8 X_3(0) + W_{16}^{12} X_4(0) \\ &= 0 + (-j)(4) + (-1)(0) + (j)4 = 0 \end{aligned}$$

$$\begin{aligned} X(5) &= X_1(1) + W_{16}^5 X_2(1) + W_{16}^{10} X_3(1) + W_{16}^{15} X_4(1) \\ &= 0 + (-j2)(-0.3826 - j0.9238) + (0)(-0.707 + j0.707) \\ &\quad + (-j2)(0.9238 + j0.382) \\ &= j0.7652 - 1.8476 - j1.8476 + 0.764 \\ &= -j1.0824 - 1.0836 \end{aligned}$$

$$\begin{aligned} X(6) &= X_1(2) + W_{16}^6 X_2(2) + W_{16}^{12} X_3(2) + W_{16}^{18} X_4(2) \\ &= 0 + 0 + 0 + 0 = 0 \end{aligned}$$

$$\begin{aligned} X(7) &= X_1(3) + W_{16}^7 X_2(3) + W_{16}^{14} X_3(3) + W_{16}^{21} X_4(3) \\ &= 0 + 0 + 0 + 0 = 0 \end{aligned}$$

$$\begin{aligned} X(8) &= X_1(0) + W_{16}^8 X_2(0) + W_{16}^{16} X_3(0) + W_{16}^{24} X_4(0) \\ &= 0 + 4(-1) + 0 + 4(1) = 0 \end{aligned}$$

$$\begin{aligned} X(9) &= X_1(1) + W_{16}^9 X_2(1) + W_{16}^{18} X_3(1) + W_{16}^{27} X_4(1) \\ &= 0 + (-j2)(-0.923 + j0.3826) + 0 + (-j2)(-0.382 + j0.923) \\ &= -j2\{-0.923 + j0.3826 - 0.382 + j0.923\} \\ &= -j2\{-1.3056 + j1.3056\} = 2.6112 + j2.6112 \end{aligned}$$

$$\begin{aligned} X(10) &= X_1(2) + W_{16}^{10} X_2(2) + W_{16}^{20} X_3(2) + W_{16}^{30} X_4(2) \\ &= 0 + 0 + 0 + 0 = 0 \end{aligned}$$

$$\begin{aligned} X(11) &= X_1(3) + W_{16}^{11} X_2(3) + W_{16}^{22} X_3(3) + W_{16}^{33} X_4(3) \\ &= 0 + 0 + 0 + 0 = 0 \end{aligned}$$

$$\begin{aligned} X(12) &= X_1(0) + W_{16}^{12} X_2(0) + W_{16}^{24} X_3(0) + W_{16}^{36} X_4(0) \\ &= 0 + 4(-j) + 0 + j4 = 0 \end{aligned}$$

$$X(13) = X_1(1) + W_{16}^{13} X_2(1) + W_{16}^{26} X_3(1) + W_{16}^{39} X_4(1)$$
$$= 0 + (-j2)(0.3826 + j0.923) + 0 + (-j2)(-0.9238 - j0.382)$$
$$= -j0.7652 + 1.85 + j1.85 - 0.7652 = 1.0848 + j1.0848$$
$$X(14) = X_1(2) + W_{16}^{14} X_2(2) + W_{16}^{28} X_3(2) + W_{16}^{42} X_4(2) = 0 + 0 + 0 + 0 = 0$$
$$X(15) = X_1(3) + W_{16}^{15} X_2(3) + W_{16}^{30} X_3(3) + W_{16}^{45} X_4(3) = 0 + 0 + 0 + 0 = 0$$

The computed DFT values of the given sequence are also shown in Fig. E6.34.

Example 6.36 *Develop a DIF FFT algorithm for decomposing the DFT for $N = 6$ and draw the flow diagrams for* (a) $N = 3.2$ *and* (b) $N = 2.3$.

Solution

(a) To develop DIF FFT algorithm for $N = 3.2$

$$X(k) = \sum_{n=0}^{5} x(n) W_6^{nk} = \sum_{n=0}^{2} x(n) W_6^{nk} + \sum_{n=3}^{5} x(n) W_6^{nk}$$

$$X(k) = \sum_{n=0}^{2} x(n) W_6^{nk} + \sum_{n=0}^{2} x(n+3)\, W_6^{(n+3)k}$$

$$= \sum_{n=0}^{2} [x(n) + x(n+3)\, W_6^{3k}]\, W_6^{nk}$$

$$X(2k) = \sum_{n=0}^{2} [x(n) + x(n+3) W_6^{6k}]\, W_6^{2nk} = \sum_{n=0}^{2} [x(n) + x(n+3)]\, W_6^{2nk}$$

$$X(2k+1) = \sum_{n=0}^{2} [x(n) + x(n+3)\, W_6^{3(2k+1)}]\, W_6^{(2k+1)n}$$

$$= \sum_{n=0}^{2} [x(n) - x(n+3)]\, W_6^{n} W_6^{2nk}$$

$$g(n) = x(n) + x(n+3), \quad h(n) = x(n) - x(n+3)$$

$$X(2k) = \sum_{n=0}^{2} g(n)\, W_6^{2nk},$$

$$X(2k+1) = \sum_{n=0}^{2} h(n)\, W_6^{n} W_6^{2nk}$$

$$g(0) = x(0) + x(3), \quad g(1) = x(1) + x(4), \quad g(2) = x(2) + x(5)$$
$$h(0) = x(0) - x(3), \quad h(1) = x(1) - x(4), \quad h(2) = x(2) - x(5)$$
$$X(0) = g(0) + g(1) + g(2),$$
$$X(2) = g(0) + g(1)\, W_6^2 + g(2)\, W_6^4$$
$$X(4) = g(0) + g(1)\, W_6^4 + g(2)\, W_6^8$$
$$X(1) = h(0) + h(1)\, W_6^1 + h(2)\, W_6^2$$
$$X(3) = h(0) + h(1)\, W_6^1 W_6^2 + h(2)\, W_6^2\, W_6^4$$
$$X(5) = h(0) + h(1)\, W_6^1 W_6^4 + h(2)\, W_6^2\, W_6^8$$

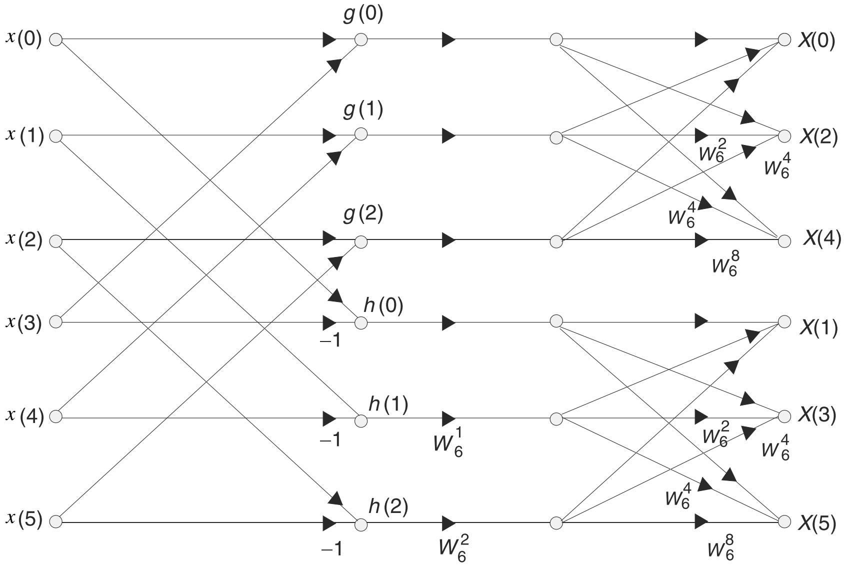


Fig. E6.36(a) *DIF FFT Flow Diagram for Decomposing the DFT for N = 6 = 3.2*

Figure E6.36(a) shows the DIF FFT flow diagram for decomposing the DFT for $N = 6$. Its input is in normal order and its output is in bit-reversed order.

(b) To develop DIF FFT algorithm for $N = 2.3$

$$X(k) = \sum_{n=0}^{5} x(n)W_6^{nk} = \sum_{n=0}^{1} x(n)W_6^{nk} + \sum_{n=2}^{3} x(n)W_6^{nk} + \sum_{n=4}^{5} x(n)W_6^{nk}$$

$$= \sum_{n=0}^{1} [x(n) + x(n+2)\, W_6^{2k} + x(n+4)W_6^{4k}]\, W_6^{nk}$$

$$X(3k) = \sum_{n=0}^{1} [x(n) + x(n+2) + x(n+4)]\, W_6^{3nk}$$

$$X(3k+1) = \sum_{n=0}^{1} [x(n) + x(n+2)\, W_6^{2} + x(n+4)\, W_6^{4}]\, W_6^{n} W_6^{3nk}$$

$$X(3k+2) = \sum_{n=0}^{1} [x(n) + x(n+2)\, W_6^{4} + x(n+4)\, W_6^{2}]\, W_6^{2n} W_6^{3nk}$$

$$f(n) = x(n) + x(n+2) + x(n+4)$$

$$g(n) = x(n) + x(n+2)W_6^2 + x(n+4)W_6^4$$

$$h(n) = x(n) + x(n+2)W_6^4 + x(n+4)W_6^2$$

$$f(0) = x(0) + x(2) + x(4), \qquad f(1) = x(1) + x(3) + x(5)$$

$$g(0) = x(0) + x(2)\, W_6^2 + x(4)\, W_6^4, \quad g(1) = x(1) + x(3)\, W_6^2 + x(5)\, W_6^4$$

$$h(0) = x(0) + x(2)\, W_6^4 + x(4)\, W_6^2, \quad h(1) = x(1) + x(3)\, W_6^4 + x(5)\, W_6^2$$

$$X(0) = f(0) + f(1)\, W_6^0, \qquad X(3) = f(0) + f(1)\, W_6^0\; W_6^3$$

$$X(1) = g(0) + g(1)\, W_6^1, \qquad X(4) = g(0) + g(1)\, W_6^1\; W_6^3$$

$$X(2) = h(0) + h(1)\, W_6^2, \qquad X(5) = h(0) + h(1)\, W_6^2\; W_6^3$$

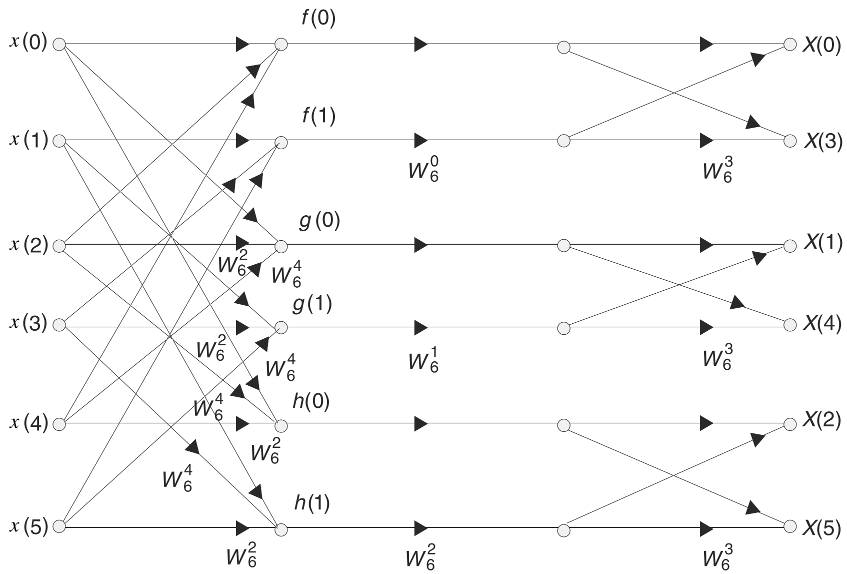


Fig. E6.36(b) *DIF FFT Flow Diagram for Decomposing the DFT for N = 6 = 2.3*

Figure E6.36 (b) shows the DIF FFT flow diagram for decomposing the DFT for $N = 6$.

Example 6.37 *Develop a Radix-3 DIF FFT algorithm for evaluating the DFT for $N = 9$.*

Solution To develop radix-3 DIF FFT algorithm for $N = 9 = 3.3$

$$X(k) = \sum_{n=0}^{2} x(n)W_9^{nk} + \sum_{n=0}^{2} x(n+3)W_9^{(n+3)k} + \sum_{n=0}^{2} x(n+6)\, W_9^{(n+6)k}$$

$$= \sum_{n=0}^{2} [x(n) + x(n+3)W_9^{3k} + x(n+6)W_9^{6k}]\, W_9^{nk}$$

$$X(3k) = \sum_{n=0}^{2} [x(n) + x(n+3) + x(n+6)]\, W_9^{3nk} = \sum_{n=0}^{2} f(n)\, W_9^{3nk} \quad (\text{since } W_N^{Nk} = 1)$$

$$X(3k+1) = \sum_{n=0}^{2} [x(n) + x(n+3)\, W_9^{3} + x(n+6)\, W_9^{6}]\, W_9^{n} W_9^{3nk}$$

$$= \sum_{n=0}^{2} g(n)\, W_9^{n} W_9^{3nk}$$

$$X(3k+2) = \sum_{n=0}^{2} [x(n) + x(n+3)\, W_9^{6} + x(n+6)W_9^{3}]\, W_9^{2n} W_9^{3nk}$$

$$= \sum_{n=0}^{2} h(n)\, W_9^{2n} W_9^{3nk}$$

$f(0) = x(0) + x(3) + x(6)$,

$f(1) = x(1) + x(4) + x(7)$, $\quad f(2) = x(2) + x(5) + x(8)$

$g(0) = x(0) + x(3)W_9^{3} + x(6)W_9^{6}$, $\quad g(1) = x(1) + x(4)W_9^{3} + x(7)W_9^{6}$

$$g(2) = x(2) + x(5)W_9^3 + x(8)W_9^6, \qquad h(0) = x(0) + x(3)W_9^6 + x(6)W_9^3$$

$$h(1) = x(1) + x(4)W_9^6 + x(7)W_9^3, \qquad h(2) = x(2) + x(5)W_9^6 + x(8)W_9^3$$

$$X(0) = f(0) + f(1) + f(2), \qquad X(3) = f(0) + f(1)W_9^3 + f(2)W_9^6$$

$$X(6) = f(0) + f(1)W_9^6 + f(2)W_9^3, \qquad X(1) = g(0) + g(1)W_9^1 + g(2)W_9^2$$

$$X(4) = g(0) + g(1)W_9W_9^3 + g(2)W_9^2W_9^6$$

$$X(7) = g(0) + g(1)W_9W_9^6 + g(2)W_9^2W_9^3$$

$$X(2) = h(0) + h(1)W_9^2 + h(2)W_9^4$$

$$X(5) = h(0) + h(1)W_9^2W_9^3 + h(2)W_9^4W_9^6$$

$$X(8) = h(0) + h(1)W_9^2W_9^6 + h(2)W_9^4W_9^3$$

Figure E6.37 shows the radix-3 DIF FFT flow diagram for decomposing the DFT for $N = 9 = 3.3$.

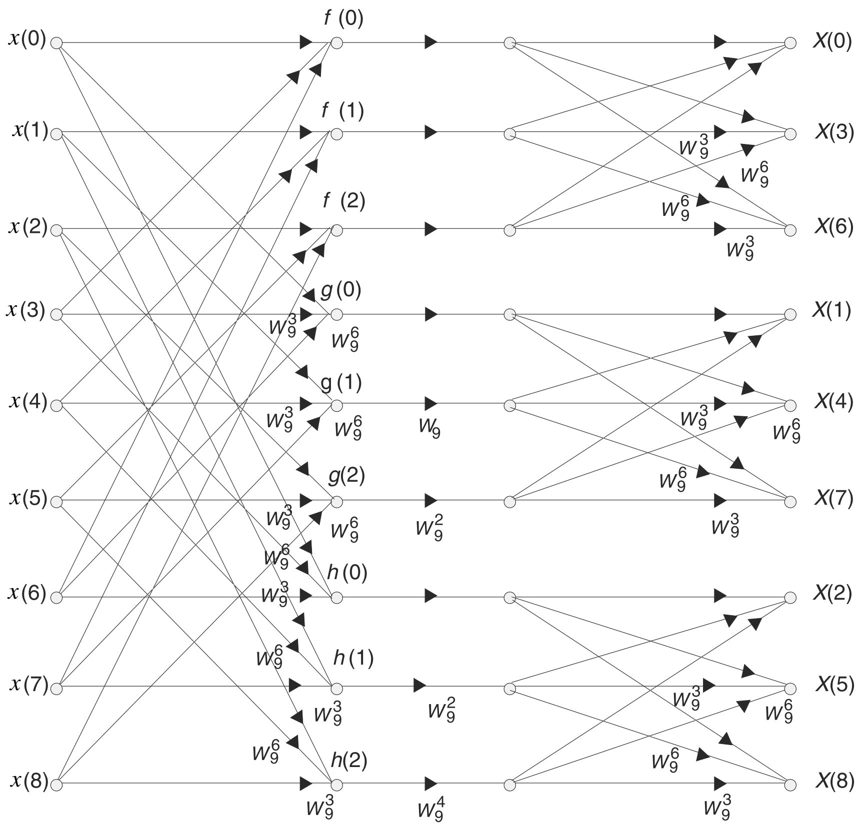


Fig. E6.37 *A Radix-3 DIF FFT Flow Diagram for Decomposing the DFT for N=9=3.3*

FAST (SECTIONED) CONVOLUTION 6.7

While implementing linear convolution in FIR filters, the input signal sequence $x(n)$ is much longer than the impulse response sequence $h(n)$ of a DSP system. Circular convolution can also be used to implement linear convolution by padding zeros. The output cannot be obtained until the entire input signal is received and hence there will be characteristic delays. Also, as the signal $N_1 + N_2 - 1$ gets longer, FFT implementation and the size of the memory needed become impractical. In order to eliminate these problems while performing filtering operation (i.e. convolution) in the frequency-domain, two signal segmentation methods, namely the **overlap−add** and the **overlap−save** techniques, can be used to perform **fast convolution** by sectioning or grouping the long input sequence into blocks or batches of samples and the final convolution output sequence can be obtained by combining the partial convolution results generated from each block.

Overlap−Add Method

Figure 6.13 shows the overlap–add fast convolution method.

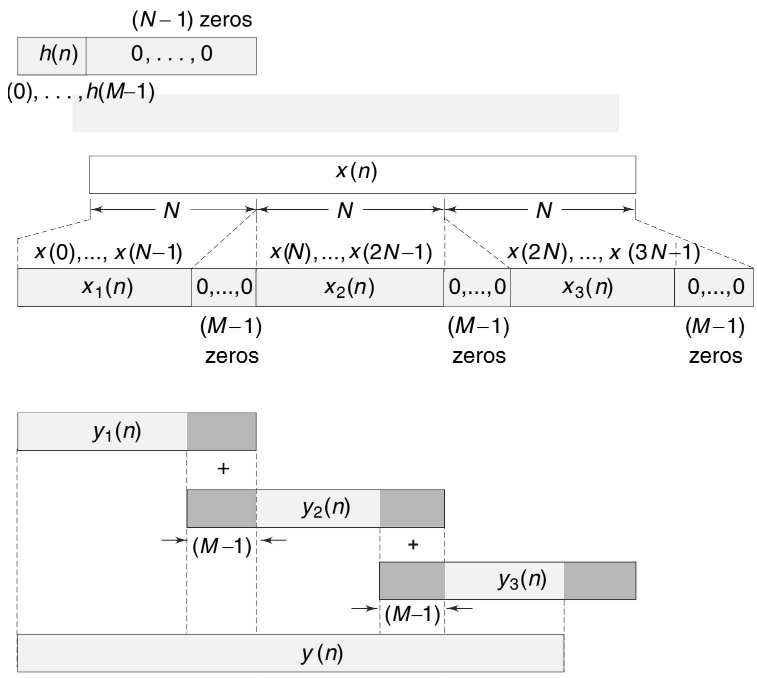


Fig. 6.13 *Overlap – Add Fast Convolution Method*

Steps Needed to Perform Overlap−Add Fast Convolution Method

(a) $(N-1)$ zeros are padded (added) at the end of the impulse response sequence $h(n)$ which is of length M and a sequence of length $M + N - 1 = L$ is obtained. Then, this L-point FFT is performed and the output values are stored.

(b) An L-point FFT on the selected data block is performed. Here each data block has N input data values and $(M-1)$ zeros.

(c) The stored frequency response of the filter, i.e. the FFT output sequence obtained in step (a) is multiplied by the FFT output sequence of the selected data block obtained in step (b).

(d) An L-point inverse FFT is performed on the product sequence obtained in step (c).

(e) The first $(M-1)$ IFFT values obtained in step (d) is overlapped with the last $(M-1)$ IFFT values for the previous block. Then addition is done to produce the final convolution output sequence $y(n)$.

(f) For the next data block, go to step (b).

Example 6.38 *An FIR digital filter has the unit impulse response sequence,* $h(n)=\{2,\ 2,\ 1\}$. *Determine the output sequence in response to the input sequence* $x(n)=\{3, 0, -2, 0, 2, 1, 0, -2, -1, 0\}$ *using the overlap – add convolution method.*

Solution The impulse response $h(n)$ has the length, $M=3$. The length of the FFT/IFFT operation is selected as $L=2^M=2^3=8$. Then, $N=L-M+1=8-3+1=6$, and the segmentation of the input sequence with the required zero padding is given in Fig. E6.38(a).

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	...
$x(n)$	3	0	−2	0	2	1	0	−2	−1	0	3	0	−2	0	...
$x_1(n)$	3	0	−2	0	2	1	0	0							
$x_2(n)$							0	−2	−1	0	3	0	0	0	
$x_3(n)$													−2	0	...

Fig. E6.38(a)

Steps (b), (c) and (d) are described below using the direct implementation of circular convolution.

Circular convolution of data blocks $x_1(n)$ and $x_2(n)$ with $h(n)$ padded with $(N-1)$, i.e. five zeros is given in Fig. E6.38(b) and (c).

$x_1(n)$	0	−2	0	2	1	0	0	3	0	−2	0	2	1	0	0
$h(-(k-n))$	0	0	0	0	0	1	2	2							
$y_1(n)$								6	6	−1	−4	2	6	4	1

(b)

$x_2(n)$	−2	−1	0	3	0	0	0	0	−2	−1	0	3	0	0	0
$h(-(k-n))$	0	0	0	0	0	1	2	2							
$y_2(n)$								0	−4	−6	−4	5	6	3	0

(c)

Fig. E6.38(b) and (c)

Overlapping each circular convolution result by $(M-1)=2$ values and adding gives the output sequence as shown in Fig. E6.38(d).

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	...
$y_1(n)$	6	6	−1	−4	2	6	4	1							
$y_2(n)$							0	−4	−6	−4	5	6	3	0	
$y_3(n)$													×	×	...
$y(n)$	6	6	−1	−4	2	6	4	−3	−6	−4	5	6	×	×	...

Fig. E6.38(d)

Overlap – Save Method

It has already been shown that multiplication of two DFTs yields a circular convolution of time-domain sequences, i.e. $y(n)=x(n)*h(n)=\text{IFFT }[X(f)\ H(f)]$. But a linear convolution is required for the implementation of an FIR filter. The overlap – save method is useful for converting a circular convolution into a linear convolution.

Figure 6.14 shows the overlap – save convolution method.

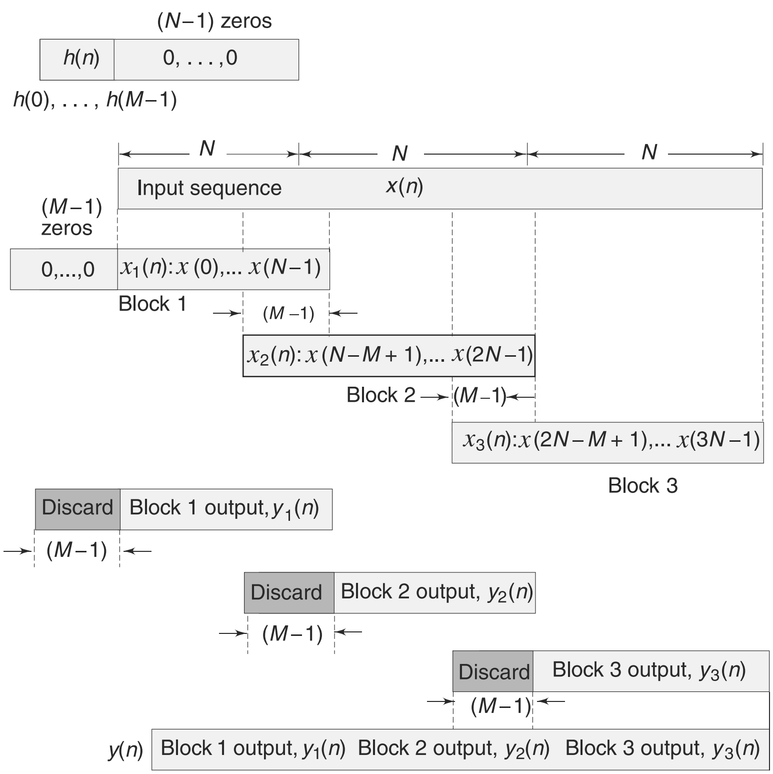


Fig. 6.14 *The Overlap – Save Convolution Method*

Steps to Perform Overlap – Save Fast Convolution Method

(a) $(N-1)$ zeros are padded (added) at the end of the impulse response sequence $h(n)$ which is of length M and a sequence of length $(M+N-1)=L$ is obtained. Then, this L-point FFT is performed and the output values are stored.

(b) An L-point FFT on the selected data block is performed. Here each data block begins with the last $(M-1)$ values in the previous data block, except the first data block which begins with $(M-1)$ zeros.

(c) The stored frequency response of the filter, i.e. the FFT output sequence obtained in step (a) is multiplied by the FFT output sequence of the selected data block obtained in step (b).

(d) An L-point inverse FFT is performed on the product sequence obtained in step (c).

(e) The first $(M-1)$ values from successive output of step (d) are discarded and the last N values of the IFFT obtained in step (d) is saved to produce the output $y(n)$.

(f) For the next data block, go to step (b).

Example 6.39 *Repeat Example 6.38 by using the overlap-save convolution method.*

Solution The impulse response $h(n)$ has length, $M=3$. The length of the FFT/IFFT operation is selected as $L=2^M=2^3=8$. Then, $N=L-M+1=8-3+1=6$, and the segmentation of the input sequence results in the data blocks given in Fig. E6.39(a).

Steps (b), (c) and (d) are described below using the direct implementation of circular convolution.

n	−2	−1	0	1	2	3	4	5	6	7	8	9	10	11	...
$x(n)$			3	0	−2	0	2	1	0	−2	−1	0	3	0	...
$x_1(n)$	0	0	3	0	−2	0	2	1							
$x_2(n)$							2	1	0	−2	−1	0	3	0	
$x_3(n)$													3	0	...

Fig. E6.39(a)

Circular convolution of data blocks $x_1(n)$ and $x_2(n)$ with $h(n)$ padded with $(n-1)$, i.e. five zeros is given in Fig. E6.39(b) and (c).

Discarding the first two values and saving the last six values of each circular convolution result gives the output sequence as shown in Fig. E6.39(d).

This output sequence is similar to the results obtained in Example 6.38.

$x_1(n)$	0	3	0	−2	0	2	1	0	0	3	0	−2	0	2	1
$h(-(k-n))$	0	0	0	0	0	1	2	2							
$y_1(n)$								4	1	6	6	−1	−4	2	6

(b)

x_2(n)	1	0	−2	−1	0	3	0	2	1	0	−2	−1	0	3	0
$h(-(k-n))$	0	0	0	0	0	1	2	2							
$y_2(n)$								7	5	4	−3	−6	−4	5	6

(c)

n	−2	−1	0	1	2	3	4	5	6	7	8	9	10	11	…
$y_1(n)$	4	1	6	6	−1	−4	2	6							
$y_2(n)$							7	5	4	−3	−6	−4	5	6	
$y_3(n)$													×	×	…
$y(n)$			6	6	−1	−4	2	6	4	−3	−6	−4	5	6	…

(d)

Fig. E6.39 (b), (c) and (d)

CORRELATION 6.8

Correlation gives a measure of similarity between two data sequences. In this process, two signal sequences are compared and the degree to which the two signals are similar is computed. Typical applications of correlation include speech processing, image processing, sonar and radar systems. Typically, in a radar system, the transmitted signal is correlated with the echo signal to locate the position of the target. Similarly, in speech processing systems, different speech waveforms are compared for voice recognition.

6.8.1 Cross-Correlation Sequences

For two sequences $x(n)$ and $y(n)$, the cross-correlation function, $r_{xy}(l)$, is defined as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l), \quad l = 0, \pm 1, \pm 2, ... \tag{6.47}$$

or equivalently,

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n+l)y(n), \quad l = 0, \pm 1, \pm 2, ... \tag{6.48}$$

where index l is the (time) shift (or lag) parameter and the subscripts xy on the cross-correlation sequence $r_{xy}(l)$ show the sequences being correlated. The order of the subscripts, with x preceding y in Eq. (6.47) indicates that $x(n)$ is kept unshifted and $y(n)$ is shifted by l units in time, to the right for l

positive and to the left for l negative. Similarly, in Eq. (6.48), $y(n)$ is kept unshifted and $x(n)$ is shifted by l units in time, to the left for l positive and to the right for l negative.

When the roles of $x(n)$ and $y(n)$ are reversed, the cross-correlation sequence becomes

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} y(n)\,x(n-l) \tag{6.49}$$

or equivalently,

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} y(n+l)\,x(n) \tag{6.50}$$

Comparing Eq. (6.47) with Eq. (6.50) or Eq. (6.48) with Eq. (6.49), we find that

$$r_{xy}(l) = r_{yx}(-l) \tag{6.51}$$

This means that $r_{yx}(l)$ is the folded version of $r_{xy}(l)$, with respect to $l = 0$. Therefore, $r_{yx}(l)$ gives exactly the same information as $r_{xy}(l)$.

6.8.2 Autocorrelation Sequences

When $y(n) = x(n)$, the cross-correlation function becomes the auto-correlation function. As a result, $y(n)$ is replaced by $x(n)$ in Eq. (6.47) and Eq. (6.48) gives the autocorrelation function, $r_{xx}(l)$, which is defined a

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n)\,x(n-l), \quad l = 0, \pm 1, \pm 2, \ldots \tag{6.52}$$

or equivalently,

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n+l)\,x(n), \quad l = 0, \pm 1, \pm 2, \ldots \tag{6.53}$$

From the above equations, it is clear that the maximum autocorrelation value occurs at $l = 0$ because of an in-phase relationship between the two sequences. As l increases, the autocorrelation value increases.

Example 6.40 *Determine the cross-correlation values of the two sequences* $x(n) = \{1, 0, 0, 1\}$ *and* $h(n) = \{4, 3, 2, 1\}$.

Solution Figure E6.40 shows the computation of the cross-correlation values for the given two sequences.

Circular Correlation of Periodic Sequences

Like circular convolution, circular correlation also processes two periodic sequences having the same length. If the two sequences are periodic with the same period N, then the circular correlation is defined as

$$r_{xy}(l) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\,y(n-l),\ l = 0, 1, \ldots, (N-1) \tag{6.54}$$

n	−3	−2	−1	0	1	2	3	4	5	6	≥7		
$x(n)$	0	0	0	1	0	0	1	0	0	0	0	l	$r_{xy}(l)$
$h(n+3)$	4	3	2	1	0	0	0	0	0	0	0	−3	[0×4 + 0×3 + 0×2 + 1×1]/4 = 0.25
$h(n+2)$	0	4	3	2	1	0	0	0	0	0	0	−2	[0×4 + 0×3 + 1×2 + 0×1]/4 = 0.5
$h(n+1)$	0	0	4	3	2	1	0	0	0	0	0	−1	[0×4 + 1×3 + 1×2 + 0×1]/4 = 0.75
$h(n)$	0	0	0	4	3	2	1	0	0	0	0	0	[1×4 + 0×3 + 0×2 +1×1]/4 = 1.25
$h(n-1)$	0	0	0	0	4	3	2	1	0	0	0	1	[0×4 + 0×3 + 1×2 + 0×1]/4 = 0.5
$h(n-2)$	0	0	0	0	0	4	3	2	1	0	0	2	[0×4 + 1×3 + 0×2 +0×1]/4 = 0.75
$h(n-3)$	0	0	0	0	0	0	4	3	2	1	0	3	[1×4 + 0×3 + 0×2 + 0×1]/4 = 1
$h(n-4)$	0	0	0	0	0	0	0	4	3	2	1	4	[0×4 + 0×3 + 0×2 + 0×1]/4 = 0

Fig. E6.40

Example 6.41 *Determine the circular correlation values of the two sequences* $x(n) = \{1, 0, 0, 1\}$ *and* $h(n) = \{4, 3, 2, 1\}$.

Solution Figure E6.41 shows the computation of the circular correlation values for the given two sequences.

n	0	1	2	3	
$x(n)$	1	0	0	1	$r_{xy}(l)$
$h(n)$	4	3	2	1	(1 × 4 + 0 × 3 + 0 × 2 + 1 × 1)/4 = 1.25
$h(n–1)$	1	4	3	2	(1 × 1 + 0 × 4 + 0 × 3 + 1 × 2)/4 = 0.75
$h(n–2)$	2	1	4	3	(1 × 2 + 0 × 1 + 0 × 4 + 1 × 3)/4 = 1.25
$h(n–3)$	3	2	1	4	(1 × 3 + 0 × 2 + 0 × 1 + 1 × 4)/4 = 1.75

Fig. E6.41

REVIEW QUESTIONS

6.1 State the symmetry property of Discrete Fourier Series.
6.2 State the complex convolution theorem.
6.3 State the periodic convolution of two discrete sequences.

6.4 Distinguish between linear and circular convolutions of two sequences.
6.5 How is the periodic convolution of two sequences $x_1(n)$ and $x_2(n)$ performed?
6.6 Give the steps to get the result of linear convolution from the method of circular convolution.
6.7 Distinguish between circular convolution and linear convolution with an example.
6.8 Define DFT for a sequence $x(n)$.
6.9 Explain how cyclic convolution of two periodic sequences can be obtained using DFT techniques.
6.10 Name any two properties of DFT.
6.11 What are 'twiddle factors' of the DFT?
6.12 State the shifting property of the DFT.
6.13 What is the difference between discrete Fourier series and discrete Fourier transform?
6.14 State the relationship between DTFT and z-transform.
6.15 Explain the difference between DTFT and DFT.
6.16 Explain the meaning of 'frequency resolution' of the DFT.
6.17 Give the number of complex additions and complex multiplications required for the direct computation of N-point DFT.
6.18 What is the need for FFT algorithm?
6.19 Why is FFT called so?
6.20 State the computational requirements of FFT.
6.21 Compare the number of multiplications required to compute the DFT of a 64-point sequence using direct computation and that using FFT.
6.22 What is DIT FFT algorithm?
6.23 What is DIF FFT algorithm?
6.24 Which FFT algorithm procedure is the lowest possible level of DFT decomposition?
6.25 Give the computation efficiency of FFT over DFT.
6.26 How can you compute DFT using FFT algorithm?
6.27 Find the exponential form of the DFS for the signal $x(n)$ given by
$x(n) = \{\ldots 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, \ldots\}$
Ans: $y(n) = \{10, 11, 7, 9, 14, 8, 5, 2\}$
6.28 An input sequence $x(n) = \{2, 1, 0, 1, 2\}$ is applied to a DSP system having an impulse sequence $h(n) = \{5, 3, 2, 1\}$. Determine the output sequence produced by (a) linear convolution and (b) verify the same through circular convolution.
Ans: $y(n) = \{10, 11, 7, 9, 14, 8, 5, 2\}$
6.29 Given two sequences of length $N = 4$ defined by $x_1(n) = (1, 2, 2, 1)$ and $x_2(n) = (2, 1, 1, 2)$, determine the periodic convolution.
Ans: $y(n) = (9, 10, 9, 8)$
6.30 Find the linear convolution through circular convolution of $x_1(n)$ and $x_2(n)$.
$x_1(n) = \delta(n) + \delta(n-1) + \delta(n-2)$
$x_2(n) = 2\delta(n) - \delta(n-1) + 2\delta(n-2)$
Ans: $x_3(n) = 2\delta(n) + \delta(n-1) + 3\delta(n-2) + \delta(n-3) + 2\delta(n-4)$
6.31 Convolve the following sequences using (a) overlap – add method and (b) overlap – save method.
$x(n) = (1 - 1, 2, 1, 2, -1, 1, 3, 1)$ and $h(n) = (1, 2, 1)$
Ans: $y(n) = (1, 1, 1, 4, 6, 4, 1, 4, 8, 5, 1)$
6.32 Obtain the discrete Fourier transform of $f(n) = (72, -56, 15, 9)$.
6.33 Show that the DFT of a sequence $x(n)$ is purely imaginary and odd if the sequence $x(n)$ is real and odd.

6.34 Find the DTFT of the sequence $x(n) = \begin{cases} \cos(\pi n/3), & -4 \le n \le 4 \\ 0, & \text{oherwise} \end{cases}$

6.35 Determine the 8-point DFT of the sequence.

$$x(n) = \begin{cases} 1, & -3 \le n \le 3 \\ 0, & \text{otherwise} \end{cases}$$

Ans: $X(k) = 2e^{-j\left(\frac{2\pi k}{8}\right)}\cos\left(\frac{\pi k}{8}\right) + 2e^{-j\left(\frac{9\pi k}{8}\right)}\cos\left(\frac{5\pi k}{8}\right)$

6.36 Determine the DFT of the sequence $h(n) = \begin{cases} \frac{1}{3} & 0 \le n \le 2 \\ 0, & \text{otherwise} \end{cases}$

Ans: $H(k) = \frac{1}{3}\left[1 + 2\cos\left(\frac{2\pi k}{3}\right)\right] e^{-j\left(\frac{2\pi k}{3}\right)}$

6.37 Determine the DFT of the sequence $x(n) = \begin{cases} 1, & 2 \le n \le 6 \\ 0, & \text{otherwise} \end{cases}$ assuming that $N = 10$.

Ans: $X(k) = e^{-j\left(\frac{4\pi k}{5}\right)} \frac{\sin\left(\frac{\pi k}{2}\right)}{\sin\left(\frac{\pi k}{10}\right)}$

6.38 For the 8-sample sequence $x(n) = \{1, 2, 3, 5, 5, 3, 2, 1\}$, the first five DFT coefficients are $\{22, -7.5355 - j3.1213, 1 + j, -0.4645 - j1.1213, 0\}$. Determine the remaining three DFT coefficients.

Ans: $X(5) = -0.4645 + j1.1213$, $X(6) = 1 - j$,
$X(7) = -7.5355 + j\,3.1213$

6.39 Compute the DFTs of the following sequences, where $N = 4$ using DIT algorithm.

(a) $x(n) = 2^n$, (b) $x(n) = 2^{-n}$

(c) $x(n) = \sin\left(\frac{n\pi}{2}\right)$, (d) $x(n) = \cos\left(\frac{n\pi}{10}\right)$

Ans: (a) $X(k) = \{15, -3 + j6, -5, -3 - j6\}$

(b) $X(k) = \left(\frac{15}{8}, \frac{3}{4}, -j\frac{3}{8}, \frac{5}{8}, \frac{3}{4} + j\frac{3}{8}\right)$

(c) $X(k) = (0, -j2, 0, j2)$

(d) $X(k) = (1, 1 - j\sqrt{2}, 1, 1 + j\sqrt{2})$

6.40 Draw the butterfly line diagram for 8-point FFT calculation and briefly explain. Use decimation-in-time.

6.41 Find the DFT of the following sequence $x(n)$ using DIT FFT.

$x(n) = (1, -1, -1, -1, 1, 1, 1, -1)$

Ans: $(0, -\sqrt{2} + j3.4142,\ 2 - j2,\ \sqrt{2} - j0.5858,$
$4, \sqrt{2} + j0.5858,\ 2 + j2,\ -\sqrt{2} - j3.4142)$

6.42 Compute the 16-point DFT of the sequence

$x(n) = \cos(\pi/2)$, $0 \le n \le 15$ using DIT algorithm.

6.43 Find DFT (8-point) for a continuous time signal
$x(t) = \sin(2\pi ft)$ with $f = 100$ Hz.

6.44 Compute the DFT of the sequence $x(n) = a^n$, where $N = 8$ and $a = 3$.

6.45 Compute the FFT for the sequence $x(n) = n^2 + 1$ where $N = 8$ using DIT algorithm.

Ans: $X(k) = 100(1.48, -0.4686 + j0.7725, -0.24 + j0.32, -0.2731 + j0.1325, -0.28, -0.2731 - j0.1325, -0.24 - j0.32, -0.4686 - j0.7725)$

6.46 Compute the FFT for the sequence $x(n) = n + 1$ where $N = 8$ using DIT algorithm.

Ans: $X(k) = (36, -4 + j9.656, -4 + j4, -4 + j1.6568, -4, -4 - j1.6568, -4 - j4, -4 - j9.656)$

6.47 Compute the DFT coefficients of a finite duration sequence (0, 1, 2, 3, 0, 0, 0, 0).

Ans. $X(k) = (6, -\sqrt{2} - j4.8284, -2 + j2,$
$\sqrt{2} - j0.8284, -2, \sqrt{2} + j0.8284, -2 - j2, -\sqrt{2} + j4.8284)$

6.48 Draw the flow graph of an 8-point DIF FFT and explain.

6.49 Draw the butterfly line diagram for 8-point FFT calculation and briefly explain. Use decimation-in-frequency.

6.50 Repeat Q6.39 using DIF algorithm.

Ans: Same as in Q6.39.

6.51 Draw the butterfly diagram for 16-point FFT calculation and briefly explain. Use decimation in frequency.

6.52 Discuss the computational efficiency of radix-2 FFT algorithm.

6.53 Develop a DIT FFT algorithm for $N = 12 = 3.2.2$ and draw the signal flow diagram.

6.54 Develop DIT FFT algorithms for (a) $N = 18 = 3.6$ and (b) $N = 18 = 3.3.2$ and draw the signal flow diagrams.

6.55 Construct a signal flow chart for DIF algorithm if $N = 6$. Derive the steps involved in the procedure.

Ans: Refer to Fig. E6.33 (a) and (b)

6.56 Develop the decimation-in-time and decimation-in-frequency FFT algorithms for decomposing the DFT for $N = 27 = 3.3.3$ and obtain the corresponding signal flow graphs.

6.57 Develop a DIT FFT algorithm for $N = 8$ using a 4-point DFT and a 2-point DFT. Compare the number of multiplications with the algorithm using only 2-point DFTs.

6.58 Explain how you would use the FFT algorithm to compute the IDFT.

6.59 Compute the inverse DFT for the sequence $X(k) = e^{-k}$ for $k = 16$.

6.60 Compute the inverse DFT for the sequence $X(k) = 2^{-k}$ where $k = 0$ to 7 using FFT algorithm.

6.61 The IDFT is given as $x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)W_N^{-nk}$. If the DIT algorithm is known, can we use it to find $x(n)$ when the sequence $X(k)$ is known?

6.62 What are the advantages of fast (sectioned) convolution?

6.63 What are the methods used to perform fast convolution? Explain any one method giving all the steps involved to perform fast convolution.

6.64 With a suitable example, show that the results obtained by both overlap – add and overlap – save fast convolution methods are identical.

6.65 What are the applications of correlation?

6.66 Define and explain the following terms: (a) Auto-correlation, (b) cross-correlation, and (c) circular-correlation.

Chapter 7 Finite Impulse Response (FIR) Filters

INTRODUCTION 7.1

A filter is essentially a network that selectively changes the waveshape of a signal in a desired manner. The objective of filtering is to improve the quality of a signal (for example, to remove noise) or to extract information from signals.

A digital filter is a mathematical algorithm implemented in hardware/software that operates on a digital input to produce a digital output. Digital filters often operate on digitised analog signals stored in a computer memory. Digital filters play very important roles in DSP. Compared with analog filters, they are preferred in a number of applications like data compression, speech processing, image processing, etc., because of the following advantages.

(i) Digital filters can have characteristics which are not possible with analog filters such as linear phase response.
(ii) The performance of digital filters does not vary with environmental changes, for example, thermal variations.
(iii) The frequency response of a digital filter can be adjusted if it is implemented using a programmable processor.
(iv) Several input signals can be filtered by one digital filter without the need to replicate the hardware.
(v) Digital filters can be used at very low frequencies.

The following are the main disadvantages of digital filters compared with analog filters:

(i) Speed limitation
(ii) Finite word length effects
(iii) Long design and development times

A discrete-time filter produces a discrete-time output sequence $y(n)$ for the discrete-time input sequence $x(n)$. A filter may be required to have a given frequency response, or a specific response to an impulse, step, or ramp, or simulate an analog system. Digital filters are classified either as **finite duration unit pulse response** (FIR) filters or **infinite duration unit pulse response** (IIR) filters, depending on the form of the unit pulse response of the system. In the FIR system, the impulse response sequence is of finite duration, i.e. it has a finite number of non-zero terms. The IIR system has an infinite number of non-zero terms, i.e. its impulse response sequence is of infinite duration. The system with the impulse response

$$h(n) = \begin{cases} 2, & |n| \le 4 \\ 0, & \text{otherwise} \end{cases}$$

has only a finite number of non-zero terms. Thus, the system is an FIR system. The system with the impulse response $h(n) = a^n\, u(n)$ is non-zero for $n \geq 0$. It has an infinite number of non-zero terms and is an IIR system. IIR filters are usually implemented using structures having feedback (recursive structures – poles and zeros) and FIR filters are usually implemented using structures with no feedback (non-recursive structures – all zeros).

Suppose a system has the following difference equation representation with input $x(n)$ and output $y(n)$

$$\sum_{k=0}^{M} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k)$$

An FIR filter of length M is described by the difference equation

$$y(n) = b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) + b_3 x(n-3) + \cdots$$
$$+ b_{M-1} x(n-M+1) = \sum_{k=0}^{M-1} b_k x(n-k)$$

where $\{b_k\}$ is the set of filter coefficients. As seen from the above equation, the response of the FIR filter depends only on the present and past input samples, whereas for the IIR filter, the present response is a function of the present and past M values of the excitation as well as past values of the response.

FIR filters have the following advantages over IIR filters:

(i) They have linear phase characteristics.
(ii) FIR filters, realised non-recursively are always stable.
(iii) The design methods are generally linear.
(iv) They can be realised efficienty in hardware
(v) The filter start-up transients have finite duration.
(vi) They have low sensitivity to finite word-length effects.

FIR filters are employed in filtering problems where linear phase characteristics within the passband of the filter is required. If this is not required, either an IIR or an FIR filter may be employed. An IIR filter has lesser number of side lobes in the stopband than an FIR filter with the same number of parameters. For this reason if some phase distortion is tolerable, an IIR filter is preferable. Also, the implementation of an IIR filter involves fewer parameters, less memory requirements and lower computational complexity.

7.2 MAGNITUDE RESPONSE AND PHASE RESPONSE OF DIGITAL FILTERS

The discrete-time Fourier transform of a finite sequence impulse response $h(n)$ is given by

$$H(e^{j\omega}) = \sum_{n=0}^{M-1} h(n) e^{-j\omega n} = \left|H(e^{j\omega})\right| e^{j\Phi(\omega)} \tag{7.1}$$

The magnitude and phase responses are given by

$$M(\omega) = \left|H(e^{j\omega})\right| = \{\mathrm{Re}[H(e^{j\omega})]^2 + \mathrm{Im}[H(e^{j\omega})]^2\}^{0.5}$$

$$\Phi(\omega) = \tan^{-1} \frac{\mathrm{Im}[H(e^{j\omega})]}{\mathrm{Re}[H(e^{j\omega})]} \tag{7.2}$$

Filters can have a linear or non-linear phase depending upon the delay function, namely the phase delay and group delay.

The phase and group delays of the filter are given by

$$\tau = \tau_p = -\frac{\Phi(\omega)}{\omega} \quad \text{and} \quad \tau_g = -\frac{d\Phi(\omega)}{d\omega}, \text{ respectively}.$$

The group delay is defined as the delayed response of the filter as a function of ω to a signal.

Linear phase filters are those filters in which the phase delay and group delay are constants, i.e. independent of frequency. Linear phase filters are also called constant time delay filters. Let us obtain the conditions FIR filters must satisfy in order to have constant phase and group delays and hence obtain the conditions for having a linear phase. For the phase response to be linear,

$$\frac{\Phi(\omega)}{\omega} = -\tau, \quad -\pi \le \omega \le \pi$$

Therefore, $\Phi(\omega) = -\omega\tau$

where τ is a constant phase delay expressed in number of samples. Using Eq. (7.2),

$$\Phi(\ \ ) = \tan^{-1}\frac{\operatorname{Im} H(e^{j\omega})}{\operatorname{Re} H(e^{j\omega})} = -\omega\tau$$

or

$$\omega\tau = \tan^{-1}\frac{\sum_{n=0}^{M-1} h(n)\sin\omega n}{\sum_{n=0}^{M-1} h(n)\cos\omega n}$$

or

$$\tan\omega\tau = \frac{\sum_{n=0}^{M-1} h(n)\sin\omega n}{\sum_{n=0}^{M-1} h(n)\cos\omega n}$$

or

$$\frac{\sin\omega\tau}{\cos\omega\tau} = \frac{\sum_{n=0}^{M-1} h(n)\sin\omega n}{\sum_{n=0}^{M-1} h(n)\cos\omega n}$$

i.e., $$\sum_{n=0}^{M-1} h(n)\cos\omega n\sin\omega\tau = \sum_{n=0}^{M-1} h(n)\sin\omega n\cos\omega\tau$$

i.e., $$\sum_{n=0}^{M-1} h(n)[\sin\omega\tau\cos\omega n - \cos\omega\tau\sin\omega n] = 0$$

Therefore,

$$\sum_{n=0}^{M-1} h(n)\sin(\omega\tau - \omega n) = 0 \tag{7.3}$$

and a solution to Eq. (7.3) is given by

$$\tau = \frac{(M-1)}{2} \tag{7.4}$$

and

$$h(n) = h(M-1-n) \text{ for } 0 < n < M-1 \tag{7.5}$$

If Eqs (7.4) and (7.5) are satisfied, then the FIR filter will have constant phase and group delays and thus the phase of the filter will be linear. The phase and group delays of the linear phase FIR filter are equal and constant over the frequency band. Whenever a constant group delay alone is preferred, the impulse response will be of the form

$$h(n) = -h(M-1-n)$$

and is antisymmetric about the centre of the impulse response sequence. The applications of FIR filters like the wideband differentiator and Hilbert transformer use such antisymmetric impulse response sequences.

Example 7.1 *The length of an FIR filter is 9. If the filter has a linear phase, show that Eq. (7.3) is satisfied.*

Solution The length of the filter, $M = 9$. Therefore, from Eq. (7.4),

$$\tau = \frac{(M-1)}{2} = 4$$

From Eq. (7.5), $h(n) = h(M-1-n)$. Therefore the filter coefficients are $h(0) = h(8)$, $h(1) = h(7)$, $h(2) = h(6)$, $h(3) = h(5)$ and $h(4)$.

Equation (7.3) can be written as,

$$\sum_{n=0}^{M-1} h(n)\sin(\omega\tau - \omega n) = \sum_{n=0}^{8} h(n)\sin\omega(\tau - n)$$

$$= h(0)\sin 4\omega + h(1)\sin 3\omega + h(2)\sin 2\omega + h(3)\sin\omega + h(4)\sin 0 + h(5)\sin(-\omega) + h(6)\sin(-2\omega) + h(7)\sin(-3\omega) + h(8)\sin(-4\omega) = 0$$

Thus Eq. (7.3) is satisfied.

Example 7.2 *The following transfer function characterises an FIR filter ($M = 11$). Determine the magnitude response and show that the phase and group delays are constant.*

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

Solution The transfer function of the filter is given by

$$H(z) = \sum_{n=0}^{M-1} h(n) z^{-n} = h(0) + h(1)z^{-1} + h(2)z^{-2} + \cdots + h(10)z^{-10}$$

The phase delay, $\tau = \dfrac{M-1}{2} = 5$. Since $\tau = 5$, the transfer function can be expressed as,

$$H(z) = z^{-5}\,[h(0)\,z^{5} + h(1)\,z^{4} + h(2)\,z^{3} + \cdots + h(9)\,z^{-4} + h(10)\,z^{-5}]$$

Since $h(n) = h(M-1-n)$,

$$H(z) = z^{-5}\,[h(0)\,(z^{5} + z^{-5}) + h(1)\,(z^{4} + z^{-4}) + h(2)\,(z^{3} + z^{-3}) + h(3)\,(z^{2} + z^{-2}) + h(4)\,(z + z^{-1}) + h(5)]$$

The frequency response is obtained by replacing z with $e^{j\omega}$,

$$H(e^{j\omega}) = e^{-j5\omega}\{h(0)[e^{5j\omega} + e^{-5j\omega}] + h(1)[e^{4j\omega} + e^{-4j\omega}] + h(2)[e^{3j\omega} + e^{-3j\omega}] + h(3)[e^{2j\omega} + e^{-2j\omega}] + h(4)[e^{j\omega} + e^{-j\omega}] + h(5)\}$$

$$= e^{-j5\omega}\left[h(5) + 2\sum_{n=0}^{4} h(n)\cos(\tau - n)\right] = e^{-j5\omega} M(\omega)$$

where $M(\omega)$ is the magnitude response and $\phi(\omega) = -5\omega$ is the phase response. The group delay, τ_g, is given by

$$\tau_p = -\frac{(\omega)}{\omega} = 5 \quad \text{and} \quad \tau_g = -\frac{d[\Phi(\omega)]}{d\omega} = -\frac{d(-5\omega)}{d\omega} = 5$$

Thus, the phase delay and the group delay are same and are constants.

FREQUENCY RESPONSE OF LINEAR PHASE FIR FILTERS 7.3

The discrete-time Fourier transform of the impulse response $h(n)$ is given by Eq. (7.1). If the filter length M is odd, then Eq. (7.1) can be written as

$$H(e^{j\omega}) = \sum_{n=0}^{\frac{M-3}{2}} h(n)e^{-j\omega nT} + h\left(\frac{M-1}{2}\right)e^{-j\omega\left(\frac{M-1}{2}\right)T} + \sum_{n=\frac{M+1}{2}}^{M-1} h(n)e^{-j\omega nT} \tag{7.6}$$

From Eq. (7.5),

$$h(n) = h(M-1-n)$$

Applying the condition given in Eq. (7.6), we get

$$H(e^{j\omega}) = \sum_{n=0}^{\frac{M-3}{2}} h(n)[e^{-j\omega nT} + e^{-j\omega(M-1-n)T}] + h\left(\frac{M-1}{2}\right)e^{-j\omega\left(\frac{M-1}{2}\right)T} \tag{7.7}$$

Factorising $e^{-j\omega(M-1)T/2}$ in the above equation,

$$H(e^{j\omega T}) = e^{-j\omega\left(\frac{M-1}{2}\right)T}\left\{\sum_{n=0}^{\frac{M-3}{2}} h(n)\left[e^{j\omega\left(\frac{M-1}{2}-n\right)T} + e^{-j\omega\left(\frac{M-1}{2}-n\right)T}\right] + h\left(\frac{M-1}{2}\right)\right\}$$

Letting $k = \frac{M-1}{2} - n$, we have

$$H(e^{j\omega}) = e^{-j\omega\left(\frac{M-1}{2}\right)T}\left\{\sum_{k=1}^{\frac{M-1}{2}} h\left(\frac{M-1}{2}-k\right)\left[e^{j\omega kT} + e^{-j\omega kT}\right] + h\left(\frac{M-1}{2}\right)\right\} \tag{7.8}$$

Equation (7.8) can be simplified as

$$H(e^{j\omega}) = e^{-j\omega\left(\frac{M-1}{2}\right)T}\left\{\sum_{k=0}^{\frac{M-1}{2}} a(k)\cos\omega kT\right\}$$

$$= e^{-j\omega\left(\frac{M-1}{2}\right)T} . M(\omega) \tag{7.9}$$

where $a(0) = h\left(\frac{M-1}{2}\right), a(k) = 2h\left(\frac{M-1}{2}-k\right)$ for $1 \le k \le \frac{M-1}{2}$

Equation (7.9) defines the frequency response of a causal FIR filter with linear phase shift. The cosine function is real and represents the frequency response. The phase response function $\Phi(\omega) = -\omega(M-1)/2$ represents the constant delay of $(M-1)/2$ units in sampling time.

DESIGN TECHNIQUES FOR FIR FILTERS 7.4

In this section we describe several methods for designing FIR filters. An important class of linear phase FIR filters is treated here.

7.4.1 Fourier Series Method

The sinusoidal steady-state transfer function of a digital filter is periodic in the sampling frequency and it can be expanded in a Fourier series.

Thus,

$$H(z)\Big|_{z=e^{j\omega}} = H(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h(n)\, e^{-j\omega nT} \tag{7.10}$$

where $h(n)$ represents the terms of the unit impulse response. In Eq. (7.10) the filter is assumed to be non-causal and it can be modified to yield a causal filter.

Rearranging the transfer function given below into real and imaginary components, we get

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

That is,

$$H(e^{j\omega}) = H_r(f) - jH_i(f)$$

where

$$H_r(f) = \sum_{n=-\infty}^{\infty} h(n)\cos 2\pi f nT \tag{7.11}$$

and

$$H_i(f) = \sum_{n=-\infty}^{\infty} h(n)\sin 2\pi f nT \tag{7.12}$$

From Eqs. (7.11) and (7.12) we infer that $H_r(f)$ is an even function and $H_i(f)$ is an odd function of frequency. If $h(nT)$ is an even sequence, the imaginary part of the transfer function, $H_i(f)$, will be zero and if $h(nT)$ is an odd sequence, the real part of the transfer function, $H_r(f)$, will be zero. Thus an even unit impulse response yields a real transfer function and an odd unit impulse response yields an imaginary transfer function. A real transfer function has 0 or $\pm\pi$ radians phase shift, while an imaginary transfer function has $\pm\pi/2$ radians phase shift. Therefore, by making the unit impulse response either even or odd, one can generate a transfer function that is either real or imaginary.

In the design of digital filters, two interesting situations are often sought after.

(i) For filtering applications, the main interest is in the amplitude response of the filter, where some portion of the input signal spectrum is to be attenuated and some portion is to be passed to the output with no attenuation. This should be accomplished without phase distortion. Thus the amplitude response is realised by using only a real transfer function. That is

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

and

$$H_i(f) \equiv 0$$

(ii) For filtering plus quadrature phase shift, the applications include integrators, differentiators and Hilbert transform devices. For all these applications, the desired transfer function is imaginary.

Thus, the required amplitude response is realised by using only $H_i(f)$. That is

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

and

$$H_r(f) \equiv 0$$

Design Equations

The term $H(e^{j\omega})$ is periodic in the sampling frequency and hence both $H_r(f)$ and $H_i(f)$ are also periodic in the sampling frequency. Both $H_r(f)$ and $H_i(f)$ can be expanded in a Fourier series. Since the real part of the transfer function, $H_r(f)$, is an even function of frequency, its Fourier series will be of the form

$$H_r(f) = a_0 + \sum_{n=1}^{\infty} a_n \cos(2\pi fnT) \tag{7.13}$$

The Fourier coefficients a_n are given by

$$a_n = \frac{2}{f_s}\int_{-f_s/2}^{f_s/2} H_r(f)\cos(2\pi fnT)\,\mathrm{d}f, \quad n \neq 0 \tag{7.14}$$

and the a_0 term is given by

$$a_0 = \frac{1}{f_s}\int_{-f_s/2}^{f_s/2} H_r(f)\,\mathrm{d}f$$

In a similar manner, the imaginary part of the transfer function, which is an odd function of frequency, can be expanded in the Fourier series

$$H_i(f) = \sum_{n=1}^{\infty} b_n \sin(2\pi fnT) \tag{7.15}$$

The Fourier coefficients b_n are given by

$$b_n = \frac{2}{f_s}\int_{-f_s/2}^{f_s/2} H_i(f)\sin(2\pi fnT)\,\mathrm{d}f \tag{7.16}$$

Since an odd function has zero average value, $b_0 \equiv 0$.

The Fourier coefficients a_n and b_n must be related to the unit pulse response of the filter.

Case (i): For simple filtering applications, consider $H_i(f) \equiv 0$. Equation (7.13) can be written as

$$\begin{aligned} H(e^{j\omega}) = H_r(f) &= a_0 + \sum_{n=1}^{\infty} a_n \cos(2\pi nfT) \\ &= a_0 + \sum_{n=1}^{\infty} \frac{a_n}{2}(z^n + z^{-n})\bigg|_{z=e^{j2\pi fT}} \end{aligned} \tag{7.17}$$

The transfer function can also be written as

$$\begin{aligned} H(e^{j\omega}) &= \sum_{n=-\infty}^{\infty} h(n)z^{-n}\,\Big|_{z=e^{j2\pi fT}} \\ &= h(0) + \sum_{n=1}^{\infty}\left[h(-n)z^n + h(n)z^{-n}\right]\Big|_{z=e^{j2\pi fT}} \end{aligned} \tag{7.18}$$

Comparing Eqs. (7.17) and (7.18), the terms of the unit pulse response $h(nT)$ can be related to the Fourier coefficients of the frequency response.

This yields

$$\left.\begin{aligned} h(0) &= a_0 \\ h(-n) &= \frac{1}{2}a_n \\ h(n) &= \frac{1}{2}a_n \end{aligned}\right\} n > 0 \qquad (7.19)$$

Case (ii): For filtering with quadrature phase shift, consider $H_r(f) \equiv 0$. From Eq. (7.15),

$$\begin{aligned} H(e^{j2\pi fT}) &= jH_i(f) = j\sum_{n=1}^{\infty} b_n \sin(2\pi fnT) \\ &= \sum_{n=1}^{\infty} \frac{b_n}{2}(z^n - z^{-n})\Big|_{z=e^{j2\pi fT}} \end{aligned} \qquad (7.20)$$

Comparing this equation with Eq. (7.18) yields

$$\left.\begin{aligned} h(-n) &= \frac{1}{2}b_n \\ h(n) &= -\frac{1}{2}b_n \end{aligned}\right\} n > 0 \qquad (7.21)$$

Equations (7.19) and (7.21) are the even and odd unit pulse responses respectively, and are used for the design of FIR filters.

Design Procedure

The procedure for designing an FIR digital filter using the Fourier series method is summarised as follows.

(i) Decide whether $H_r(f)$ or $H_i(f)$ is to be set equal to zero. For filtering applications we typically set $H_i(f) = 0$. For integrators, differentiators and Hilbert transformers, we set $H_r(f) = 0$.
(ii) Expand $H_r(f)$ or $H_i(f)$ in a Fourier series.
(iii) The unit pulse response is determined from the Fourier coefficients using Eqs (7.19) and (7.21).

The number of taps, i.e. the value of M may have to be increased in order to get a satisfactory sinusoidal steady-state response.

There are two problems involved in the implementation of FIR filters using this technique. The transfer function $H(e^{j\omega})$ represents a non-causal digital filter of infinite duration. A finite duration causal filter can be obtained by truncating the infinite duration impulse response and delaying the resulting finite duration impulse response. This modification does not effect 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 the points of discontinuity. This effect is known as the **Gibbs phenomenon**. These undesirable oscillations can be reduced by multiplying the desired impulse response coefficients by an appropriate window function.

Example 7.3 *Use the Fourier series method to design a low-pass digital filter to approximate the ideal specifications given by*

$$H(e^{j\omega}) = \begin{cases} 1, & \text{for } |f| \le f_p \\ 0, & f_p < |f| \le F/2 \end{cases}$$

where f_p is the passband frequency and F is the sampling frequency.

Solution For filtering applications,

$$H(e^{j\omega}) = H_r(f) = \begin{cases} 1, & \text{for} \mid f \mid \le f_p \\ 0, & f_p < \mid f \mid \le F/2 \end{cases}$$

Using Eq. (7.14),

$$a_n = \frac{2}{F}\int_{-F/2}^{F/2} H_r(f)\cos(2\pi f n T)\,\mathrm{d}f, \quad n \neq 0$$

$$= \frac{2}{F}\int_{-f_p}^{f_p} 1\cdot\cos(2\pi f n T)\,\mathrm{d}f = \frac{2}{F}\left[\frac{\sin 2\pi f n T}{2\pi n T}\right]_{-f_p}^{f_p}$$

Replacing T by $1/F$, and multiplying both numerator and denominator by f_p, we get

$$a_n = \left(\frac{2f_p}{F}\right)\left[\frac{\sin 2\pi n f_p / F}{\pi n(f_p/F)}\right]$$

$$a_0 = \frac{1}{F}\int_{-F/2}^{F/2} H_r(f)\,\mathrm{d}f = \frac{1}{F}\int_{-f_p}^{f_p} 1\cdot\mathrm{d}f = \frac{2f_p}{F}$$

Therefore, from Eq. (7.19),

$$h(0) = a_0 = \frac{2f_p}{F}, h(-n) = \left(\frac{f_p}{F}\right)\left[\frac{\sin 2\pi n f_p / F}{\pi n(f_p/F)}\right]$$

and $$h(n) = \left(\frac{f_p}{F}\right)\left[\frac{\sin 2\pi n f_p / F}{\pi n(f_p/F)}\right]$$

Example 7.4 *Design a Finite Impulse Response low-pass filter with a cut-off frequency of 1 kHz and sampling rate of 4 kHz with eleven samples using Fourier series.*

Solution Given f_c = 1 kHz, f_s = 4 kHz, and M = 11

$$H_d(\omega T) = \begin{cases} 1, -\omega_c \le \omega \le \omega_c \\ 0, -\dfrac{\omega_s}{2} \le \omega \le -\omega_c \\ 0, \omega_c \le \omega \le -\dfrac{\omega_s}{2} \end{cases}$$

$$\omega_c = 2\pi f_c = 6283.18 \text{ rad/ sec}$$

$$\omega_s = 2\pi f_s = 2\pi f_s = 25132.74 \text{ rad/ sec}$$

$$T = \frac{1}{f_s} = 0.25\times 10^{-3} = 0.25 \text{ msec}$$

To find h_d (n)

$$h_d(n) = \frac{1}{\omega_s}\int_{-\omega_c}^{\omega_c} e^{j\omega nT}\,d\omega = \frac{1}{\omega_s}\left[\frac{e^{j\omega nT}}{jnT}\right]_{-\omega_c}^{\omega_c}$$

$$= \frac{1}{\omega_s}\left[\frac{e^{j\omega_c nT}}{jnT} - \frac{e^{-j\omega_c nT}}{jnT}\right] = \frac{2}{\omega_s nT}\sin\omega_c nT$$

$$h_d(n) = \frac{2}{\omega_s nT}\sin\omega_c nT$$

When n ≠ 0

$$h_d(n) = \frac{2}{2\pi f_s nT}\sin\omega_c nT = \frac{\sin\omega_c nT}{\pi f_s nT} = \frac{\sin\omega_c nT}{\pi\frac{1}{T}nT} = \frac{\sin\omega_c nT}{n\pi}$$

When n = 0

$$h_d(n) = \underset{n\to 0}{Lt}\frac{2\sin\omega_c nT}{\omega_s nT} = \frac{2}{\omega_s}\underset{n\to 0}{Lt}\frac{\sin\omega_c nT}{nT} = \frac{2\omega_c}{\omega_s}$$

$$h(n) = h_d(n) \text{ for LPF} = \begin{cases}\dfrac{\sin\omega_c nT}{n\pi}; \text{ for } n = 1 \text{ to } M-1\\[2ex] \dfrac{2\omega_c}{\omega_s}; \quad \text{for } n = 0\end{cases}$$

$$h(0) = \frac{2\omega_c}{\omega_s} = 0.5$$

$$h(1) = \frac{\sin(6.283\times 1000\times 1\times 0.25\times 10^{-3})}{\pi} = 0.3183$$

Similarly,

$h(2) = 0$	$h(3) = -0.1061$
$h(4) = 0$	$h(5) = 0.0637$
$h(6) = 0$	$h(7) = -0.0455$
$h(8) = 0$	$h(9) = 0.0354$
$h(10) = 0$	

To find the transfer function

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

$$= z^{-5}\left[h(0) + \sum_{n=1}^{5} h(n)\left(z^n + z^{-n}\right)\right] = h(0)z^{-5} + \sum_{n=1}^{5} h(n)\left(z^{-5}z^n + z^{-5}z^{-n}\right)$$

$$= h(0)z^{-5} + h(1)z^{-4} + h(2)z^{-3} + h(3)z^{-2} + h(4)z^{-1} + h(5)z^0$$
$$+ h(1)z^{-6} + h(2)z^{-7} + h(3)z^{-8} + h(4)z^{-9} + h(5)z^{-10}$$

$$= h(5) + h(3)z^{-2} + h(1)z^{-4} + h(0)z^{-5} + h(1)z^{-6} + h(3)z^{-8} + h(5)z^{-10}$$

$$= 0.0637 + 0.3183(z^{-4} + z^{-6}) - 0.1061(z^{-2} + z^{-8}) + 0.25z^{-5} + 0.0637z^{-10}$$

$$H(z) = 0.0637(1 + z^{-10}) - 0.1061(z^{-2} + z^{-8}) + 0.3183(z^{-4} + z^{-6}) + 0.25z^{-5}$$

Example 7.5 *Design an ideal lowpass filter with a frequency response*

$$H_d\left(e^{j\omega}\right) = \begin{cases} 1, & -\dfrac{\pi}{2} \le \omega \le \dfrac{\pi}{2} \\ 0, & \dfrac{\pi}{2} \le |\omega| \le \pi \end{cases}$$

Determine h(n) and H(z) for M = 11 and plot the magnitude response.

Solution The filter coefficients are given by

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d\left(e^{j\omega}\right) e^{j\omega n} d\omega$$

$$= \frac{1}{2\pi}\int_{-\pi/2}^{\pi/2} e^{j\omega n} d\omega = \frac{1}{2\pi}\left[\frac{e^{j\omega n}}{jn}\right]_{-\pi/2}^{\pi/2} = \frac{\sin\left(\dfrac{\pi}{2}\right)n}{\pi n}, n \ne 0$$

Therefore, $h(n) = \begin{cases} \dfrac{\sin\left(\dfrac{\pi}{2}\right)n}{\pi n}, & |n| \le 5 \\ 0, & \text{otherwise} \end{cases}$

The filter coefficients are symmetrical about $n = 0$, i.e. $h(n) = h(-n)$.

For $n = 0$, $h(0) = \underset{n\to 0}{Lt} \dfrac{\sin\left(\dfrac{\pi}{2}\right)n}{\pi n} = \dfrac{1}{2} \underset{n\to 0}{Lt} \dfrac{\sin\left(\dfrac{\pi}{2}\right)n}{\left(\dfrac{\pi}{2}\right)n} = \dfrac{1}{2}$

Similarly, we can calculate

For $n = 1$, $h(1) = h(-1) = \dfrac{\sin(\pi/2)}{\pi} = \dfrac{1}{\pi} = 0.3183$

For $n = 2$, $h(2) = h(-2) = \dfrac{\sin 2(\pi/2)}{2\pi} = 0$

For $n = 3$, $h(3) = h(-3) = \dfrac{\sin 3(\pi/2)}{3\pi} = -0.106$

For $n = 4$, $h(4) = h(-4) = \dfrac{\sin 2\pi}{4\pi} = 0$

For $n = 5$, $h(5) = h(-5) = \dfrac{\sin 5(\pi/2)}{5\pi} = 0.06366$

The transfer function of the filter is given by

$$H(z) = \sum_{n=-\left(\frac{M-1}{2}\right)}^{\left(\frac{M-1}{2}\right)} h(n)\, z^{-n}$$

Therefore,

$$H(z) = 0.06366\, z^5 - 0.106\, z^3 + 0.3183\, z + 0.5 + 0.3183\, z^{-1} - 0.106\, z^{-3} + 0.06366\, z^{-5}$$

The transfer function of the realisable filter is

$$H'(z) = z^{-\left(\frac{M-1}{2}\right)} H(z)$$

$$= 0.06366 - 0.106\, z^{-2} + 0.3183\, z^{-4} + 0.5\, z^{-5} + 0.3183\, z^{-6} - 0.106\, z^{-8} + 0.06366\, z^{-10}$$

$$= 0.5\, z^{-5} + 0.3183\, (z^{-4} + z^{-6}) - 0.106\, (z^{-2} + z^{-8}) + 0.06366\, (1 + z^{-10})$$

The frequency response is given by

$$\left| H(e^{j\omega}) \right| = h\left[\frac{M-1}{2}\right] + \sum_{n=1}^{\left(\frac{M-1}{2}\right)} 2h\left[\frac{M-1}{2} - n\right]\cos\omega n$$

$$= 0.5 + 0.6366 \cos\omega - 0.212 \cos 3\omega + 0.127 \cos 5\omega$$

The magnitude response is

ω	0	$\frac{\pi}{6}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$	$\frac{4\pi}{6}$	$\frac{5\pi}{6}$	π
$\lvert H(e^{j\omega})\rvert$ in dB	0.44	–0.53	0.78	–6.02	–20.56	–24.63	–25.75

The magnitude response of the filter is shown in Fig. E7.5.

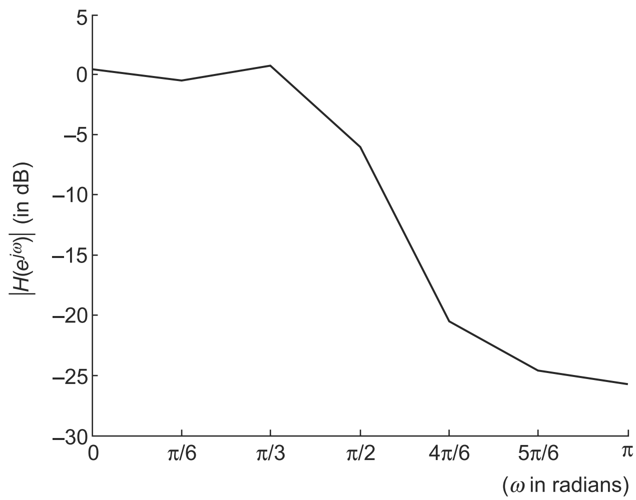


Fig. E7.5 *Magnitude Response*

Example 7.6 *Design an ideal highpass filter with a frequency response*

$$H_d\left(e^{j\omega}\right) = \begin{cases} 1, & \frac{\pi}{4} \le |\omega| \le \pi \\ 0, & |\omega| \le \frac{\pi}{4} \end{cases}$$

Determine h(n) and H(z) for M = 11 *and plot the magnitude response.*

Solution The desired response of the ideal highpass filter is

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d\left(e^{j\omega}\right) e^{j\omega n}\, d\omega$$

$$= \frac{1}{2\pi}\left[\int_{-\pi}^{-\pi/4} e^{j\omega n} d\omega + \int_{\pi/4}^{\pi} e^{j\omega n} d\omega\right] = \frac{1}{2\pi}\left[\left[\frac{e^{j\omega n}}{jn}\right]_{-\pi}^{-\pi/4} + \left[\frac{e^{j\omega n}}{jn}\right]_{\pi/4}^{\pi}\right] = \frac{1}{\pi n}\left(\sin \pi n - \sin\left(\frac{\pi}{4}\right)n\right)$$

Therefore, $h(n) = \begin{cases} h_d(n), & |n| \le 5 \\ 0, & \text{otherwise} \end{cases}$

The filter coefficients are symmetrical about $n = 0$ satisfying the condition $h(n) = h(-n)$.

For $\quad h(0) = \underset{n\to 0}{Lt} \frac{1}{\pi n}\left(\sin \pi n - \sin\left(\frac{\pi}{4}\right)n\right) = \frac{3}{4} = 0.75$

Similarly, we can calculate

For $n = 1$, $\quad h(1) = h(-1) = \frac{1}{\pi}\left(\sin \pi - \sin(\pi/4)\right) = -0.225$

For $n = 2$, $\quad h(2) = h(-2) = \frac{1}{2\pi}\left(\sin 2\pi - \sin(2\pi/4)\right) = -0.159$

For $n = 3$, $\quad h(3) = h(-3) = \frac{1}{3\pi}\left(\sin 3\pi - \sin(3\pi/4)\right) = -0.075$

For $n = 4$, $\quad h(4) = h(-4) = \frac{1}{4\pi}\left(\sin 4\pi - \sin(\pi)\right) = -0$

For $n = 5$, $\quad h(5) = h(-5) = \frac{1}{5\pi}\left(\sin 5\pi - \sin(5\pi/4)\right) = -0.045$

The transfer function of the filter is given by

$$H(z) = \sum_{n=-\frac{M-1}{2}}^{\frac{M-1}{2}} h(n)\, z^{-n}$$

$$= 0.045z^5 - 0.075z^3 - 0.159z^2 - 0.225z + 0.75 - 0.225z^{-1} - 0.159z^{-2} - 0.075z^{-3} + 0.045\, z^{-5}$$

The transfer function of the realizable filter is

$$H'(z) = z^{-\left(\frac{M-1}{2}\right)} H(z) = z^{-5}\, H(z)$$

$$= 0.045 - 0.075z^{-2} - 0.159z^{-3} - 0.225z^{-4} + 0.75z^{-5} - 0.225z^{-6} - 0.159z^{-7} - 0.075z^{-8} + 0.045\, z^{-10}$$

$$= 0.75z^{-5} - 0.225\left(z^{-4} + z^{-6}\right) - 0.159\left(z^{-3} + z^{-7}\right) - 0.075\left(z^{-2} + z^{-8}\right) + 0.045\left(1 + z^{-10}\right)$$

The magnitude response is

$$\left|H\left(e^{j\omega}\right)\right| = h\left[\frac{M-1}{2}\right] + \sum_{n-1}^{\left(\frac{M-1}{2}\right)} 2h\left[\frac{M-1}{2} - n\right]\cos \omega n$$

$$= 0.75 - 0.45 \cos \omega - 0.318 \cos 2\omega - 0.15 \cos 3\omega + 0.09 \cos 5\omega$$

ω	0	$\frac{\pi}{6}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$	$\frac{2\pi}{3}$	$\frac{5\pi}{6}$	π
$\lvert H(e^{j\omega})\rvert$ in dB	–22.16	–18.18	–1.12	0.57	–0.55	0.49	–0.52

The magnitude response of the filter is shown in Fig. E7.6.

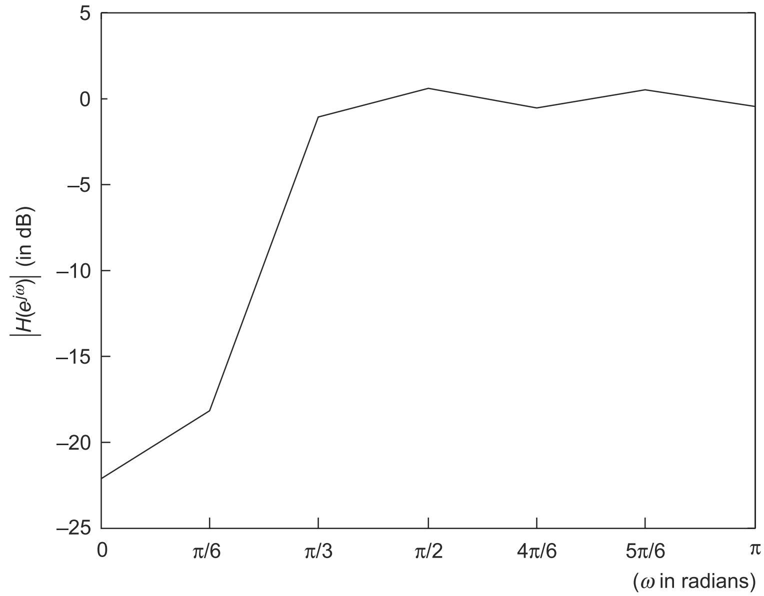


Fig. E7.6 *Magnitude Response*

Example 7.7 *Design an ideal bandpass filter with a frequency response*

$$H_d\left(e^{j\omega}\right)=\begin{cases}1, & \dfrac{\pi}{4}\le|\omega|\le\dfrac{3\pi}{4}\\ 0, & \text{otherwise}\end{cases}$$

Find h(n) and H(z) for M = 11 and plot the magnitude response.

Solution The ideal response is given by

$$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{\pi}H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega$$

$$h_d(n)=\frac{1}{2\pi}\left[\int_{-3\pi/4}^{-\pi/4}e^{j\omega n}d\omega+\int_{\pi/4}^{3\pi/4}e^{j\omega n}d\omega\right]=\frac{1}{\pi n}\left[\sin\left(\frac{3\pi}{4}\right)n-\sin\left(\frac{\pi}{4}\right)n\right]$$

Therefore, $h(n)=\begin{cases}h_d(n), & |n|\le 5\\ 0, & \text{otherwise}\end{cases}$

The filter coefficients are symmetrical about $n = 0$ satisfying the condition $h(n) = h(-n)$.

For $n = 0$, $$h(0)=\underset{n\to 0}{Lt}\frac{1}{\pi n}\left[\sin\frac{3\pi}{4}n-\sin\frac{\pi}{4}n\right]=0.5$$

Similarly, we can calculate

For $n = 1$, $$h(1)=h(-1)=\frac{1}{\pi}\left(\sin\frac{3\pi}{4}-\sin\frac{\pi}{4}\right)=0$$

For $n = 2$, $h(2) = h(-2) = \frac{1}{2\pi}\left(\sin\frac{3\pi}{2} - \sin\frac{\pi}{2}\right) = -0.3183$

For $n = 3$, $h(3) = h(-3) = \frac{1}{3\pi}\left(\sin\frac{9\pi}{4} - \sin\frac{3\pi}{4}\right) = 0$

For $n = 4$, $h(4) = h(-4) = \frac{1}{4\pi}(\sin 3\pi - \sin \pi) = 0$

For $n = 5$, $h(5) = h(-5) = \frac{1}{5\pi}\left(\sin\frac{15\pi}{4} - \sin\frac{5\pi}{4}\right) = 0$

The transfer function of the filter is

$$H(z) = \sum_{n=-\frac{M-1}{2}}^{\frac{M-1}{2}} h(n)\, z^{-n} \quad = 0.3183z^2 + 0.5 - 0.3183z^{-2}$$

The transfer function of the realisable filter is

$$H'(z) = z^{-\left(\frac{M-1}{2}\right)} H(z) \quad = -0.3183z^{-3} + 0.5z^{-5} - 0.3183z^{-7}$$

The magnitude response is

$$[H(e^{j\omega})] = 0.5 - 0.6366 \cos 2\omega$$

ω	0	$\frac{\pi}{6}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$	$\frac{2\pi}{3}$	$\frac{5\pi}{6}$	π
$\lvert H(e^{j\omega})\rvert$ in dB	–17.29	–14.81	–1.74	1.11	–1.74	–14.81	–17.29

The magnitude response of the filter is shown in Fig. E7.7.

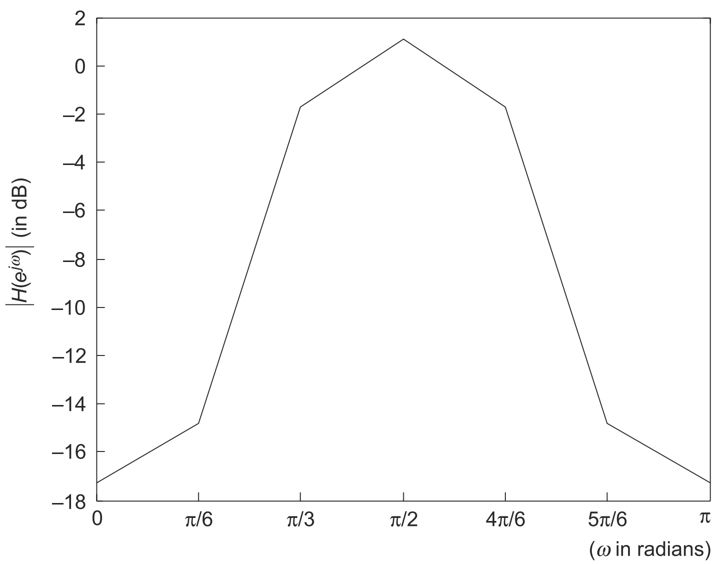


Fig. E7.7 *Magnitude Response*

Example 7.8 *Design an ideal band reject filter with a desired frequency response*

$$H_d\left(e^{j\omega}\right)=\begin{cases}1, & |\omega| \le \dfrac{\pi}{3} \text{ and } |\omega| \ge \dfrac{2\pi}{3}\\ 0, & otherwise\end{cases}$$

Find h(n) and H(z) for M = 11 and plot the magnitude response.

Solution The desired response is

$$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega$$

$$=\frac{1}{2\pi}\left[\int_{-\pi}^{-2\pi/3} e^{j\omega n}d\omega+\int_{-\pi/3}^{\pi/3} e^{j\omega n}d\omega+\int_{2\pi/3}^{\pi} e^{j\omega n}d\omega\right]$$

$$=\frac{1}{\pi n}\left[\sin\pi n+\sin\left(\frac{\pi}{3}\right)n-\sin\left(\frac{2\pi}{3}\right)n\right]$$

Therefore, $h(n)=\begin{cases}h_d(n), & |n|\le 5\\ 0, & \text{otherwise}\end{cases}$

The filter coefficients are symmetrical about $n = 0$ satisfying the condition $h(n) = h(-n)$.

For $n = 0$, $$h(0)=\underset{n\to 0}{Lt}\frac{1}{\pi n}\left[\frac{1}{\pi n}\left(\sin\pi n+\sin\left(\frac{\pi}{3}\right)n-\sin\left(\frac{2\pi}{3}\right)n\right)\right]=0.667$$

Similarly, we can calculate

For $n = 1$, $$h(1)=h(-1)=\frac{1}{\pi}\left(\sin\pi+\sin\frac{\pi}{3}-\sin\frac{2\pi}{3}\right)=0$$

For $n = 2$, $$h(2)=h(-2)=\frac{1}{2\pi}\left(\sin 2\pi+\sin\frac{2\pi}{3}-\sin\frac{4\pi}{3}\right)=0.2757$$

For $n = 3$, $$h(3)=h(-3)=\frac{1}{3\pi}\left(\sin 3\pi+\sin\pi-\sin 2\pi\right)=0$$

For $n = 4$, $$h(4)=h(-4)=\frac{1}{4\pi}\left(\sin 4\pi+\sin\frac{4\pi}{3}-\sin\frac{8\pi}{3}\right)=-0.1378$$

For $n = 5$, $$h(5)=h(-5)=\frac{1}{5\pi}\left(\sin 5\pi+\sin\frac{5\pi}{3}-\sin\frac{10\pi}{3}\right)=0$$

The transfer function of the filter is

$$H(z)=\sum_{n=-\left(\frac{M-1}{2}\right)}^{\left(\frac{M-1}{2}\right)} h(n)z^{-n}$$

$$= -0.1378z^4 + 0.2757z^2 + 0.667 + 0.2757z^{-2} - 0.1378z^{-4}$$

The transfer function of the realisable filter is

$$H'(z) = z^{-\left(\frac{M-1}{2}\right)} H(z)$$
$$= -0.1378z^{-1} + 0.2757z^{-3} + 0.667z^{-5} + 0.2757z^{-7} - 0.1378z^{-9}$$
$$= 0.667\,(z^{-5}) + 0.2757\,(z^{-3} + z^{-7}) - 0.1378\,(z^{-1} + z^{-9})$$

$H(e^{j\omega}) = 0.667 + 0.5514\cos 2\omega - 0.2756\cos 4\omega.$

ω	0	$\frac{\pi}{6}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$	$\frac{2\pi}{3}$	$\frac{5\pi}{3}$	π
$\lvert H(e^{j\omega})\rvert$ in dB	–0.51	0.67	–5.53	–15.92	–5.53	0.67	–0.51

The magnitude response of the filter is shown in Fig. E7.8.

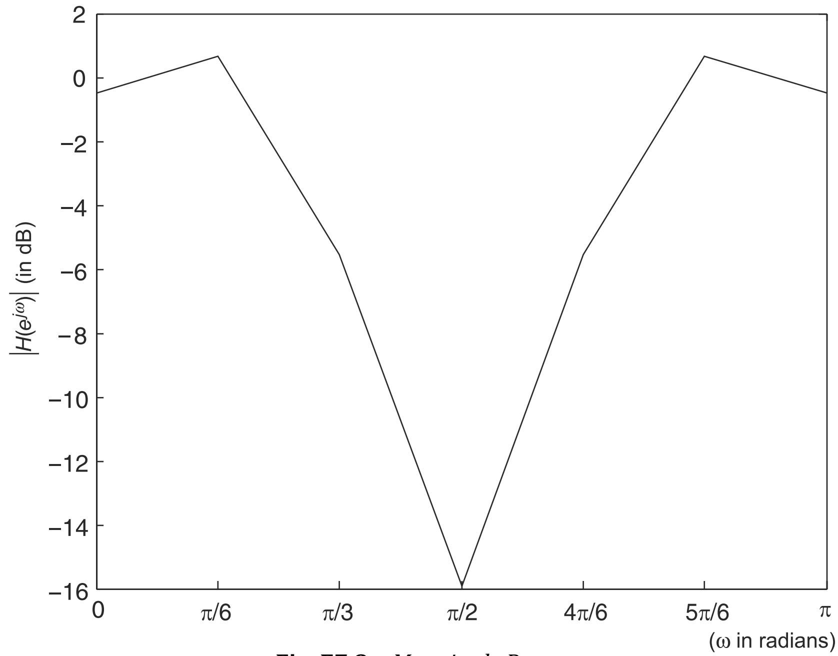


Fig. E7.8 *Magnitude Response*

7.4.2 Frequency Sampling Method

In this method, a set of samples is determined from the desired frequency response and are identified as discrete Fourier-transform (DFT) coefficients. The inverse discrete Fourier transform (IDFT) of this set of samples then gives the filter coefficients. The set of sample points used in this procedure can be determined by sampling a desired frequency response $H_d(e^{j\omega})$ at M points ω_k, $k = 0, 1, \ldots, M-1$, uniformly spaced around the unit circle. Two design techniques are available, viz., Type-I design and Type-II design. In the Type-I design, the set of frequency samples includes the sample at frequency $\omega = 0$. In some cases, it may be desirable to omit the sample at $\omega = 0$ and use some other set of samples. Such a design procedure is referred to as the Type-II design.

Type-I Design

The samples are taken at the frequency

$$\omega_k = \frac{2\pi k}{M}, \; k = 0, 1, \ldots, (M-1) \tag{7.22}$$

The samples of the desired frequency response at these frequencies are given by

$$\begin{aligned}\tilde{H}(k) &= H_d(e^{j\omega})\Big|_{\omega=\omega_k}, \quad k = 0,1,\ldots,(M-1)\\ &= H_d(e^{j2\pi k/M}), \quad k = 0,1,\ldots,(M-1)\end{aligned} \tag{7.23}$$

This set of points can be considered as DFT samples, then the filter coefficients $h(n)$ can be computed using the IDFT,

$$h(n) = \frac{1}{M}\sum_{k=0}^{M-1} \tilde{H}(k)e^{j2\pi nk/M}, \quad n = 0,1,\ldots,(M-1) \tag{7.24}$$

If these numbers are all real, then these can be considered as the impulse response coefficients of an FIR filter. This can happen when all the complex terms appear in complex conjugate pairs, and then all the terms can be matched by comparing the exponentials. The term $\tilde{H}(k)e^{j2\pi nk/M}$ should be matched with the term that has the exponential $e^{-j2\pi nk/M}$ as a factor. The matching terms are then $\tilde{H}(k)e^{j2\pi nk/M}$ and $\tilde{H}(M-k)e^{j2\pi n(M-k)/M}$ since $2\pi n(M-k)/M = 2\pi n - (2\pi nk/M)$. These terms are complex conjugates if $\tilde{H}(0)$ is real and

(i) *For M odd*

$$\tilde{H}(M-k) = \tilde{H}^*(k), \quad k = 1,2,\ldots,(M-1)/2 \tag{7.25a}$$

The sampled frequency response is given by

$$\tilde{H}(k) = \begin{cases} |H(k)|e^{-j(M-1)\pi k/M}, & k = 0,1,\ldots,\left(\dfrac{M-1}{2}\right)\\ |H(k)|e^{j\left[(M-1)\pi - \frac{(M-1)\pi k}{M}\right]}, & k = \left(\dfrac{M+1}{2}\right),\ldots,(M-1)\end{cases} \tag{7.25b}$$

(ii) *For M even*

$$\begin{aligned}&\tilde{H}(M-k) = \tilde{H}^*(k), \quad k = 1,2,\ldots,(M/2)-1\\ &\tilde{H}(M/2) = 0\end{aligned} \tag{7.26a}$$

The sampled frequency response is given by

$$\tilde{H}(k) = \begin{cases} |H(k)|e^{-j(M-1)\pi k/M}, & k = 0,1,\ldots,\left(\dfrac{M}{2}-1\right)\\ |H(k)|e^{j\left[(M-1)\pi - \frac{(M-1)\pi k}{M}\right]}, & k = \left(\dfrac{M}{2}+1\right),\ldots,(M-1)\\ 0, & k = \dfrac{M}{2}\end{cases} \tag{7.26b}$$

The desired frequency response $H_d(e^{j\omega})$ is chosen such that it satisfies Eqs. (7.25) and (7.26) for M odd or even, respectively. The filter coefficients can then be written as

$$h(n) = \frac{1}{M}\left[\tilde{H}(0) + 2\sum_{k=1}^{(M-1)/2} \mathrm{Re}\left[\tilde{H}(k)e^{j2\pi nk/M}\right]\right], M \text{ odd} \tag{7.27}$$

and

$$h(n)=\frac{1}{M}\left[\tilde{H}(0)+2\sum_{k=1}^{(M/2)-1}\text{Re}\left[\tilde{H}(k)e^{j2\pi nk/M}\right]\right], M \text{ even} \tag{7.28}$$

The system function of the filter can be determined from the filter coefficients $h(n)$,

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

Replacing z by $e^{j\omega}$, we get, $H(e^{j2\pi k/M})=\sum_{n=0}^{M-1}h(n)e^{-j2\pi kn/M}$

As per the definition of DFT, the above equation is simply the DFT of $\tilde{H}(k)$. Using Eq. (7.23),

$$H(e^{j2\pi k/M})=\tilde{H}(k)=H_d(e^{-j2\pi k/M})$$

At $\omega = 2\pi k/M$, the values $H(e^{j\omega})$ and $H_d(e^{j\omega})$ are same and for other values of ω, $H(e^{j\omega})$ can be thought of as an interpolation of the sampled desired frequency response. $H_d(e^{j\omega})$ and/or M can be varied to get the suitable frequency response $H(e^{j\omega})$.

Type-II Design

The frequency sample in Eq. (7.23) included the sample at frequency $\omega = 0$. In Type-II design, the sample at frequency $\omega = 0$ is omitted and the samples are taken at frequencies

$$\omega_k=\frac{2\pi(2k+1)}{2M},\quad k=0,1,\ldots,(M-1)$$
$$=\frac{2\pi(k+1/2)}{M},\quad k=0,1,\ldots,(M-1) \tag{7.29}$$

and the DFT coefficients are defined by

$$\tilde{H}(k)=H_d(e^{j\pi(2k+1)M}),\quad k=0,1,\ldots,(M-1) \tag{7.30}$$

(i) *For M odd*

$$\tilde{H}(M-k-1)=\tilde{H}^*(k),\quad k=0,1,\ldots,\left(\frac{M-1}{2}\right)$$
$$\tilde{H}\left(\frac{M-1}{2}\right)=0 \tag{7.31}$$

(ii) *For M even*

$$\tilde{H}(M-k-1)=\tilde{H}^*(k),\quad k=0,1,\ldots,\left(\frac{M-1}{2}\right) \tag{7.32}$$

The filter coefficients are then written as

$$h(n)=\frac{2}{M}\sum_{k=0}^{(M-3)/2}\text{Re}\left[\tilde{H}(k)e^{j\pi n(2k+1)/M}\right], M \text{ odd} \tag{7.33}$$

and

$$h(n)=\frac{2}{M}\sum_{k=0}^{\frac{M}{2}-1}\text{Re}\left[\tilde{H}(k)e^{j\pi n(2k+1)/M}\right], M \text{ even} \tag{7.34}$$

Example 7.9 *A low-pass filter has the desired response as given below*

$$H_d(e^{j\omega})=\begin{cases}e^{-j3\omega} & 0\le\omega<\frac{\pi}{2}\\ 0 & \frac{\pi}{2}\le\omega\le\pi\end{cases}$$

Determine the filter coefficients h(n) for $M = 7$ *using Type-I frequency sampling technique.*

Solution The samples of the given frequency response are taken uniformly at $\omega_k = 2\pi k/M$. For $0 \le \omega \le \frac{\pi}{2}$, the values of k = 0, 1. For $\frac{\pi}{2} \le \omega \le \frac{3\pi}{2}, k = 2,3,4,5$ and for $\frac{3\pi}{2} \le \omega \le 2\pi, k = 6$. Using Eq. (7.25b), the sampled frequency response is obtained as

$$\tilde{H}(k) = \begin{cases} e^{-j6\pi k/7,} & k = 0,1 \\ 0 & k = 2,3,4,5 \\ e^{-j6\pi k/7,} & k = 6 \end{cases}$$

The filter coefficients $h_d(n)$ are given by the inverse discrete Fourier transform,

$$h_d(n) = \frac{1}{M}\sum_{k=0}^{M-1} \tilde{H}(k)e^{j2\pi kn/M} = \frac{1}{7}\sum_{k=0}^{6} \tilde{H}(k)e^{j2\pi kn/7}$$

$$= \frac{1}{7}\left[\sum_{k=0}^{1} e^{-j6\pi k/7}e^{j2\pi kn/7} + e^{-j6\pi k/7}e^{j2\pi kn/7}\,|_{k=6}\right]$$

$$= \frac{1}{7}\left[\sum_{k=0}^{1} e^{j2\pi k(n-3)/7} + e^{j12\pi(n-3)/7}\right] = \frac{1}{7}\left[1 + e^{j2\pi(n-3)/7} + e^{j12\pi(n-3)/7}\right]$$

Since $\tilde{H}(k) = \tilde{H}^*(M-k)$, we have $e^{j12\pi(n-3)/7} = e^{-j2\pi(n-3)/7}$

Therefore,

$$h_d(n) = \frac{1}{7}\left[1 + e^{j2\pi(n-3)/7} + e^{-j2\pi(n-3)/7}\right]$$

$$h_d(n) = 0.1429 + 0.2857\cos[0.898(n-3)]$$

Example 7.10 *Determine the filter coefficients h(n) obtained by sampling*

$$H_d\left(e^{j\omega}\right) = \begin{cases} e^{-j(M-1)\omega/2} & ; \; 0 \le |\omega| \le \frac{\pi}{2} \\ 0 & ; \; \frac{\pi}{2} \le |\omega| \le \pi \end{cases}$$

for M = 7

Solution The samples of the given frequency response are taken uniformly at $\omega = \frac{2\pi\, k}{M}$.

For $0 \le \omega \le \frac{\pi}{2}$, $\quad k = 0,1$

For $\frac{\pi}{2} \le \omega \le \frac{3\pi}{2}$, $\quad k = 2, 3, 4, 5$

For $\frac{3\pi}{2} \le \omega \le 2\pi$, $\quad k = 6$

Using Eq. (7.25b), the sampled frequency response is obtained as

$$\tilde{H}\left(k\right) = \begin{cases} e^{-j6\pi k/7}, & k = 0,1 \\ 0, & k = 2,3,4,5 \\ e^{-j6\pi(k-7)/7}, & k = 6 \end{cases}$$

The filter coefficients $h_d(n)$ are given by the inverse discrete Fourier transform

$$h_d(n) = \frac{1}{M}\sum_{K=0}^{M-1} \tilde{H}(k)\, e^{j2\pi kn/M}$$

$$= \frac{1}{7}\sum_{k=0}^{6}\tilde{H}(k)\,e^{j2\pi kn/7} \quad = \frac{1}{7}\left[1 + e^{-j6\pi/7}e^{-j2\pi n/7} + e^{j6\pi/7}e^{j2\pi n/7}\right]$$

$$h_d(n) = \frac{1}{7}\left[1 + 2\cos\frac{2\pi}{7}(n-3)\right]$$

$h(n) = h_d(n)$; $n = 0, 1, \ldots (M-1)$

$$h(0) = h(6) = \frac{1}{7}\left(1 + 2\cos\frac{6\pi}{7}\right) = -0.1146$$

$$h(1) = h(5) = \frac{1}{7}\left(1 + 2\cos\frac{4\pi}{7}\right) = 0.0793$$

$$h(2) = h(4) = \frac{1}{7}\left(1 + 2\cos\frac{2\pi}{7}\right) = 0.321$$

$$h(3) = \frac{1}{7}(1+2) = 0.4286$$

$h(n) = \{-0.1146, 0.0793, 0.321, 0.4286, 0.321, 0.0793, -0.1146\}$

Example 7.11 *Determine the filter coefficients of a linear phase FIR filter of length M = 15 which has a symmetric unit sample response and a frequency response that satisfies the conditions*

$$H\left[\frac{2\pi k}{15}\right] = \begin{cases} 1, & k = 0, 1, 2, 3 \\ 0, & k = 4, 5, 6, 7 \end{cases}$$

Solution Using Eq. (7.25b), the sampled frequency response is obtained as

$$\tilde{H}(k) = \begin{cases} e^{-j14\pi k/15}, & k = 0, 1, 2, 3 \\ 0, & k = 4, 5, 6, 7, 8, 9, 10, 11 \\ e^{-j14\pi(k-15)/15}, & k = 12, 13, 14 \end{cases}$$

The filter coefficients are given by

$$h_d(n) = \frac{1}{M}\sum_{K=0}^{M-1}\tilde{H}(k)e^{j2\pi kn/M} \quad = \frac{1}{15}\sum_{K=0}^{14}\tilde{H}(k)e^{j2\pi kn/15}$$

$$= \frac{1}{15}\left(1 + e^{-j14\pi/15}e^{j2\pi n/15} + e^{-j28\pi/15}e^{j4\pi n/15} + e^{-j42\pi/15}e^{j6\pi n/15} + e^{j42\pi/15}e^{j24\pi n/15}\right.$$
$$\left. + e^{j28\pi/15}e^{j26\pi n/15} + e^{j14\pi/15}e^{j28\pi n/15}\right)$$

$$= \frac{1}{15}\left[1 + 2\cos\frac{2\pi}{15}(7-n) + 2\cos\frac{4\pi}{15}(7-n) + 2\cos\frac{6\pi}{15}(7-n)\right]$$

Substituting $n = 0, 1, \ldots 14$, we get

$h(n) = [-0.05, 0.041, -0.1078, 0.0666, -0.0365, 0.034, 0.3188, 0.466, 0.3188, 0.034, -0.0365, 0.0666, -0.1078, 0.041, -0.05]$

$$H(z) = -0.05 + 0.041z^{-1} - 0.1078z^{-2} + 0.0666z^{-3} - 0.0365z^{-4} + 0.034z^{-5} + 0.3188\,z^{-6} + 0.466z^{-7}$$
$$+ 0.3188z^{-8} + 0.034z^{-9} - 0.0365z^{-10} + 0.0666z^{-11} - 0.1078z^{-12} + 0.041z^{-13} - 0.05z^{-14}$$
$$= -0.05\,(1 + z^{-14}) + 0.014\,(z^{-1} + z^{-13}) - 0.1078\,(z^{-2} + z^{-12}) + 0.0666\,(z^{-3} + z^{-11})$$
$$- 0.0365\,(z^{-4} + z^{-10}) + 0.034\,(z^{-5} + z^{-9}) - 0.3188\,(z^{-6} + z^{-8}) + 0.466\,z^{-7}$$

Example 7.12 *Design a band pass filter with the following specifications:*

Sampling frequency, F = 8 kHz

Cut off frequencies, f_{cl} = 1 kHz and f_{c2} = 3 kHz

Use frequency sampling method and determine the filter coefficients for M = 7.

Solution $\omega_{c1} = 2\pi f_{c1} T = \frac{2\pi f_{c1}}{F} = \frac{2\pi \times 1000}{8000} = \frac{\pi}{4}$

$$\omega_{c2} = 2\pi f_{c2} T = \frac{2\pi f_{c2}}{T} = \frac{2\pi \times 3000}{8000} = \frac{3\pi}{4}$$

The samples for the given frequency response are taken uniformly at $\omega_k = \frac{2\pi k}{M}$.

For $0 \le \omega \le \frac{\pi}{4}$, $k = 0$

For $\frac{\pi}{4} \le \omega \le \frac{3\pi}{4}$, $k = 1, 2$

For $\frac{3\pi}{4} \le \omega \le \frac{5\pi}{4}$, $k = 3, 4$

For $\frac{5\pi}{4} \le \omega \le \frac{7\pi}{4}$, $k = 5, 6$

Using Eq. (7.25b), the sampled frequency response is obtained as

$$\tilde{H}(k) = \begin{cases} 0, & k = 0,3,4 \\ e^{-j6\pi(k-7)/7}, & k = 1,2,5,6 \end{cases}$$

The filter coefficients $h_d(n)$ are given the inverse discrete Fourier transform

$$h_d(n) = \frac{1}{M}\sum_{K=0}^{M-1} \tilde{H}(k)\, e^{j2\pi kn/M} = \frac{1}{7}\sum_{K=0}^{6} \tilde{H}(k) e^{j2\pi kn/7}$$

$$= \frac{1}{7}\left[e^{-j6\pi/7} e^{j2\pi n/7} + e^{-j12\pi/7} e^{j4\pi n/7} + e^{-j30\pi/7} e^{j10\pi n/7} + e^{-j36\pi/7} e^{j12\pi n/7}\right]$$

$$= \frac{1}{7}\left[2\cos\frac{2\pi}{7}(3-n) + 2\cos\frac{4\pi}{7}(3-n)\right]$$

Substituting n = 0, 1, ..., 6, we get

$h(n)$ = {–0.0793, –0.321, 0.1146, 0.57, 0.1146, –0.321, –0.0793}

Example 7.13 *(a) Use frequency sampling method to design an FIR low-pass filter $\omega_c = \frac{\pi}{4}$ for M =15, (b) Repeat part (a) by selecting an additional sample |H(k)| = 0.5 in the transition band.*

Solution (a) Using Eq. (7.25b), the sampled frequency response is obtained as

$$\tilde{H}(k) = \begin{cases} e^{-j14\pi k/15}, & k = 0,1 \\ 0, & k = 2,3,....13 \\ e^{-j14\pi(k-15)/15} & k = 14 \end{cases}$$

The filter coefficients $h_d(n)$ are given by

$$h_d(n) = \frac{1}{M}\sum_{K=0}^{M-1} \tilde{H}(k)\, e^{j2\pi kn/M}$$

$$= \frac{1}{15}\sum_{K=0}^{14}\tilde{H}(k)\,e^{j2\pi kn/15} = \frac{1}{15}\left[\,1+e^{-j14\pi/15}e^{j2\pi n/15}+e^{j14\pi/15}e^{j28\pi n/15}\right]$$

$$= \frac{1}{15}\left[1+2\cos\frac{2\pi}{15}\,(7-n)\,\right]$$

Substituting $n = 0, 1, \ldots, 14$, we get

$h(n) =$ [–0.0637, –0.0412, 0, 0.0527, 0.1078, 0.156, 0.188, 0.2, 0.188, 0.156, 0.1078, 0.0527, 0, –0.0412, –0.0637]

(b) By introducing an additional sample in the transition band, the magnitude of frequency response is given by

$$\left|\tilde{H}(k)\right| = \begin{cases} 1, & k = 0,1,14 \\ 0.5, & k = 2,13 \\ 0, & k = 3,4,\ldots,12 \end{cases}$$

Therefore, the sampled frequency response is given by

$$\tilde{H}(k) = \begin{cases} e^{-j14\pi k/15}, & k = 0,1 \\ 0.5\,e^{-j14\pi k/15}, & k = 2 \\ 0, & k = 3,4,\ldots,12 \\ 0.5\,e^{-j14\pi(k-15)/15}, & k = 13 \\ e^{-j14\pi(k-15)/15}, & k = 14 \end{cases}$$

Thus, the filter coefficients $h_d(n)$ are given by

$$h_d(n) = \frac{1}{M}\sum_{K=0}^{M-1}\tilde{H}(k)\,e^{j2\pi kn/M}$$

$$= \frac{1}{15}\sum_{K=0}^{14}\tilde{H}(k)e^{j2\pi kn/15}$$

$$= \frac{1}{15}\left[1+e^{-j14\pi/15}e^{j2\pi n/15}+0.5\,e^{-j28\pi/15}\;e^{j4\pi n/15}+0.5e^{j28\pi/15}e^{j26\pi n/15}+e^{j14\pi/15}\;e^{j28\pi n/15}\;\right]$$

$$= \frac{1}{15}\left[1+2\cos\frac{2\pi}{15}(7-n)+2\cos\frac{4\pi}{15}(7-n)\right]$$

Substituting $n = 0, 1, \ldots, 14$, we get

$h(n) =$ [–0.0029, –0.0206, –0.0333, –0.0125, 0.0540.1489, 0.233, 0.267, 0.233, 0.1489, 0.054, –0.0125, –0.0333, –0.0206, –0.0029]

7.4.3 Window Techniques

The desired frequency response of any digital filter is periodic in frequency and can be expanded in a Fourier series, i.e.

$$H_d(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h_d(n)e^{-j\omega n} \tag{7.35}$$

where
$$h_d(n) = \frac{1}{2\pi} \int_0^{2\pi} H_d(e^{j\omega}) e^{j\omega n}\, \mathrm{d}\omega \tag{7.36}$$

The Fourier coefficients of the series $h_d(n)$ are identical to the impulse response of a digital filter. There are two difficulties with the implementation of Eq. (7.35) for designing a digital filter. First, the impulse response is of infinite duration and second, the filter is non-causal and unrealisable. No finite amount of delay can make the impulse response realisable. Hence the filter resulting from a Fourier series representation of $H(e^{j\omega})$ is an unrealisable IIR filter.

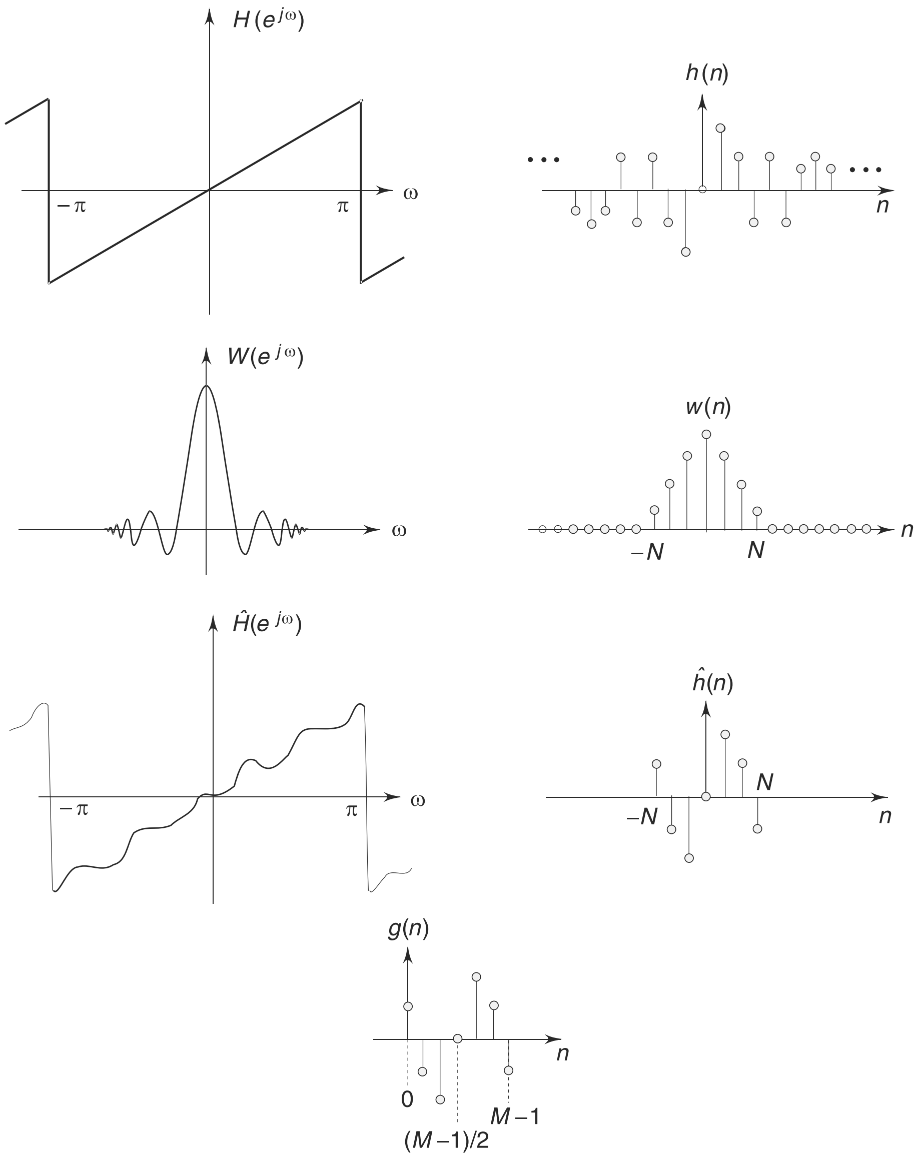


Fig. 7.1 *Illustration of the Window Technique*

The infinite duration impulse response can be converted to a finite duration impulse response by truncating the infinite series at $n = \pm M$. But, this results in undesirable oscillations in the passband and stopband of the digital filter. This is due to the slow convergence of the Fourier series near the points of discontinuity. These undesirable oscillations can be reduced by using a set of time-limited weighting functions, $w(n)$, referred to as window functions, to modify the Fourier coefficients. The windowing technique is illustrated in Fig. 7.1.

The desired frequency response $H(e^{j\omega})$ and its Fourier coefficients $\{h(n)\}$ are shown at the top of this figure. The finite duration weighting function $w(n)$ and its Fourier transform $W(e^{j\omega})$ are shown in the second row. The Fourier transform of the weighting function consists of a main lobe, which contains most of the energy of the window function and side lobes which decay rapidly. The sequence $\hat{h}(n) = h(n).w(n)$ is obtained to get an FIR approximation of $H(e^{j\omega})$. The sequence $\hat{h}(n)$ is exactly zero outside the interval $-M \le n \le M$. The sequence $\hat{h}(n)$ and its Fourier transform $\hat{H}(e^{j\omega})$ are shown in the third row. $\hat{H}(e^{j\omega})$ is nothing but the circular convolution of $H(e^{j\omega})$ and $W(e^{j\omega})$. The realisable causal sequence $g(n)$, which is obtained by shifting $\hat{h}(n)$, is shown in the last row and this can be used as the desired filter impulse response.

A major effect of windowing is that the discontinuities in $H(e^{j\omega})$ are converted into transition bands between values on either side of the discontinuity. The width of these transition bands depends on the width of the main lobe of $W(e^{j\omega})$. A secondary effect of windowing is that the ripples from the side lobes of $W(e^{j\omega})$ produces approximation errors for all ω. Based on the above discussion, the desirable characteristics can be listed as follows

(i) The Fourier transform of the window function $W(e^{j\omega})$ should have a small width of main lobe containing as much of the total energy as possible.

(ii) The Fourier transform of the window function $W(e^{j\omega})$ should have side lobes that decrease in energy rapidly as ω tends to π. Some of the most frequently used window functions are described in the following sections.

Rectangular Window Function

The weighting function for the rectangular window is given by

$$w_R(n) = \begin{cases} 1, & \text{for } |n| \le \dfrac{M-1}{2} \\ 0, & \text{otherwise} \end{cases} \tag{7.37}$$

The spectrum of $w_R(n)$ can be obtained by taking Fourier transform of Eq. (7.37) as

$$W_R(e^{j\omega T}) = \sum_{n=\frac{-(M-1)}{2}}^{\frac{(M-1)}{2}} e^{-j\omega nT}$$

Substituting $n = m - (M-1)/2$ and replacing m by n, we get

$$W_R(e^{j\omega T}) = e^{j\omega\frac{(M-1)}{2}T} \sum_{n=0}^{(M-1)} e^{-j\omega nT} = e^{j\omega\frac{(M-1)}{2}T}\left[\frac{1-e^{-j\omega MT}}{1-e^{-j\omega T}}\right] = \frac{e^{j\omega\frac{M}{2}T} - e^{-j\omega\frac{M}{2}T}}{e^{j\omega\frac{T}{2}} - e^{-j\omega\frac{T}{2}}}$$

$$= \frac{\sin\left(\dfrac{\omega MT}{2}\right)}{\sin\left(\dfrac{\omega T}{2}\right)} \tag{7.38}$$

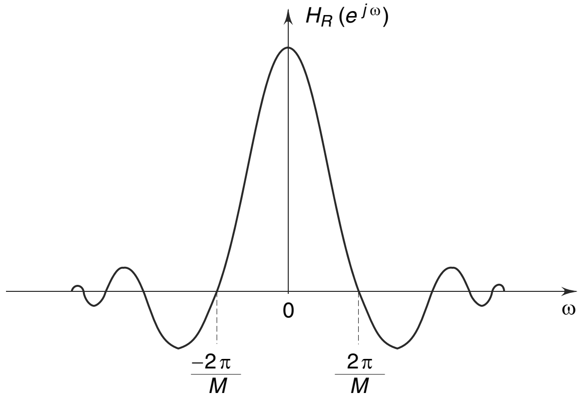


Fig. 7.2 *Frequency Response of a Rectangular Window*

A sketch of Eq. (7.38) is shown in Fig. 7.2. The transition width of the main lobe is approximately $4\pi/M$. The first sidelobe will be 13 dB down the peak of the main lobe and the rolloff will be at 20 dB per decade. For a causal rectangular window, the frequency response will be

$$W_R(e^{j\omega T}) = \sum_{n=0}^{(M-1)} e^{-j\omega nT}$$

$$= e^{-j\omega\frac{(M-1)}{2}T}\left[\frac{\sin\left(\frac{\omega MT}{2}\right)}{\sin\left(\frac{\omega T}{2}\right)}\right] \tag{7.39}$$

From Eq. (7.38) and (7.39), it is noted that the linear phase response of the causal filter is given by $\theta(\omega) = \omega(M-1)T/2$, and the non-causal impulse response has a zero phase shift.

Hamming Window Function

The causal Hamming window function is expressed by

$$w_H(n) = \begin{cases} 0.54 - 0.46\cos\dfrac{2\pi n}{M-1}, & 0 \le n < M-1 \\ 0, & \text{otherwise} \end{cases}$$

The non-causal Hamming window function is given by

$$w_H(n) = \begin{cases} 0.54 + 0.46\cos\dfrac{2\pi n}{M-1}, & \text{for } |n| \le \dfrac{M-1}{2} \\ 0, & \text{otherwise} \end{cases} \tag{7.40}$$

It can be noted that the non-causal Hamming window function is related to the rectangular window function as shown below

$$w_H(n) = w_R(n)\left[0.54 + 0.46\cos\frac{2\pi n}{M-1}\right] \tag{7.41}$$

The spectrum of Hamming window can then be obtained as

$$W_H(e^{j\omega T}) = 0.54\frac{\sin\left(\frac{\omega MT}{2}\right)}{\sin\left(\frac{\omega T}{2}\right)} + 0.46\frac{\sin\left(\frac{\omega MT}{2} - \frac{M\pi}{(M-1)}\right)}{\sin\left(\frac{\omega T}{2} - \frac{\pi}{(M-1)}\right)}$$
$$+0.46\frac{\sin\left(\frac{\omega MT}{2} + \frac{M\pi}{(M-1)}\right)}{\sin\left(\frac{\omega T}{2} + \frac{\pi}{(M-1)}\right)} \tag{7.42}$$

The width of the main lobe is approximately $8\pi/M$ and the peak of the first side lobe is at −43 dB. The side lobe rolloff is 20 dB/decade. For a causal Hamming window, the second and third terms are negative as shown below

$$W_H(e^{j\omega T}) = 0.54\frac{\sin\left(\frac{MT}{2}\right)}{\sin\left(\frac{\omega T}{2}\right)} - 0.46\frac{\sin\left(\frac{\omega MT}{2} - \frac{M\pi}{(M-1)}\right)}{\sin\left(\frac{\omega T}{2} - \frac{\pi}{(M-1)}\right)}$$
$$-0.46\frac{\sin\left(\frac{\omega MT}{2} + \frac{M\pi}{(M-1)}\right)}{\sin\left(\frac{\omega T}{2} + \frac{\pi}{(M-1)}\right)} \tag{7.43}$$

Hanning Window Function

The window function of a causal Hanning window is given by

$$w_{Hann}(n) = \begin{cases} 0.5 - 0.5\cos\frac{2\pi n}{M-1}, & 0 \le n \le M-1 \\ 0, & \text{otherwise} \end{cases} \tag{7.44}$$

The window function of a non-causal Hanning window is expressed by

$$w_{Hann}(n) = \begin{cases} 0.5 + 0.5\cos\frac{2\pi n}{M-1}, & 0 < |n| < \frac{M-1}{2} \\ 0, & \text{otherwise} \end{cases}$$

The width of the main lobe is approximately $8\pi/M$ and the peak of the first side lobe is at −32 dB.

Blackman Window Function

The window function of a causal Blackman window is expressed by

$$w_B(n) = \begin{cases} 0.42 - 0.5\cos\frac{2\pi n}{M-1} + 0.08\cos\frac{4\pi n}{M-1}, & 0 \le n \le M-1 \\ 0, & \text{otherwise} \end{cases}$$

The window function of a non-causal Blackman window is given by

$$w_B(n) = \begin{cases} 0.42 + 0.5\cos\dfrac{2\pi n}{M-1} + 0.08\cos\dfrac{4\pi n}{M-1}, & \text{for } |n| < \dfrac{M-1}{2} \\ 0, & \text{otherwise} \end{cases} \tag{7.45}$$

The width of the main lobe is approximately $12\pi/M$ and the peak of the first side-lobe is at –58 dB.

Bartlett Window Function

The window function of a non-causal Bartlett window is expressed by

$$w_{Bart}(n) = \begin{cases} 1+n, & -\dfrac{M-1}{2} < n < 1 \\ 1-n, & 1 < n < \dfrac{M-1}{2} \end{cases}$$

Table 7.1 gives the important frequency-domain characteristics of some window functions.

Table 7.1 *Frequency-domain characteristics of some window functions*

Type of Window	*Approximate Transition Width of Main Lobe*	*Minimum Stopband Attenuation (dB)*	*Peak of first Sidelobe (dB)*
Rectangular	$4\pi/M$	-21	-13
Bartlett	$8\pi/M$	-25	-27
Hanning	$8\pi/M$	-44	-32
Hamming	$8\pi/M$	-53	-43
Blackman	$12\pi/M$	-74	-58

Example 7.14 *Determine the unit sample response of the ideal low-pass filter. Why is it not realisable?*

Solution The desired frequency response of the ideal low-pass digital filter is given by

$$H_d(e^{j\omega}) = \begin{cases} e^{-j\omega\alpha}; & -\omega_c \le \omega \le \omega_c \\ 0 & ; \omega_c \le |\omega| < \pi \end{cases}$$

Therefore, the unit sample response of the linear phase low-pass filter is

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}\,d\omega = \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} e^{-j\omega\alpha}\,e^{j\omega n}\,d\omega$$

$$= \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} e^{j\omega(n-\alpha)}\,d\omega = \frac{1}{2\pi}\left[\frac{e^{j\omega(n-\alpha)}}{j(n-\alpha)}\right]_{-\omega_c}^{\omega_c}$$

$$= \frac{1}{2\pi}\left[\frac{e^{j\omega_c(n-\alpha)}}{j(n-\alpha)} - \frac{e^{j(-\omega_c)(n-\alpha)}}{j(n-\alpha)}\right] = \frac{1}{\pi(n-\alpha)}\left[\frac{e^{j\omega_c(n-\alpha)} - e^{j(-\omega_c)(n-\alpha)}}{2j}\right]$$

$$= \frac{1}{\pi(n-\alpha)}\sin[\omega_c(n-\alpha)]$$

$$h_d(n) = \frac{\sin[\omega_c(n-\alpha)]}{\pi(n-\alpha)}$$

Since this impulse response is non-causal and is of doubly infinite length, this ideal low-pass filter is not realisable.

Note: Also, as the impulse response is not absolutely summable, the corresponding transfers function is not BIBO stable. In order to develop stable and realisable transfer function, the infinite duration impulse response is converted to a finite duration impulse response by truncating the impulse response to a finite number of terms. The magnitude response of the FIR low-pass filter obtained by truncating the impulse response of the ideal low-pass filter does not have a sharp transition from passband to stop-band but exhibits a gradual roll-off.

Example 7.15 *Design a filter with*

$$H_d\left(e^{-j\omega}\right)=\begin{cases} e^{-j3\omega}, & -\frac{\pi}{4}\le\omega\le\frac{\pi}{4} \\ 0, & \frac{\pi}{4}\le|\omega|\le\pi \end{cases}$$

using a Hamming window with M = 7.

Solution Given

$$H_d\left(e^{-j\omega}\right)=\begin{cases} e^{-j3\omega}, & -\frac{\pi}{4}\le\omega\le\frac{\pi}{4} \\ 0, & \frac{\pi}{4}\le|\omega|\le\pi \end{cases}$$

Here, $\alpha = 3$; $M = 7$ and $\frac{M-1}{2}=3$. Since the frequency response has a term $e^{-j\omega\alpha}$, the impulse response $h(n)$ is symmetric about $\alpha = 3$.

$$h_d(n)=\frac{1}{2\pi}\int_{-\pi/4}^{\pi/4} e^{-j3\omega}e^{j\omega n}\,d\omega$$

$$=\frac{1}{2\pi}\int_{-\pi/4}^{\pi/4} e^{j(n-3)\omega}\,d\omega \quad =\frac{\sin\frac{\pi}{4}(n-3)}{\pi(n-3)}$$

$h_d(0) = h_d(6) = 0.075$

$h_d(1) = h_d(5) = 0.159$

$h_d(2) = h_d(4) = 0.22$

$$h_d(3)=\underset{n\to 3}{Lt}\frac{\sin\frac{\pi}{4}(n-3)}{4\frac{\pi}{4}(n-3)}=\frac{1}{4}=0.25$$

$$w_H(n)=\begin{cases} 0.54-0.46\cos\left(\frac{2\pi n}{M-1}\right), & 0\le n\le M-1 \\ 0, & \text{otherwise} \end{cases}$$

$$w_H(n) = \begin{cases} 0.54 - 0.46\cos\left(\dfrac{2\pi n}{6}\right), & 0 \le n \le 6 \\ 0, & \text{otherwise} \end{cases}$$

$w_H(0) = 0.08$
$w_H(1) = 0.31$
$w_H(2) = 0.77$
$w_H(3) = 1$
$w_H(4) = 0.77$
$w_H(5) = 0.31$
$w_H(6) = 0.08$

The filter coefficients are

$h(n) = h_d(n) \times w_H(n)$

$h(0) = h(6)$
$= h_d(0) \times w_H(0) = 0.006$

$h(1) = h(5)$
$= h_d(0) \times w_H(1) = 0.049$

$h(2) = h(4)$
$= h_d(0) \times w_H(2) = 0.1694$

$h(3) = h_d(3) w_H(s) = 0.25$

The transfer function of the filter is

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

$$= \sum_{n=0}^{6} h(n) z^{-n}$$

$= h(0) + h(1)\, z^{-1} + h(2)\, z^{-2} + h(3)\, z^{-3} + h(4)\, z^{-4} + h(5)\, z^{-5} + h(6)\, z^{-6}$

$= h(0)\,[\,1 + z^{-6}] + h(1)\,[z^{-1} + z^{-5}] + h(2)\,[z^{-2} + z^{-4}] + h(3) z^{-3}$

$= 0.006\,[\,1 + z^{-6}] + 0.049\,[z^{-1} + z^{-5}] + 0.1694\,[z^{-2} + z^{-4}] + 0.25\, z^{-3}$

Example 7.16 *Design an ideal highpass filter with a frequency response*

$$H_d\left(e^{j\omega}\right) = \begin{cases} 1, & \dfrac{\pi}{4} \le |\omega| \le \pi \\ 0, & |\omega| \le \dfrac{\pi}{4} \end{cases}$$

Find h(n) and H(z) for M = 11 using (a) Hamming Window (b) Hanning Window and plot the magnitude response.

Solution $h_d(n) = \dfrac{1}{2\pi}\displaystyle\int_{-\pi}^{\pi} H_d\left(e^{j\omega}\right) e^{j\omega n} d\omega \;= \dfrac{1}{2\pi}\left[\int_{-\pi}^{-\pi/4} e^{j\omega n} d\omega + \int_{\pi/4}^{\pi} e^{j\omega n} d\omega\right]$

$$= \frac{1}{\pi n}\left[\sin(\pi n) - \sin\left(\frac{\pi}{4} n\right)\right]$$

For $|n| \le 5$,

$$h_d(0) = \underset{n\to 0}{Lt}\ \frac{1}{\pi n}\left(\sin(\pi n) + \sin\left(\frac{\pi}{4}n\right)\right) = 0.75$$

$h_d(1) = h_d(-1) = -0.225$

$h_d(2) = h_d(-2) = -0.159$

$h_d(3) = h_d(-3) = -0.075$

$h_d(4) = h_d(-4) = -0$

$h_d(5) = h_d(-5) = -0.045$

(a) Using Hamming Window

The Hamming window sequence is given by

$$w_H(n) = \begin{cases} 0.54 + 0.46\cos\dfrac{2\pi n}{M-1}, & -\dfrac{M-1}{2} \le n \le \dfrac{M-1}{2} \\ 0, & \text{otherwise} \end{cases}$$

The window sequence for $M = 11$ is given by

$$w_H(n) = \begin{cases} 0.54 + 0.46\cos\dfrac{\pi n}{5}, & -5 \le n \le 5 \\ 0, & \text{otherwise} \end{cases}$$

$w_H(0) = 0.54 + 0.46 = 1$

$$w_H(-1) = w_H(1) = 0.54 + 0.46\cos\frac{\pi}{5} = 0.912$$

$$w_H(-2) = w_H(2) = 0.54 + 0.46\cos\frac{2\pi}{5} = 0.682$$

$$w_H(-3) = w_H(3) = 0.54 + 0.46\cos\frac{3\pi}{5} = 0.398$$

$$w_H(-4) = w_H(4) = 0.54 + 0.46\cos\frac{4\pi}{5} = 0.1678$$

$$w_H(-5) = w_H(5) = 0.54 + 0.46\cos\frac{5\pi}{5} = 0.08$$

The filter coefficients using Hamming window sequence are

$$h(n) = \begin{cases} h_d(n)\, w_H(n), & -5 \le n \le 5 \\ 0, & \text{otherwise} \end{cases}$$

$h(0) = h_d(0)\, w_H(0) = (1)\,(0.75) = 0.75$

$h(1) = h(-1) = h_d(1)\, w_H(1) = (-0.225)(0.912) = -0.2052$

$h(2) = h(-2) = h_d(2)\, w_H(2) = (-0.159)(0.682) = -0.1084$

$h(3) = h(-3) = h_d(3)\, w_H(3) = (-0.075)(0.398) = -0.03$

$h(4) = h(-4) = h_d(4)\, w_H(4) = (0)(0.1678) = 0$

$h(5) = h(-5) = h_d(5)\, w_H(5) = (-0.045)(0.08) = -0.0036$

The transfer function of the filter is given by

$$H(z) = h(0) + \sum_{n=1}^{5}\left[\ h(z)\left(z^{-n} + z^{n}\right)\right]$$

$$= 0.75 - 0.2052\ (z^{-1} + z) - 0.1084\ (z^{-2} + z^{2}) - 0.03\ (z^{-3} + z^{3}) + 0.0036\ (z^{-5} + z^{5})$$

The transfer function of the realisable filter is

$$H'\ (z) = z^{-5}\ H(z)$$

$$= 0.0036 - 0.03z^{-2} - 0.1084z^{-3} - 0.2052z^{-4} + 0.75z^{-5} - 0.2052z^{-6} - 0.1084z^{-7} - 0.03z^{-8} - 0.0036z^{-10}$$

The magnitude response of the realisable filter is given by

$$\left|H\left(e^{j\omega}\right)\right| = \sum_{n=0}^{\frac{M-1}{2}} a(n)\ \cos \omega n$$

$$a(0)\ =\ h\left[\frac{M-1}{2}\right] = h\ (5) = 0.75$$

$$a(n)\ = 2\ h\left[\frac{M-1}{2} - n\right]$$

$a(1) = 2h[5–1] = 2h\ (4) = –0.4104$

$a(2) = 2h[5–2] = 2h\ (3) = –0.2168$

$a(3) = 2h[5–3] = 2h\ (2) = –0.06$

$a(4) = 2h[5–4] = 2h\ (1) = 0$

$a(5) = 2h[5–5] = 2h\ (0) = –0.0072$

$H(e^{j\omega}) = 0.75 - 0.4104 \cos\omega - 0.2168\ \cos 2\omega - 0.06 \cos 3\omega - 0.0072 \cos 5\omega$

The magnitude response is

ω	0	$\frac{\pi}{12}$	$\frac{\pi}{6}$	$\frac{\pi}{4}$	$\frac{\pi}{3}$	$\frac{5\pi}{12}$	$\frac{\pi}{2}$	$\frac{7\pi}{12}$	$\frac{2\pi}{3}$	$\frac{3\pi}{4}$	$\frac{5\pi}{6}$	$\frac{11\pi}{12}$	π
$\lvert H(e^{j\omega})\rvert$ in dB	–25.09	–18.31	–10.68	–5.89	–2.98	–1.24	–0.29	–0.074	0.06	–0.06	0.08	–0.02	0.09

The magnitude response of the filter is shown in Fig. E7.16(a).

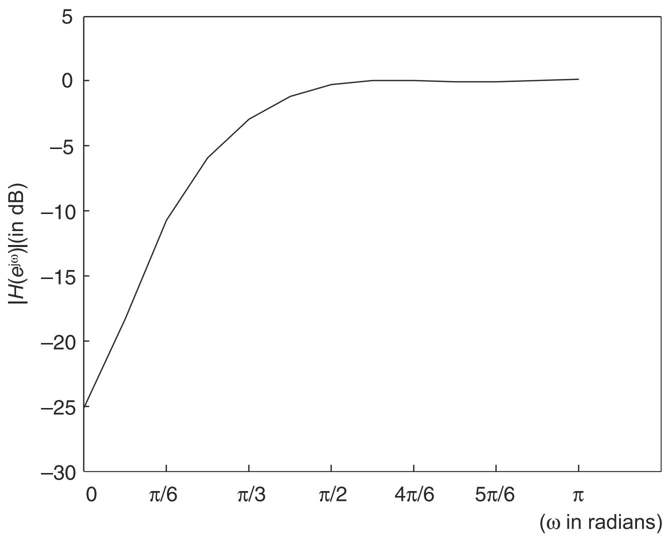


Fig. E7.16(a) *Magnitude Response*

(b) *Using Hanning Window*

$$w_{Hann}(n)=\begin{cases}0.5+0.5\cos\dfrac{2\pi n}{M-1}, & -\dfrac{M-1}{2}\le n\le\dfrac{M-1}{2}\\ 0, & \text{otherwise}\end{cases}$$

For $M = 11$,

$$w_{Hann}(n)=\begin{cases}0.5+0.5\cos\dfrac{\pi n}{5}, & -5\le n\le 5\\ 0, & \text{otherwise}\end{cases}$$

Therefore, $w_{Hann}(0) = 0.5 + 0.5 = 1$

$$w_{Hann}(1)=w_{Hann}(-1)=0.5+0.5\ \cos\frac{\pi}{5}=0.9045$$

$$w_{Hann}(2)=w_{Hann}(-2)=0.5+0.5\cos\frac{2\pi}{5}=0.655$$

$$w_{Hann}(3)=w_{Hann}(-3)=0.5+0.5\cos\frac{3\pi}{5}=0.345$$

$$w_{Hann}(4)=w_{Hann}(-4)=0.5+0.5\ \cos\frac{4\pi}{5}=0.0945$$

$$w_{Hann}(5)=w_{Hann}(-5)=0.5+0.5\ \cos\pi=0$$

The filter coefficients using Hanning Window are

$$h(n)=\begin{cases}h_d(n)w_{Hann}(n), & -5\le n\le 5\\ 0, & \text{otherwise}\end{cases}$$

$h(0) = h_d(0).\ w_{Hann}(0) = (0.75)\ (1) = 0.75$

$h(1) = h(–1) = h_d(1).\ w_{Hann}(1) = (–0.225)(0.905) = –0.204$

$h(2) = h(–2) = h_d(2).\ w_{Hann}(2) = (–0.159)(0.655) = –0.104$

$h(3) = h(–3) = h_d(3).\ w_{Hann}(3) = (–0.075)(0.345) = –0.026$

$h(4) = h(–4) = h_d(4).\ w_{Hann}(4) = (0)(0.8145) = 0$

$h(5) = h(–5) = h_d(5).\ w_{Hann}(5) = (0.045)(0) = 0$

Hence, the transfer function of the filter is given by

$$H(z)=h(0)+\sum_{n=1}^{5}h(n)\left[z^{-n}+z^{n}\right] = 0.75 - 0.204(z^{-1} + z) - 0.104(z^{-2} + z) - 0.206(z^{-3} + z^{3})$$

The transfer function of the realisable filter is

$$H'(z) = z^{-5}\ H(z)$$
$$= -0.026z^{-2} - 0.104z^{-3} - 0.204z^{-4} + 0.752z^{-5} - 0.204z^{-6} - 0.104z^{-7} - 0.026z^{-8}$$

The magnitude response of the realisable filter is given by

$$\left|H\left(e^{j\omega}\right)\right| = \sum_{n=0}^{\frac{M-1}{2}} a(n)\ \cos\omega n$$

$$a(0)=h\left(\frac{M-1}{2}\right)=h\ (5)=0.75$$

$$a(n)=2\ h\left(\frac{M-1}{2}-n\right)$$

$a(1) = 2h(5–1) = 2h\,(4) = –0.408$ $a(2) = 2h(5–2) = 2h\,(3) = –0.208$
$a(3) = 2h(5–3) = 2h\,(2) = –0.052$ $a(4) = 2h(5–4) = 2h\,(1) = 0$
$a(5) = 2h(5–5) = 2h\,(0) = 0$

$H(e^{j\omega}) = 0.75 – 0.408\cos\omega – 0.208\cos 2\omega – 0.052\cos 3\omega$

The magnitude response is

ω	0	$\frac{\pi}{12}$	$\frac{\pi}{6}$	$\frac{\pi}{4}$	$\frac{\pi}{3}$	$\frac{5\pi}{12}$	$\frac{\pi}{2}$	$\frac{7\pi}{12}$	$\frac{2\pi}{3}$	$\frac{3\pi}{4}$	$\frac{5\pi}{6}$	$\frac{11\pi}{12}$	π
$\lvert H(e^{j\omega})\rvert$ in dB	–21.72	–17.14	–10.67	–6.05	–3.07	–1.297	–0.3726	–0.0087	0.052	0.015	0	0	0.017

The magnitude response of the filter is shown in Fig. E7.16(b).

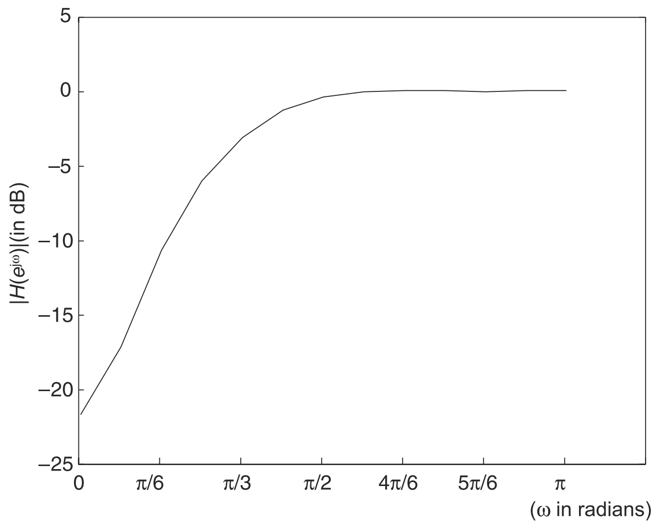


Fig. E7.16(b) *Magnitude Response*

Example 7.17 *A low-pass filter is to be designed with the following desired frequency response*

$$H_d(e^{j\omega}) = \begin{cases} e^{-j2\omega}, & -\pi/4 \le \omega \le \pi/4 \\ 0, & \pi/4 < |\omega| \le \pi \end{cases}$$

Determine the filter coefficients $h_d(n)$ if the window function is defined as $w(n) = \begin{cases} 1, & 0 \le n \le 4 \\ 0, & otherwise \end{cases}$
Also, determine the frequency response $H(e^{j\omega})$ of the designed filter.

Solution Given $H_d(e^{j\omega}) = \begin{cases} e^{-j2\omega}, & -\pi/4 \le \omega \le \pi/4 \\ 0, & \pi/4 < |\omega| \le \pi \end{cases}$

Therefore,

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}\,d\omega$$

$$= \frac{1}{2\pi}\int_{-\pi/4}^{\pi/4} e^{-j2\omega}e^{j\omega n}\,d\omega = \frac{1}{2\pi}\int_{-\pi/4}^{\pi/4} e^{j\omega(n-2)}\,d\omega$$

$$= \frac{1}{\pi(n-2)}\left[\frac{e^{j(n-2)\pi/4} - e^{-j(n-2)\pi/4}}{2j}\right] = \frac{1}{\pi(n-2)}\sin\frac{\pi}{4}(n-2), \quad n \ne 2$$

For $n = 2$, the filter coefficient can be obtained by applying L'Hospital's rule to the above expression. Thus,

$$h_d(2) = \frac{1}{4}$$

The other filter coefficients are given by

$$h_d(0) = \frac{1}{2\pi} = h_d(4) \quad \text{and} \quad h_d(1) = \frac{1}{\sqrt{2}\pi} = h_d(3)$$

The filter coefficients of the filter would be then

$$h(n) = h_d(n).w(n)$$

Therefore,

$$h(0) = \frac{1}{2\pi} = h(4), \quad h(1) = \frac{1}{\sqrt{2}\pi} = h(3) \quad \text{and} \quad h(2) = \frac{1}{4}$$

The frequency response $H(e^{j\omega})$ is given by

$$H(e^{j\omega}) = \sum_{n=0}^{4} h(n)e^{-j\omega n}$$

$$= h(0) + h(1)e^{-j\omega} + h(2)e^{-j2\omega} + h(3)e^{-j3\omega} + h(4)e^{-j4\omega}$$

$$= e^{-j2\omega}[h(0)e^{j2\omega} + h(1)e^{j\omega} + h(2) + h(3)e^{-j\omega} + h(4)e^{-j2\omega}]$$

$$= e^{-j2\omega}\{h(2) + h(0)[e^{j2\omega} + e^{-j2\omega}] + h(1)[e^{j\omega} + e^{-j\omega}]\}$$

$$= e^{-j2\omega}\left\{\frac{1}{4} + \frac{1}{2\pi}\left[e^{j2\omega} + e^{-j2\omega}\right] + \frac{1}{\sqrt{2}\pi}\left[e^{j\omega} + e^{-j\omega}\right]\right\}$$

The frequency response of the designed low-pass filter is then,

$$H(e^{j\omega}) = e^{-j2\omega}\left\{\frac{1}{4} + \frac{\sqrt{2}}{\pi}\cos\omega + \frac{1}{\pi}\cos 2\omega\right\}$$

Example 7.18 *A filter is to be designed with the following desired frequency response*

$$H_d(e^{j\omega}) = \begin{cases} 0, & -\pi/4 \le \omega \le \pi/4 \\ e^{-j2\omega}, & \pi/4 < |\omega| \le \pi \end{cases}$$

Determine the filter coefficients $h_d(n)$ if the window function is defined as $w(n) = \begin{cases} 1, & 0 \le n \le 4 \\ 0 & \textit{otherwise} \end{cases}$

Also, determine the frequency response $H(e^{j\omega})$ of the designed filter.

Solution

Given $H_d(e^{j\omega}) = \begin{cases} 0, & -\pi/4 \le \omega \le \pi/4 \\ e^{-j2\omega}, & \pi/4 < |\omega| \le \pi \end{cases}$

Therefore,

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}\,d\omega$$

$$= \frac{1}{2\pi}\int_{-\pi}^{-\pi/4} e^{-j2\omega}e^{j\omega n}\,d\omega + \frac{1}{2\pi}\int_{\pi/4}^{\pi} e^{-j2\omega}e^{j\omega n}\,d\omega$$

$$= \frac{1}{2\pi}\int_{-\pi}^{-\pi/4} e^{j\omega(n-2)}\,d\omega + \frac{1}{2\pi}\int_{\pi/4}^{\pi} e^{j\omega(n-2)}\,d\omega$$

$$= \frac{1}{\pi(n-2)}\left\{\left[\frac{e^{j(n-2)\pi} - e^{-j(n-2)\pi}}{2j}\right] - \left[\frac{e^{j(n-2)\pi/4} - e^{-j(n-2)\pi/4}}{2j}\right]\right\}$$

$$= \frac{1}{\pi(n-2)}[\sin \pi(n-2) - \sin(n-2)\pi/4], \quad n \neq 2$$

The filter coefficients are given by,

$$h_d(2) = \frac{3}{4}, \quad h_d(0) = \frac{1}{2\pi} = h_d(4) \text{ and } h_d(1) = \frac{1}{\sqrt{2}\pi} = h_d(3)$$

and by applying the window function, the new filter coefficients are

$$h(2) = \frac{3}{4}, \quad h(0) = \frac{1}{2\pi} = h(4) \text{ and } h(1) = \frac{1}{\sqrt{2}\pi} = h(3)$$

The frequency response $H(e^{j\omega})$, is obtained as in the previous example,

$$H(e^{j\omega}) = e^{-j2\omega}\left[0.75 - \frac{\sqrt{2}}{\pi}\cos\omega - \frac{1}{\pi}\cos 2\omega\right]$$

Example 7.19 *A low-pass filter should have the frequency response given below. Find the filter coefficients $h_d(n)$. Also determine τ so that $h_d(n) = h_d(-n)$.*

$$H_d(e^{j\omega}) = \begin{cases} e^{-j\omega\tau}, & -\omega_c \leq \omega \leq \omega_c \\ 0, & \omega_c < |\omega| \leq \pi \end{cases}$$

Solution The filter coefficients are given by

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}\,d\omega = \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} e^{-j\omega\tau}e^{j\omega n}\,d\omega$$

$$h_d(n) = \frac{\sin\omega_c(n-\tau)}{\pi(n-\tau)}, \quad n \neq \tau \text{ and } h_d(\tau) = \frac{\omega_c}{\pi}$$

when $h_d(n) = h_d(-n)$,

$$\frac{\sin\omega_c(n-\tau)}{\pi(n-\tau)} = \frac{\sin\omega_c(-n-\tau)}{\pi(-n-\tau)}$$

That is,

$$\frac{\sin\omega_c(n-\tau)}{\pi(n-\tau)} = \frac{-\sin\omega_c(n+\tau)}{-\pi(n+\tau)} = \frac{\sin\omega_c(n+\tau)}{\pi(n+\tau)}$$

This is possible only when $(n - \tau) = (n + \tau)$ or $\tau = 0$.

Example 7.20 *The desired response of a low-pass filter is*

$$H_d(e^{j\omega}) = \begin{cases} e^{-j3\omega}, & -3\pi/4 \leq \omega \leq 3\pi/4 \\ 0, & 3\pi/4 < |\omega| \leq \pi \end{cases}$$

Determine $H(e^{j\omega})$ for $M = 7$ using a Hamming window.

Solution The filter coefficients are given by

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}\,d\omega = \frac{1}{2\pi}\int_{-3\pi/4}^{3\pi/4} e^{-j3\omega}e^{j\omega n}\,d\omega$$

$$h_d(n) = \frac{\sin 3\pi(n-3)/4}{\pi(n-3)}, n \neq 3 \text{ and } h_d(3) = \frac{3}{4}$$

The filter coefficients are

$$h_d(0) = 0.0750,\ h_d(1) = -0.1592,\ h_d(2) = 0.2251,\ h_d(3) = 0.75$$
$$h_d(4) = 0.2251,\ h_d(5) = -0.1592,\ h_d(6) = 0.0750$$

The Hamming window function is

$$w_H(n) = \begin{cases} 0.54 - 0.46\cos\dfrac{2\pi n}{M-1}, & 0 \le n \le M-1 \\ 0, & \text{otherwise} \end{cases}$$

Therefore, with $M = 7$,

$$w_H(0) = 0.08,\ w_H(1) = 0.31,\ w_H(2) = 0.77,\ w_H(3) = 1,\ w_H(4) = 0.77,$$
$$w_H(5) = 0.31,\ w_H(6) = 0.08.$$

The filter coefficients of the resultant filter are then,

$$h(n) = h_d(n) . w_H(n) \text{ for } n = 0, 1, 2, 3, 4, 5, 6.$$

Therefore,

$$h(0) = 0.006,\ h(1) = -0.0494,\ h(2) = 0.1733,\ h(3) = 0.75,$$
$$h(4) = 0.1733,\ h(5) = -0.0494 \text{ and } h(6) = 0.006.$$

The frequency response is given by

$$H(e^{j\omega}) = \sum_{n=0}^{6} h(n)e^{-j\omega n}$$
$$= e^{-j3\omega}[h(3) + 2h(0)\cos 3\omega + 2h(1)\cos 2\omega + 2h(2)\cos\omega]$$
$$= e^{-j3\omega}[0.75 + 0.3466\cos\omega - 0.0988\cos 2\omega + 0.012\cos\omega]$$

Example 7.21 *Design a high pass filter using Hamming window with a cut-off frequency of 1.2 rad and M = 9.*

Solution $\omega_c = 1.2$ rad and $M = 9$

$$H_d(e^{j\omega}) = \begin{cases} e^{-j\omega\alpha}, & -\pi \le \omega \le -\omega_c \text{ and } \omega_c \le \omega \le \pi \\ 0, & \text{otherwise} \end{cases}$$

To find $h_d(n)$

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}\,d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{-j\omega\alpha}e^{j\omega n}\,d\omega$$
$$= \frac{1}{2\pi}\int_{-\pi}^{-\omega_c} e^{-j\omega\alpha}e^{j\omega n}\,d\omega + \frac{1}{2\pi}\int_{\omega_c}^{\pi} e^{-j\omega\alpha}e^{j\omega n}\,d\omega$$
$$= \frac{1}{2\pi}\int_{-\pi}^{-\omega_c} e^{j\omega(n-\alpha)}\,d\omega + \frac{1}{2\pi}\int_{\omega_c}^{\pi} e^{j\omega(n-\alpha)}\,d\omega = \frac{1}{2\pi}\left[\frac{e^{j\omega(n-\alpha)}}{j(n-\alpha)}\right]_{-\pi}^{-\omega_c} + \frac{1}{2\pi}\left[\frac{e^{j\omega(n-\alpha)}}{j(n-\alpha)}\right]_{\omega_c}^{\pi}$$
$$= \frac{1}{2\pi}\left[\frac{e^{-j\omega_c(n-\alpha)} - e^{-\pi(n-\alpha)} + e^{j\omega(n-\alpha)} - e^{j\omega_c(n-\alpha)}}{j(n-\alpha)}\right]$$
$$= \frac{\sin(n-\alpha)\pi - \sin\omega_c(n-\alpha)}{\pi(n-\alpha)}$$

$$h_d(n) = \frac{\sin(n-\alpha)\pi - \sin\omega_c(n-\alpha)}{\pi(n-\alpha)} \text{ for } n \neq \alpha$$

$$h_d(n) = \frac{1}{\pi}\left(\underset{n\to\infty}{Lt} \frac{\sin(n-\alpha)\pi}{(n-\alpha)} - \underset{n\to\infty}{Lt} \frac{\sin(n-\alpha)\pi}{(n-\alpha)} \right)$$

$$= \frac{1}{\pi}(\pi - \omega_c) = \left[1 - \frac{\omega_c}{\pi}\right], \quad \text{for} \quad n \neq \alpha$$

$$h(n) = h_d(n) w_H(n)$$

$w_H(n)$ = Window sequence for Hamming window.

$$= 0.54 - 0.46\cos\left[\frac{2\pi n}{M-1}\right] \quad \text{for} \quad n = 0 \text{ to } M-1$$

$$h(n) = \frac{1}{\pi(n-\alpha)}\left[\sin(n-\alpha)\pi - \sin(n-\alpha)\omega_c\right]\left[0.54 - 0.46\left[\frac{2\pi n}{M-1}\right]\right]; \text{for } n \neq \alpha$$

$$= \left(1 - \frac{\omega_c}{\pi}\right)\left[0.54 - 0.46\cos\left(\frac{2\pi n}{M-1}\right)\right]; \text{for } n = \alpha$$

Here

$$\alpha = \frac{M-1}{2} = \frac{5-1}{2} = 4$$

$$h(n) = \frac{-\sin(n-4)\omega_c}{\pi(n-4)}\left[0.54 - 0.46\cos\frac{n\pi}{4}\right]; \text{for } n \neq 4$$

$$= \left(1 - \frac{\omega_c}{\pi}\right) \times \left[0.54 - 0.46\cos\frac{n\pi}{4}\right]; \text{for } n = 4$$

$$h(0) = \frac{-\sin(-4)\times(1.2)}{\pi\times(0-4)}\left[0.54 - 0.46\cos 0\right] = 0.0063$$

Similarly,

$h(1) = 0.0101$
$h(2) = 0.0581$
$h(3) = 0.2567$

$$h(4) = \left[\left(1 - \frac{1.2}{\pi}\right) \times (0.54 - 0.46\cos\pi)\right] = 0.6180$$

$h(5) = -0.2567$
$h(6) = -0.0581$
$h(7) = -0.0101$
$h(8) = -0.0063$

Since impulse response is symmetrical with centre of symmetry at $n = 4$.

$h(0) = h(8)$
$h(1) = h(7)$
$h(7) = h(6)$
$h(8) = h(5)$

To find magnitude response

$$|H(\omega)| = h\left(\frac{M-1}{2}\right) + \sum_{n=1}^{\frac{M-1}{2}} 2h\left(\frac{M-1}{2} - n\right)\cos\omega n$$

$$= h(4) + 2h(3)\cos\omega + 2h(2)\cos 2\omega + 2h(1)\cos 3\omega + 2h(0)\cos 4\omega$$
$$= 0.618 - 0.5134\cos\omega - 0.1162\cos 2\omega - 0.0202\cos 3\omega + 0.0126\cos 4\omega$$

To find the transfer function

$$H(z) = \sum_{n=0}^{M-1} h(n)z^{-n}$$
$$= h(0)\left[z^{0} + z^{-8}\right] + h(1)\left[z^{-1} + z^{-7}\right] + h(2)\left[z^{-2} + z^{-6}\right] + h(3)\left[z^{-3} + z^{-5}\right] + h(4)z^{-4}$$
$$H(z) = \frac{Y(z)}{X(z)}$$

Therefore,

$$Y(z) = h(0)\left[X(z) + z^{-8}X(z)\right] + h(1)\left[z^{-1}X(z) + z^{-7}X(z)\right] + h(2)\left[z^{-2}X(z) + z^{-6}X(z)\right] + h(3)\left[z^{-3}X(z) + z^{-5}X(z)\right] + h(4)\left[z^{-4}X(z)\right]$$

Example 7.22 *Design an FIR digital filter to approximate an ideal low-pass filter with pass-band gain of unity, cut-off frequency of 850 Hz and working at a sampling frequency of f_s = 5000 Hz. The length of the impulse response should be 5. Use a rectangular window.*

Solution The desired response of the ideal low-pass filter is given by

$$H_d(e^{j\omega}) = \begin{cases} 1, & 0 \le f \le 850 \text{ Hz} \\ 0, & f > 850 \text{ Hz} \end{cases}$$

The above response can be equivalently specified in terms of the normalised ω_c. The normalised $\omega_c = 2\pi f_c/f_s = 2\pi(850)/(5000) = 1.068$ rad/sec. Hence, the desired response is

$$H_d(e^{j\omega}) = \begin{cases} 1, & 0 \le |\omega| \le 1.068 \\ 0, & 1.068 < |\omega| \le \pi \end{cases}$$

The filter coefficients are given by

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}\,d\omega = \frac{1}{2\pi}\int_{-1.068}^{1.068} e^{j\omega n}\,d\omega$$

$$h_d(n) = \frac{\sin 1.068n}{\pi n}, \quad n \ne 0 \text{ and } h_d(0) = \frac{1.068}{\pi} = 0.3400$$

Using the rectangular window function and for $M = 5$,

$$h(n) = h_d(n)\,.\,w(n)\; n = 0, 1, 2, 3, 4.$$

Therefore,

$$h(0) = 0.34,\ h(1) = 0.2789,\ h(2) = 0.1344,\ h(3) = -0.0066,$$
$$h(4) = -0.0720.$$

Example 7.23 *An FIR linear phase filter has the following impulse response.*

$$h_d(n) = \begin{cases} 1, & \text{for } 0 \le n \le 4 \\ 0, & \text{otherwise} \end{cases}$$

Use Bartlett's window and compute the impulse response of the filter. Find its magnitude and phase response as a function of frequency.

Solution For the given FIR filter, $h_d(n) = \begin{cases} 1, & \text{for} \quad 0 \le n \le 4 \\ 0, & \text{otherwise} \end{cases}$ and $M = 5$

To find the impulse response of the filter using Bartlett's window:

Magnitude Response $|H(e^{j\omega})| = |1 + \cos\omega|$

ω	0	$\pi/6$	$\pi/4$	$\pi/3$	$\pi/2$	$2\pi/3$	$3\pi/4$	$5\pi/6$	π
$\lvert H(e^{j\omega})\rvert$	2	1.866	1.707	1.5	1	0.5	0.293	0.134	0

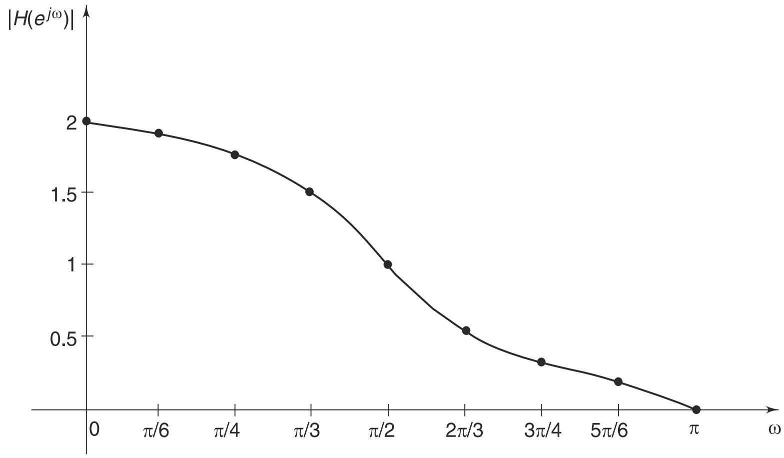


Fig. E7.23 (a)

Phase Response $\phi(\omega) = \angle H(e^{j\omega}) = -2\omega$

ω	$-\pi$	$-3\pi/$	$-\pi/2$	$-\pi/$	0	$\pi/4$	$\pi/2$	$3\pi/4$	π
$\phi(\omega)$	2π	$3\pi/2$	π	$\pi/2$	0	$-\pi/2$	$-\pi$	$-3\pi/2$	-2π

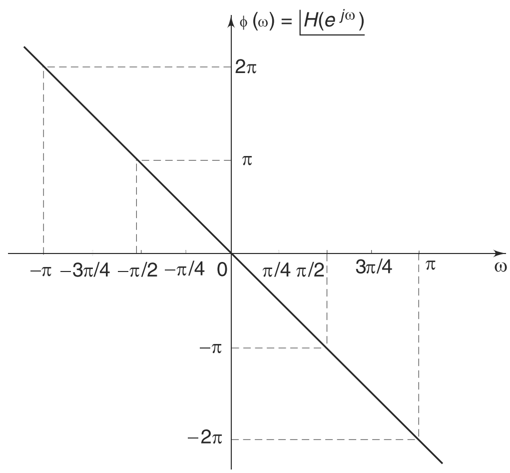


Fig. E7.23 (b)

Step I: To find $h(n)$:

$$w_{Bart}(n) = \begin{cases} 1 - \dfrac{2\left|n - \dfrac{M-1}{2}\right|}{M-1}, & 0 \le n \le M-1 \\ 0, & \text{otherwise} \end{cases}$$

Given $M = 5$ (odd)

For linear phase FIR filter,

$h(n) = h_d(n)\, w_{Bart}(n)$

$h(0) = h(4),\ h(1) = h(3)$

$w_{Bart}(0) = 0,\ w_{Bart}(1) = 0.5,\ w_{Bart}(2) = 1$

$h(0) = h(4) = 0,\ h(1) = h(3) = 0.5,\ h(2) = 1$

Step II: To find $H(e^{j\omega})$:

$$H(e^{j\omega}) = e^{-j\omega\left(\frac{M-1}{2}\right)}\left[h\left(\frac{M-1}{2}\right) + 2\sum_{n=0}^{(M-3)/2} h(n)\cos\left(\frac{\omega}{\frac{M-1}{2} - n}\right)\right]$$

$$= e^{-j2\omega}\left[h(2) + 2\sum_{n=0}^{1} h(n)\cos(\omega(2-n))\right] = e^{-j2\omega}\left[1 + 2[0.5\cos(\omega(1))]\right]$$

Therefore, $H(e^{j\omega}) = e^{-j2\omega}\,[1 + \cos\omega]$

The magnitude response and phase response of the given FIR filter are shown in Fig. E7.23(a) and Fig. E7.23(b) respectively.

Example 7.24 *Design a bandbass filter to pass frequencies in the range 1–2 rad using Hanning window M = 5.*

Solution Given $\omega_c = 1$ to 2 rad and $M = 5$.

$$H_d(e^{j\omega}) = \begin{cases} e^{-j\omega\alpha}, & -\omega_{c2} \le \omega \le \omega_{c1} \text{ and } \omega_{c1} \le \omega \le \omega_{c2} \\ 0, & \text{otherwise} \end{cases}$$

To find h

To find $h_d(n)$

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(\omega)e^{j\omega n}\,d\omega$$

$$= \frac{1}{2\pi}\int_{-\omega_{c2}}^{-\omega_{c1}} e^{-j\omega\alpha}e^{j\omega n}\,d\omega + \frac{1}{2\pi}\int_{\omega_{c1}}^{\omega_{c2}} e^{-j\omega\alpha}e^{j\omega n}\,d\omega$$

$$= \frac{1}{2\pi}\left[\frac{e^{j\omega(n-\alpha)}}{j(n-\alpha)}\right]_{-\omega_{c2}}^{-\omega_{c1}} + \frac{1}{2\pi}\left[\frac{e^{j\omega(n-\alpha)}}{j(n-\alpha)}\right]_{\omega_{c1}}^{\omega_{c2}}$$

$$= \frac{1}{2\pi}\left[\frac{e^{j\omega_{c1}(n-\alpha)} - e^{-j\omega_{c2}(n-\alpha)}}{j(n-\alpha)}\right] + \frac{1}{2\pi}\left[\frac{e^{j\omega_{c2}(n-\alpha)} - e^{j\omega_{c1}(n-\alpha)}}{j(n-\alpha)}\right]$$

$$h_d(n) = \frac{1}{\pi(n-\alpha)}\left[\sin\omega_{c2}(n-\alpha) - \sin\omega_{c1}(n-\alpha)\right], \text{ for } n \ne \alpha$$

$$h_d(n) = \frac{1}{\pi}\left(\underset{n\to\alpha}{\text{Lt}}\frac{\sin\omega_{c2}(n-\alpha)}{(n-\alpha)} - \underset{n\to\alpha}{\text{Lt}}\frac{\sin\omega_{c1}(n-\alpha)}{(n-\alpha)}\right)$$

$$h_d(n) = \frac{\omega_{c2} - \omega_{c1}}{\pi}, \text{ for } n = \alpha$$

$h(n) = h_d(n)w_{Hann}(n)$, where $w_{Hann}(n)$ is the Hanning Window Sequence.

$$w_{Hann}(n) = \left(0.5 - 0.5\cos\frac{2\pi n}{M-1}\right), \text{ for } n = 0 \text{ to } M-1.$$

$$h(n) = \frac{\sin[\omega_{c2}(n-\alpha)] - \sin[\omega_{c1}(n-\alpha)]}{\pi(n-\alpha)} \times \left[0.5 - 0.5\cos\frac{2\pi n}{M-1}\right], \text{ for } n \neq \alpha$$

$$h(n) = \left[\frac{\omega_{c2} - \omega_{c1}}{\pi}\right]\left[0.5 - 0.5\cos\frac{2\pi n}{M-1}\right], \quad \text{for } n = \alpha$$

Here $\alpha = \frac{M-1}{2} = \frac{5-1}{2} = 2$

$$h(0) = \frac{\sin[2\times(0-2)] - \sin[1\times(0-2)]}{\pi(0-2)} \times [0.5 - 0.5\cos 0] = 0$$

Similarly,

$h(1) = 0.0108$

$h(2) = \frac{1}{\pi} = 0.3183$

$h(3) = 0.0108$

$$\left.\begin{aligned} h(4) &= 0 \\ h(4) &= h(0) \\ h(3) &= h(1) \end{aligned}\right\} \text{Impulse response is symmetry with centre at } n = 2.$$

To find magnitude response

$$|H(\omega)| = h\left(\frac{M-1}{2}\right) + \sum_{n=1}^{\frac{M-1}{2}} 2h\left(\frac{M-1}{2} - n\right)\cos\omega n$$

$$= h(2) + 2h(1)\cos\omega + 2h(0)\cos 2\omega = 0.3183 + 0.0216\cos\omega + 0 = 0.3183 + 0.0216\cos\omega$$

To find the transfer function

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

$$= h(1)[z^{-1} + z^{-3}] + h(0)[z^{-0} + z^{-4}] + h(2)z^{-2}$$

$$Y(z) = h(1)[z^{-1}X(z) + z^{-3}X(z)] + h(2)[z^{-2}X(z)]$$

Example 7.25 *Using a rectangular window, design a low-pass filter with passband gain of unity, cut-off frequency of 1 kHz and working at a sampling frequency of 5 kHz. The length of the impulse response should be 7.*

Solution Given $f_c = 1$ kHz and $f_s = 5$ kHz

$$\omega_c = 2\pi f_c T = \frac{2\pi f_c}{f_s} = \frac{2\pi\times 1000}{5000} = \frac{2\pi}{5}$$

Therefore, $$H_d\left(e^{j\omega}\right) = \begin{cases} 1, & |\omega| \leq \frac{2\pi}{5} \\ 0, & \text{otherwise} \end{cases}$$

$$h_d(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega = \frac{1}{2\pi}\int_{-2\pi/5}^{2\pi/5} e^{j\omega n}\mathrm{d}\omega = \frac{1}{2\pi}\left[\frac{e^{j\omega n}}{jn}\right]_{-2\pi/5}^{2\pi/5} = \frac{\sin\frac{2\pi}{5}n}{\pi n}$$

The rectangular window is given by

$$w_R(n) = \begin{cases} 1, & -3 \le n \le 3 \\ 0, & \text{otherwise} \end{cases}$$

The filter coefficients are given by

$$h(n) = \begin{cases} h_d(n) w_R(n), & -5 \le n \le 5 \\ 0, & \text{otherwise} \end{cases}$$

$$h(0) = h_d(0)\, w_R(0) = h_d(0)$$

$$= \underset{n\to 0}{Lt} \frac{\sin \frac{2\pi}{5} n}{\pi\, n} = \frac{2}{5} \underset{n\to 0}{Lt} \frac{\sin \frac{2\pi}{5} n}{\frac{2\pi}{5} n} = \frac{2}{5} \underset{n\to 0}{Lt} \operatorname{sinc}\left(\frac{2\pi}{5} n\right) = \frac{2}{5} = 0.4$$

Similarly, we can calculate

$$h(1) = h(-1) = \frac{\sin \frac{2\pi}{5}}{\pi} = 0.3027$$

$$h(2) = h(-2) = \frac{\sin \frac{4\pi}{5}}{\pi} = 0.0935$$

$$h(3) = h(-3) = \frac{\sin \frac{6\pi}{5}}{\pi} = -0.0624$$

The filter coefficients are

$$h(n) = \{-0.0624, 0.0935, 0.3027, 0.4, 0.3027, 0.0935, -0.0624\}$$

The transfer function of the realisable filter is

$$H'(z) = -0.0624 + 0.0935z^{-1} + 0.3027z^{-2} + 0.4z^{-3} + 0.3027z^{-4} + 0.0935z^{-5} - 0.0624z^{-6}$$

7.4.4 Kaiser Window

From the frequency-domain characteristics of the window functions listed in Table 7.1, it can be seen that the width of the main lobe is inversely proportional to the length of the filter. As the length of the filter is increased, the width of the main lobe becomes narrower and narrower, and the transition band is reduced considerably. The attenuation in the side-lobes is, however, independent of the length and is a function of the type of the window. Therefore, a proper window function is to be selected in order to achieve a desired stopband attenuation. A window function with minimum stopband attenuation has the maximum main lobe width. Therefore, the length of the filter must be increased considerably to reduce the main lobe width and to achieve the desired transition band.

A desirable property of the window function is that the function is of finite duration in the time domain and that the Fourier transform has maximum energy in the main lobe or a given peak side lobe amplitude. The prolate spheroidal functions have this desirable property; however, these functions are complicated and difficult to compute. A simple approximation to these functions have been developed by Kaiser in terms of zeroth order modified Bessel functions of the first kind. In a Kaiser window, the side lobe level can be controlled with respect to the mainlobe peak by varying a parameter, α. The width of the main lobe can be varied by adjusting the length of the filter. The Kaiser window function is given by

$$w_K(n) = \begin{cases} \frac{I_0(\beta)}{I_0(\alpha)}, & \text{for } |n| \le \frac{M-1}{2} \\ 0, & \text{otherwise} \end{cases} \tag{7.46}$$

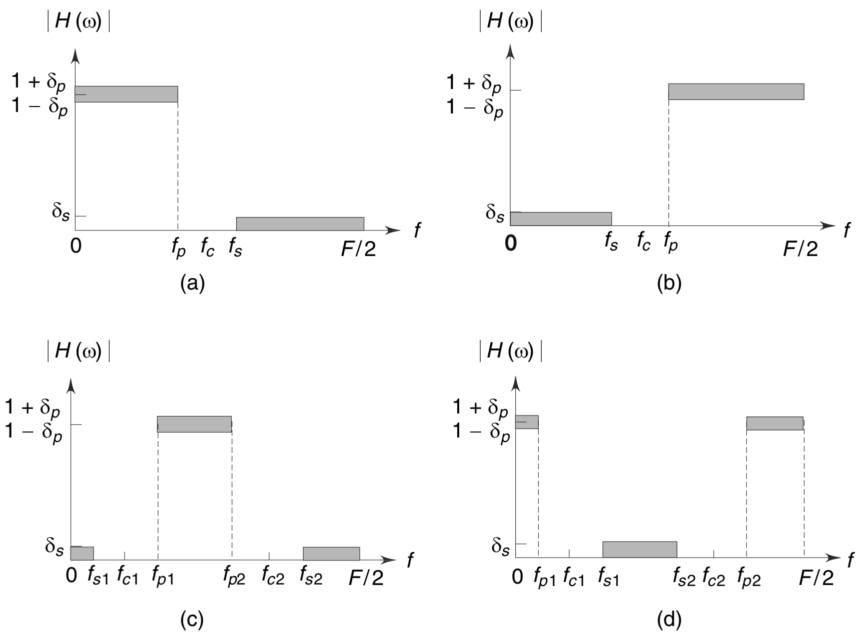


Fig. 7.3 *Idealised Frequency Responses (a) Low-pass Filter; (b) High-pass Filter; (c) Bandpass Filter; (d) Bandstop Filter*

where α is an independent variable determined by Kaiser. The parameter β is expressed by

$$\beta = \alpha\left[1-\left(\frac{2n}{M-1}\right)^2\right]^{0.5} \tag{7.47}$$

The modified Bessel function of the first kind, $I_0(x)$, can be computed from its power series expansion given by

$$I_0(x) = 1+\sum_{k=1}^{\infty}\left[\frac{1}{k!}\left(\frac{x}{2}\right)^k\right]^2$$

$$= 1+\frac{0.25x^2}{(1!)^2}+\frac{(0.25x^2)}{(2!)^2}+\frac{(0.25x^2)^3}{(3!)^2}+\ldots \tag{7.48}$$

Figure 7.3 shows the idealised frequency responses of different filters with their passband and stopband specifications. Considering the design specifications of the filters in Fig. 7.3, the actual passband ripple (A_P) and minimum stopband attenuation (A_S) are given by

$$A_p = 20\log_{10}\frac{1+\delta_p}{1-\delta_p}\,\text{dB} \tag{7.49}$$

and

$$A_S = -20\log_{10}\delta_S\,\text{dB} \tag{7.50}$$

The transition bandwidth is

$$\Delta F = f_S - f_P \tag{7.51}$$

Let and be the specified passband ripple and minimum stopband attenuation, respectively and,

$$A_P \le A'_P$$
$$A_S \le A'_S \tag{7.52}$$

where A_P and A_S are the actual passband peak-to-peak ripple and minimum stopband attenuation, respectively.

7.4.5 Design Using the Kaiser Window Function

Design Specifications

(i) Filter type: Low-pass, high-pass, bandpass or bandstop.
(ii) Passband and stopband frequencies in Hertz:
For low-pass / high-pass: f_P and f_S.
For band-pass / band-stop: f_{P1}, f_{P2}, f_{S1} and f_{S2} .
(iii) Passband ripple and minimum stopband attenuation in positive decibels: A'_P and A'_S.
(iv) Sampling frequency in Hertz: F
(v) Filter order M-odd.

Design Procedure

(i) Determine δ according to Eqs. (7.49), (7.50) and (7.52), where the actual design parameter can be determined from

$$\delta = \min(\delta_P, \delta_S) \tag{7.53}$$

where from Eqs. (7.49) and (7.50), we get

$$\delta_S = 10^{-0.05A'_s} \text{ and } \delta_p = \frac{10^{0.05A'_p} - 1}{10^{0.05A'_p} + 1} \tag{7.54}$$

(ii) Calculate A_S from Eq. (7.50).
(iii) Determine the parameter α from the Kaiser's design equation

$$\alpha = \begin{cases} 0, & \text{for } A_s \le 21 \\ 0.5842(A_s - 21)^{0.4} + 0.07886(A_s - 21), & \text{for } 21 < A_s \le 50 \\ 0.1102(A_s - 8.7), & \text{for } A_s > 50 \end{cases} \tag{7.55}$$

(iv) Determine the parameter D from the Kaiser's design equation

$$D = \begin{cases} 0.9222, & \text{for } A_s \le 21 \\ \dfrac{A_s - 7.95}{14.36}, & \text{for } A_s > 21 \end{cases} \tag{7.56}$$

(v) Calculate the filter order for the lowest odd value of M.

$$M \ge \frac{FD}{\Delta F} + 1 \tag{7.57}$$

(vi) The modified impulse response is computed using

$$h(n) = w_K(n) h_d(n), \quad \text{for} \, |n| \le \frac{M-1}{2} \tag{7.58}$$

(vii) The transfer function is

$$H(z) = z^{-(M-1)/2} \left[h(0) + 2 \sum_{n=1}^{(M-1)/2} h(n) \left(z^n + z^{-n} \right) \right] \tag{7.59}$$

where

$$h(0) = w_K(0)\, h_d(0)$$
$$h(n) = w_K(n)\, h_d(n) \tag{7.60}$$

The magnitude response can be obtained from Eq. (7.59)

$$M(\omega) = h(0) + 2 \sum_{n=1}^{(M-1)/2} h(n) \cos 2\pi f n T \tag{7.61}$$

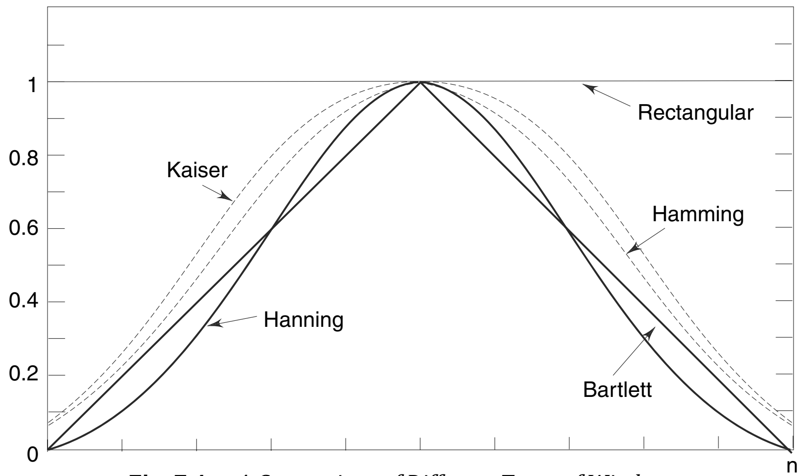


Fig. 7.4 *A Comparison of Different Types of Windows*

Figure 7.4 shows the different window functions.

The design equations for the low-pass, high-pass, band-pass and bandstop FIR digital filters are given below.

Low-Pass FIR Filter

$$h_d(n) = \begin{cases} \left(\dfrac{2f_c}{F}\right)\dfrac{\sin 2\pi n f_c / F}{2\pi n f_c / F}, & \text{for } n > 0 \\ \dfrac{2f_c}{F}, & \text{for } n = 0 \end{cases} \tag{7.62}$$

where

$$f_c = 0.5\,(f_p + f_s) \text{ and } \Delta F = f_s - f_p \tag{7.63}$$

High-Pass FIR Filter

$$h_d(n) = \begin{cases} \left(\dfrac{2f_c}{F}\right)\dfrac{\sin(2\pi n f_c / F)}{(2\pi n f_c / F)}, & \text{for } n > 0 \\ \dfrac{2f_c}{F}, & \text{for } n = 0 \end{cases} \tag{7.64}$$

where

$$f_c = 0.5\,(f_p + f_s) \text{ and } \Delta F = f_p - f_s \tag{7.65}$$

Bandpass FIR Filter

$$h_d(n) = \begin{cases} \dfrac{1}{n\pi}\left[\sin(2\pi n f_{c_2} / F) - \sin(2\pi n f_{c_1} / F)\right], & \text{for } n > 0 \\ \dfrac{2}{F}(f_{c_2} - f_{c_1}), & \text{for } n = 0 \end{cases} \tag{7.66}$$

where

$$f_{c_1} = f_{p_1} - \frac{\Delta F}{2}, \qquad f_{c_2} = f_{p_2} + \frac{\Delta F}{2}$$

$$\Delta F_l = f_{p1} - f_{s1}, \qquad \Delta F_h = f_{s2} - f_{p2} \tag{7.67}$$

$$\Delta F = \min\,[\Delta F_l, \Delta F_h]$$

Bandstop FIR Filter

$$h_d(n) = \begin{cases} \dfrac{1}{n\pi}\left[\sin(2\pi n f_{c_1}/F) - \sin(2\pi n f_{c_2}/F)\right], & \text{for } n > 0 \\ \dfrac{2}{F}(f_{c_1} - f_{c_2}) + 1, & \text{for } n = 0 \end{cases} \tag{7.68}$$

where

$$f_{c_1} = f_{p_1} + \frac{\Delta F}{2}, \qquad f_{c_2} = f_{p_2} - \frac{\Delta F}{2}$$

$$\Delta F_l = f_{s1} - f_{p1}, \qquad \Delta F_h = f_{p2} - f_{s2} \tag{7.69}$$

$$\Delta F = \min[\Delta F_l, \Delta F_h]$$

Example 7.26 *Design a low-pass digital FIR filter using Kaiser window satisfying the specifications given below.*

Passband cut-off frequency, f_p = 150 Hz, stopband cut-off frequency, f_s = 250 Hz, passband ripple, A_p = 0.1 dB, stopband attenuation, A_s = 40 dB and sampling frequency, F = 1000 Hz.

Solution A computer program can be written for the design of Kaiser window digital filter using the functions given in Appendix. The computer output is given below.

From Eq. (7.53), $\delta = 0.005756$.

The actual stopband attenuation, A_s = 44.796982 (from Eq. 7.50)

The value of α = 3.952357 (from Eq. (7.55)) and D = 2.565946 (from Eq. (7.56)).

The length of the filter, M = 27 (from Eq. 7.57).

The desired filter coefficients $\{h_d(n)\}$ are obtained from Eq. 7.63. The filter coefficients of the non-causal digital filter $\{h(n) = h(-n)\}$ along with the Kaiser window coefficients are listed.

$h_d(0)$ = 0.400000006	$a(0)$ = 1.000000000	$h(0)$ = 0.400000006
$h_d(1)$ = 0.302658588	$a(1)$ = 0.990451336	$h(1)$ = 0.299768597
$h_d(2)$ = 0.093381047	$a(2)$ = 0.962219954	$h(2)$ = 0.089853108
$h_d(3)$ = −0.062470987	$a(3)$ = 0.916526139	$h(3)$ = −0.057256293
$h_d(4)$ = −0.075602338	$a(4)$ = 0.855326056	$h(4)$ = −0.064664647
$h_d(5)$ = 0.000160935	$a(5)$ = 0.781202257	$h(5)$ = 0.000125723
$h_d(6)$ = 0.050484315	$a(6)$ = 0.697217405	$h(6)$ = 0.035198543
$h_d(7)$ = 0.026587145	$a(7)$ = 0.606746852	$h(7)$ = 0.016131667
$h_d(8)$ = −0.023507830	$a(8)$ = 0.513293743	$h(8)$ = −0.012066422
$h_d(9)$ = −0.033573132	$a(9)$ = 0.420304537	$h(9)$ = −0.014110940
$h_d(10)$ = 0.000160934	$a(10)$ = 0.330991328	$h(10)$ = 0.000053268
$h_d(11)$ = 0.027559206	$a(11)$ = 0.248175934	$h(11)$ = 0.006839531
$h_d(12)$ = 0.015454730	$a(12)$ = 0.174161583	$h(12)$ = 0.002691620
$h_d(13)$ = −0.014516240	$a(13)$ = 0.110641472	$h(13)$ = −0.00160609

Example 7.27 *Design a high-pass digital FIR filter using Kaiser window satisfying the specifications given below.*

Passband cut-off frequency, f_p = 3200 Hz, *stopband cut-off frequency, f_s* = 1600 Hz, *passband ripple, A_p* = 0.1 dB, *stopband attenuation, A_s* = 40 dB *and sampling frequency, F* = 10000 Hz.

Solution From Eq. (7.53), the value of $\delta = 0.005756$.

The actual stopband attenuation, A_s = 44.796982 dB (from Eq. (7.50))

The value of $\alpha = 3.952357$ (from Eq. (7.55)) and $D = 2.565946$ (from Eq. (7.56)).

The length of the filter, $M = 18$ (from Eq. (7.57)).

The desired filter coefficients $\{h_d(n)\}$ are obtained from Eq. (7.65). The filter coefficients of the non-causal digital filter $\{h(n) = h(-n)\}$ along with the Kaiser window coefficients are listed as follows.

$h_d(0) = 0.519999981$	$a(0) = 1.000000000$	$h(0) = 0.519999981$
$h_d(1) = -0.317566037$	$a(1) = 0.976742625$	$h(1) = -0.310180277$
$h_d(2) = -0.019747764$	$a(2) = 0.909421921$	$h(2) = -0.017959049$
$h_d(3) = 0.104217999$	$a(3) = 0.805053890$	$h(3) = 0.083901107$
$h_d(4) = 0.019595038$	$a(4) = 0.674255788$	$h(4) = 0.013212068$
$h_d(5) = -0.060581177$	$a(5) = 0.529811919$	$h(5) = -0.032096628$
$h_d(6) = -0.019342067$	$a(6) = 0.384986669$	$h(6) = -0.007446438$
$h_d(7) = 0.041210357$	$a(7) = 0.251848936$	$h(7) = 0.010378785$
$h_d(8) = 0.018991198$	$a(8) = 0.139850453$	$h(8) = 0.002655928$

7.4.6 FIR Half-band Digital Filters

A low-pass filter with the following frequency response has a desired filter response as shown in Fig. 7.5.

$$H(e^{j\omega T}) = \begin{cases} 1, & \text{for } 0 < f < f_p \\ 0, & \text{for } \dfrac{F}{2} - f_p < f < \dfrac{F}{2} \end{cases} \tag{7.70}$$

Such filters, exhibiting symmetry as shown in Fig. 7.5 are known as ***half-band filters***. These filters have an odd numbered filter length and are used in applications like decimation or interpolation of the sampling rate by a factor of two. It can be seen from Fig. 7.5 that the frequency response has an odd symmetry around $F/4$. This results in zero even filter coefficients. Consequently, the realisation of a half-band filter requires only half as many multiplies as general FIR filters.

The general properties of half-band FIR filters are listed below.

(i) Half-band FIR filters have ripples of equal amplitude in both the passband and the stopband.

$$\delta_p = \delta_s = \delta$$

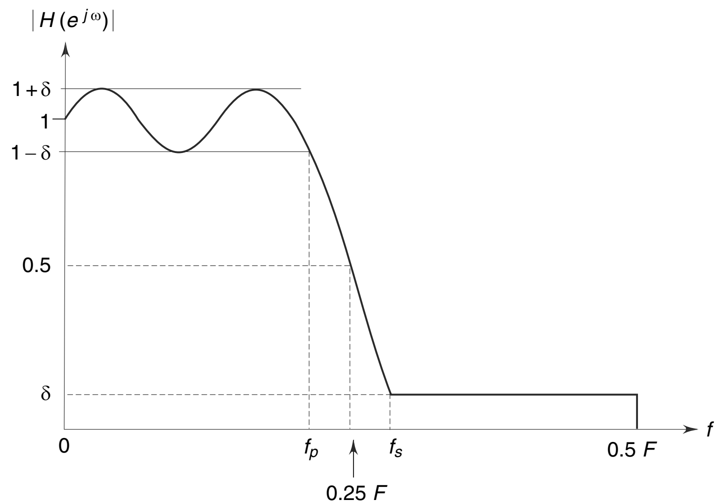


Fig. 7.5 *Frequency Response of Half-Band Filters*

(ii) The stopband frequency is

$$f_s = 0.5F - f_p$$

(iii) The transition bandwidth of the filter is then given by

$$\Delta F = 0.5\,F - 2f_p$$

(iv) The impulse-response at $n = 0$ is 0.5, i.e.

$$h(0) = \frac{1}{2}$$

(v) The frequency response is antisymmetric around the center of the band, i.e. at $f = 0.25F$.

$$|H(0.25F + f)| = 1 - |H(0.25F - f)|$$

At $f = 0.25\,F$,

$$|H(e^{j\omega T})| = |H(e^{j\pi/2})| = \frac{1}{2}$$

Table 7.1 *Comparison of Different Windows*

Sl. No.	Rectangular Window	Hanning Window	Hamming Window	Blackman Window	Kaiser Window
1	The width of the main lobe is $4\pi/M$.	The width of the main lobe is $8\pi/M$.	The width of the main lobe is $8\pi/M$.	The width of the main lobe in window spectrum is $12\pi/M$.	The width of the main lobe in window spectrum depends on α and M.
2	The maximum magnitude of the side lobe in window spectrum is –13 dB.	The maximum magnitude of the side lobe in window spectrum is –31 dB.	The maximum magnitude of the side lobe in window spectrum is –41dB.	The maximum magnitude of the side lobe in window spectrum is –58 dB.	The maximum magnitude of the side lobe with respect to peak of main lobe is variable using the parameter α.
3	The magnitude of the side lobe slightly decreases with increasing ω.	The magnitude of the side lobe decreases with increase in ω.	The magnitude of the side lobe remains constant. Here, the increased side-lobe attenuation is achieved at the expense of constant attenuation at high frequencies.	The magnitude of the side lobe decreases rapidly with increasing ω.	The magnitude of the side lobe decreases with increasing ω.
4	The minimum stopband attenuation is 22 dB.	The minimum stopband attenuation is 44 dB.	The minimum stopband attenuation is 51 dB.	The minimum stopband attenuation is 78 dB.	The minimum stopband attenuation is variable and depends on the value of α.

HILBERT TRANSFORMER 7.5

The frequency response of an ideal digital Hilbert transformer over one period is defined as

$$H_d(\omega) = H_d\left(e^{j\omega}\right) = \begin{cases} -j, & \text{for } 0 < \omega < \pi \\ +j, & \text{for } -\pi < \omega < 0 \end{cases}$$

which is shown in Fig. 7.6.

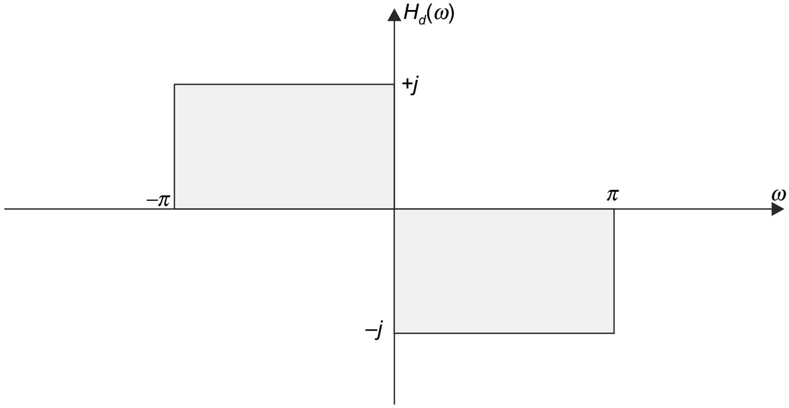


Fig. 7.6 *Frequency response of an ideal Hilbert transformer*

The frequency responses of an ideal Hilbert transformer and an ideal low-pass filter are similar, each having discontinuities separated by π. Here, $|H_d(\omega)| = 1$ for all frequencies and has a –90° phase-shift for $0 < \omega < \pi$ and a + 90° phase-shift for $-\pi < \omega < 0$. As a result, an ideal Hilbert transformer is also called a 90° phase-shifter.

The impulse response $h_d(n)$ of an ideal Hilbert transformer is found by taking inverse DTFT of $H_d(\omega)$.

$$h_d(n) = \frac{1}{2\pi}\left[\int_{-\pi}^{0} j\,e^{j\omega n}\,d\omega + \int_{0}^{\pi} -j\,e^{j\omega n}\,d\omega\right]$$

$$= \begin{cases} 0 & ;\ n = 0 \\ \dfrac{1}{\pi n}\left[1-(-1)^n\right] & ;\ n \neq 0 \end{cases}$$

or

$$h_d(n) = \begin{cases} \dfrac{2}{\pi}\dfrac{\sin^2\left(\dfrac{\pi n}{2}\right)}{n} & ;\ n \neq 0 \\ 0 & ;\ n = 0 \end{cases}$$

Here, $h_d(n)$ is infinite in duration and noncausal. The implication of $h_d(n)$ being a two-sided infinite-length impulse response is that the ideal Hilbert transformer is an unrealisable system. Also, since $h_d(n) = -h_d(-n)$, $h_d(n)$ is antisymmetric. Here, the design of linear-phase FIR Hilbert transformers is focused with an antisymmetric impulse response: $h(n) = -h(M-1-n)$.

By translating $h_d(n)$ to right by an amount $\alpha = \dfrac{M-1}{2}$, we will have antisymmetry about $n = \alpha$. That is, the impulse response of an ideal Hilbert transformer that is antisymmetric about $n = \alpha$ is

$$h'_d(n) = h_d(n-\alpha)$$

$$= \begin{cases} \dfrac{2}{\pi}\dfrac{\sin^2\left[\dfrac{\pi}{2}(n-\alpha)\right]}{n-\alpha} & ;\ n \neq \alpha \\ 0 & ;\ n = \alpha \end{cases}$$

The finite impulse response $h(n)$ of a Hilbert transformer will have linear phase if its impulse response exhibits either symmetry or antisymmetry about the midpoint, $n = \frac{M-1}{2}$. Here, $h(n)$ is designed to have antisymmetry since $h'_d(n)$ is antisymmetric about $n = \alpha$.

The finite impulse response $h(n)$ of a Hilbert transformer is obtained by truncating $h'_d(n)$ by a causal window that is symmetric about $n = \alpha$. That is,

$$h(n) = h'_d(n)w(n), \quad 0 \le n \le M-1$$

$$h(n) = \begin{cases} \dfrac{2}{\pi}\dfrac{\sin^2\left[\dfrac{\pi}{2}(n-\alpha)\right]}{n-\alpha} \times w(n)\ ; 0 \le n \le M-1,\ n \ne \alpha \\ 0 \qquad ; n = \alpha \end{cases}$$

Since $h(n)$ is antisymmetric about $n = \alpha$ and is zero at $n = \alpha$, it implies that M has to be an odd integer only.

The magnitude response for M odd and $h(n) = -h(M-1-n)$ is

$$|H(\omega)| = |H_r(\omega)|$$

$$= \left| 2 \sum_{n=0}^{\left(\frac{M-3}{2}\right)} h(n)\ \sin\left[\omega\left(\left(\frac{M-1}{2}\right) - n\right)\right] \right|$$

The Hilbert transformer can also be designed using the frequency-sampling method.

The Hilbert transformer is generally used to generate the single-sideband modulated signals, radar signal processing and speech signal processing.

Example 7.28 *Design an ideal Hilbert transformer having frequency response*

$$H(e^{j\omega}) = \begin{cases} +j, & -\pi \le \omega \le 0 \\ -j, & 0 \le \omega \le \pi \end{cases}$$

using (a) Rectangular window, and (b) Blackman window with M = 11.

Solution Given $H\left(e^{j\omega}\right) = \begin{cases} j, & -\pi \le \omega \le 0 \\ -j, & 0 \le \omega \le \pi \end{cases}$

(a) Rectangular Window

$$h_d(n) = \frac{1}{2\pi}\left[\int_{-\pi}^{0} je^{j\omega n} d\omega + \int_{0}^{\pi} -je^{j\omega n} d\omega\right]$$

$$= \frac{j}{2\pi}\left[\left[\frac{e^{j\omega n}}{jn}\right]_{-\pi}^{0} - \left[\frac{e^{j\omega n}}{jn}\right]_{0}^{\pi}\right] = \frac{1}{2\pi}\frac{1}{jn}\left[e^0 - e^{-j\pi n} - e^{j\pi n} + e^0\right]$$

$$= \frac{1}{2\pi n}\left[2 - e^{j\pi n} + e^{-j\pi n}\right] = \frac{1}{2\pi n}\left[2 - 2\cos\pi n\right]$$

$$= \frac{1-\cos\pi n}{\pi n} = \begin{cases} \dfrac{2\sin^2\left(\dfrac{\pi n}{2}\right)}{\pi n}, & n \ne 0 \\ 0, & n = 0 \end{cases}$$

$$h_d(n) = \frac{1-\cos \pi n}{\pi n}$$

Therefore, $h_d(n)$ is antisymmetry.

$$h_d(0) = \frac{1-\cos 0}{\pi 0} = 0$$

$$h_d(1) = \frac{1-\cos \pi}{\pi} = \frac{2}{\pi} = h_d(-1)$$

$$h_d(2) = 0 = -h_d(-2)$$

$$h_d(3) = \frac{2}{3\pi} = -h_d(-3)$$

$$h_d(4) = 0 = -h_d(-4)$$

$$h_d(5) = \frac{2}{5\pi} = -h_d(-5)$$

We know that

$h(n) = h_d(n)\, w_R(n)$, where $w_R(n) = 1$

$h(n) = h_d(n)$

$h(0) = 0$

$$h(1) = -h(-1) = \frac{2}{\pi}$$

$h(2) = -h(-2) = 0$

$$h(3) = -h(-3) = \frac{2}{3\pi}$$

$h(4) = -h(-4) = 0$

$$h(5) = -h(-5) = \frac{2}{5\pi}$$

$$H(z) = z^{-5} \sum_{n=-5}^{5} h(n) z^{-n}$$

$$= z^{-5}\left[\frac{2}{\pi}\left[z - z^{-1}\right] + \frac{2}{3\pi}\left[z^3 - z^{-3}\right] + \frac{2}{5\pi}\left[z^5 - z^{-5}\right]\right]$$

$$= \frac{2}{\pi}\left[z^{-4} - z^{-6}\right] + \frac{2}{3\pi}\left[z^{-2} - z^{-8}\right] + \frac{2}{5\pi}\left[1 - z^{-10}\right]$$

(b) Blackman Window

$$w_B(n) = 0.42 + 0.5\cos\frac{\pi n}{5} + 0.08\cos\frac{2\pi n}{5} \qquad \text{for } -5 \le n \le 5$$

$w_B(0) = 1$

$w_B(1) = w_B(-1) = 0.849$

$w_B(2) = w_B(-2) = 0.509$

$w_B(3) = w_B(-3) = 0.2$

$w_B(4) = w_B(-4) = 0.4$

$w_B(5) = w_B(-5) = 0$

We know that $h(n) = h_d(n)w_B(n)$

Therefore, $h(0) = 0$

$$h(1) = -h(-1) = \left(\frac{2}{\pi}\right)0.849 = 0.5405$$

$$h(2) = -h(-2) = 0$$

$$h(3) = -h(-3) = \left(\frac{2}{3\pi}\right)0.2 = 0.0423$$

$$h(4) = -h(-4) = 0$$

$$h(5) = -h(-5) = \left(\frac{2}{5\pi}\right) = 0$$

Hence, $H(z) = z^{-5}\sum_{n=5}^{5} h(n)z^{-n}$

$$= z^{-5}\,[0.54\,(z - z^{-1}) + 0.0424\,(z^3 - z^{-3})]$$
$$= 0.54\,(z^{-4} - z^{-6}) + [0.0424\,(z^{-2} - z^{-8})]$$

DESIGN OF OPTIMAL LINEAR PHASE FIR FILTERS 7.6

In both frequency sampling and windowing methods of designing FIR filters, there was a problem with the precise control of the critical frequencies. In the optimal design method, to be discussed in this section, it is considered as a Chebyshev approximation problem. It is viewed that the weighted approximation error between the actual frequency response and the desired filter response is spread across the passband and the stopband and the maximum error is minimised. This design method results in passband and the stopband having ripples. The design procedure is explained using a low-pass filter with passband and stopband edge frequencies ω_p and ω_s, respectively. From Fig. 7.7, the frequency response of the filter in the passband is

$$1-\delta_p \le |H(e^{j\omega})| \le 1 + \delta_p,\ |\omega| \le \omega_p \tag{7.71}$$

The frequency response in the stopband is

$$-\delta_s \le |H(e^{j\omega})| \le \delta_s,\ |\omega| \ge \omega_s \tag{7.72}$$

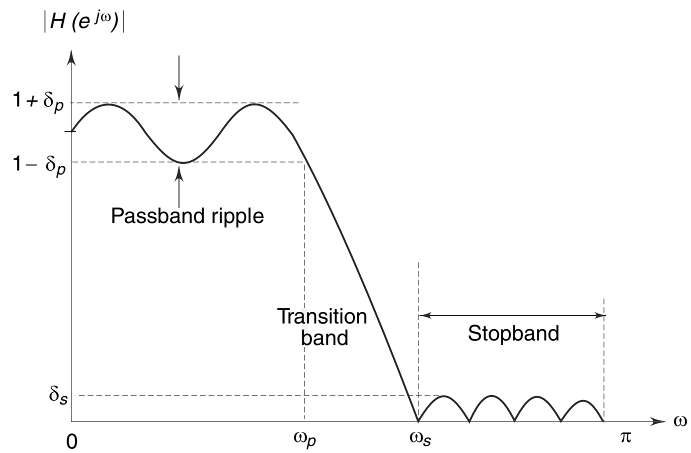


Fig. 7.7 *Frequency Response Characteristics of Physically Realisable Filters*

The term δ_p represents the passband ripple and δ_s is the maximum attenuation in the stopband.

There are four different cases that result in a linear phase FIR filter, viz., (i) symmetric unit impulse response and the length of the filter, M odd, (ii) symmetric unit impulse response and M even, (iii) anti-symmetric unit impulse response and M odd, and (iv) antisymmetric unit impulse response and M even. The first case is discussed below in detail and other cases are listed in Table 7.2.

In the symmetric unit impulse response case, $h(n) = H(M - 1- n)$. The real-valued frequency response characteristics $|H(e^{j\omega})| = |H_r(e^{j\omega})|$, given in Eq. (7.14), is

$$|H(e^{j\omega})| = h\left(\frac{M-1}{2}\right) + 2\sum_{n=0}^{\frac{(M-3)}{2}} h(n)\cos\omega\left(\frac{M-1}{2} - n\right) \tag{7.73}$$

Let $k = (M - 1)/2 - n$. Then Eq. (7.73) can be written as

$$|H(e^{j\omega})| = \sum_{k=0}^{\frac{(M-1)}{2}} a(k)\cos\omega k \tag{7.74}$$

where

$$a(0) = h\left(\frac{M-1}{2}\right)$$

$$a(k) = 2h\left(\frac{M-1}{2} - k\right) \text{ for } 1 \le k \le \frac{M-1}{2}$$

The magnitude response for the other cases are similarly converted to a compact form as given in Table 7.2.

From Table 7.2, it can be seen that the magnitude response function can be written as given in Eq. (7.75), for the four different cases.

$$|H(e^{j\omega})| = Q(\omega)\, P(\omega) \tag{7.75}$$

where

$$Q(\omega) = \begin{cases} 1 & \text{case (i)} \\ \cos\dfrac{\omega}{2} & \text{case (ii)} \\ \sin\omega & \text{case (iii)} \\ \sin\dfrac{\omega}{2} & \text{case (iv)} \end{cases} \tag{7.76}$$

and $P(\omega)$ is of the common form

$$P(\omega) = \sum_{k=0}^{L} \tilde{a}(k)\cos\omega k \tag{7.77}$$

$\{\tilde{a}(k)\}$ are the filter parameters and these are linearly related to the unit-impulse response $h(n)$ of the filter. The upper limit L changes from case to case. In the design of optimal filters, the desired frequency response $H_d(\omega)$ and the weighting function $W(\omega)$ on the approximation error are also defined. The desired frequency response is defined to be 1 in the passband and 0 in the stopband. The weighting function helps in hoosing the relative size of the errors in the frequency bands. The weighting function is usually normalised to unity in the stopband and $W(\omega) = (\delta_s / \delta_p)$ in the passband.

$$W(\omega) = \begin{cases} \delta_s / \delta_p, & \text{passband} \\ 1, & \text{stopband} \end{cases} \tag{7.78}$$

Table 7.2 *Magnitude Response Functions for Linear Phase FIR Filters*

Filter Type	$Q(\omega)$	$P(\omega)$
Case (i) - Symmetric and *M* odd $h(n) = h(M-1-n)$	1	$\sum_{k=0}^{(M-1)/2} a(k)\cos\omega k$
Case (ii) - Symmetric and *M* even $h(n) = h(M-1-n)$	$\cos\frac{\omega}{2}$	$\sum_{k=0}^{(M/2)-1} b(k)\cos\omega k$
Case (iii) - Antisymmetric and *M* odd $h(n) = -h(M-1-n)$	$\sin\omega$	$\sum_{k=0}^{(M-3)/2} c(k)\cos\omega k$
Case (iv) - Antisymmetric and *M* even $h(n) = -h(M-1-n)$	$\sin\frac{\omega}{2}$	$\sum_{k=0}^{(M/2)-1} d(k)\cos\omega k$

The weighted approximation error is now defined as

$$\begin{aligned} E(\omega) &= W(\omega)[H_d(\omega) - H(e^{j\omega})] \\ &= W(\omega)[H_d(\omega) - Q(\omega)p(\omega)] \\ &= W(\omega)Q(\omega)\left[\frac{H_d(\omega)}{Q(\omega)} - P(\omega)\right] \end{aligned} \tag{7.79}$$

Let us define the modified weighting function $\hat{W}(\omega)$ and a modified desired frequency response $\hat{H}_d(\omega)$ as shown below.

$$\hat{W}(\omega) = W(\omega)Q(\omega) \text{ and } \hat{H}_d(\omega) = \frac{H_d(\omega)}{Q(\omega)} \tag{7.80}$$

The approximation error is then

$$E(\omega) = \hat{W}(\omega)[\hat{H}_d(\omega) - P(\omega)] \tag{7.81}$$

The above expression for the approximation error is valid for all four types of linear phase FIR filters as given in Table 7.2. Once the error function is given, the filter parameters $\{\tilde{a}(k)\}$ are determined such that the maximum absolute value of $E(\omega)$ is minimised. Mathematically, this is equivalent to seeking the solution to the problem

$$\min_{[\tilde{a}(k)]}\left[\max_{\omega\in S} |E(\omega)|\right] = \min_{[\tilde{a}(k)]}\left[\max_{\omega\in S}\left|\hat{W}(\omega)[\hat{H}_d(\omega) - \sum_{k=0}^{L}\tilde{a}(k)\cos\omega k]\right|\right] \tag{7.82}$$

where S is the set of frequency bands over which the optimisation is to be performed. Parks and McClellan applied the alternation theorem in the Chebyshev approximation and obtained the solution to the problem specified in Eq. (7.82). The ***alternation theorem*** is stated below. Let S be a compact subset of the interval $[0, \pi]$. A necessary and sufficient condition for

$$P(\omega) = \sum_{k=0}^{L}\tilde{a}(k)\cos\omega k$$

to be the unique, best weighted Chebyshev approximation to $\hat{H}_d(\omega)$ in S is that the error function $E(\omega)$ exhibit at least $L + 2$ extremal frequencies in S. That is, there must exist at least $L + 2$ frequencies $\{\omega_i\}$ in S such that $\omega_1 < \omega_2 < ..., \omega_{L+2}$, $E(\omega_i) = -E(\omega_{i+1})$, and

$$|E(\omega_i)| = \max_{\omega \in S} |E(\omega)|, \quad i = 1, 2, \ldots, L+2$$

The error function $E(\omega)$ alternates in sign between two successive extremal frequencies. Hence the theorem is called the alternation theorem. The filter designs containing more than $L + 2$ extremal frequencies are called ***extra ripple filters*** and when the filter design contains the maximum number of alternations, it is called a ***maximal ripple filter***. The alternation theorem guarantees a unique solution for the approximation problem and for a given set of extremal frequencies $\{\omega_n\}$ the error function may be written as

$$E(\omega_n) = (-1)^n\,\delta = \hat{W}(\omega)[\hat{H}_d(\omega) - P(\omega) - P(\omega)], \; n = 0, 1, \ldots, L+1 \tag{7.83}$$

where δ represents the maximum value of the error function $E(\omega)$. Equation (7.83) can be put in matrix form as

$$\begin{bmatrix} 1 & \cos\omega_0 & \cos 2\omega_0 & \cdots & \cos L\omega_0 & \dfrac{1}{\hat{W}(\omega_0)} \\ 1 & \cos\omega_1 & \cos 2\omega_1 & \cdots & \cos L\omega_1 & \dfrac{-1}{\hat{W}(\omega_1)} \\ . & . & . & \cdots & . & . \\ . & . & . & \cdots & . & . \\ . & . & . & \cdots & . & . \\ 1 & \cos\omega_{L+1} & \cos 2\omega_{L+1} & \cdots & \cos L\omega_{L+1} & \dfrac{(-1)^{L+1}}{\hat{W}(\omega_{L+1})} \end{bmatrix} \begin{bmatrix} \tilde{a}(0) \\ \tilde{a}(1) \\ . \\ . \\ . \\ \tilde{a}(L) \\ \delta \end{bmatrix} = \begin{bmatrix} \hat{H}_d(\omega_0) \\ \hat{H}_d(\omega_1) \\ . \\ . \\ . \\ \hat{H}_d(\omega_{L+1}) \end{bmatrix} \tag{7.84}$$

Therefore, if the extremal frequencies are known, the coefficients $\{h(n)\}$, the peak error δ, and hence the frequency response of the filter, can be computed by inverting the matrix. As matrix inversion is time consuming and inefficient, the peak error δ can be computed using the Remez exchange algorithm. In this algorithm, a set of extremal frequencies is first assumed, the values of $P(\omega)$ and δ are determined and then the error function $E(\omega)$ is computed. This error function is then used to determine another set of $L + 2$ extremal frequencies and the iterative process is repeated until the error function converges to the optimal set of extremal frequencies. A flow chart of the Remez exchange algorithm is shown in Fig. 7.8. A computer-aided iterative procedure for designing an optimal FIR filter has been developed by Parks and McClellan.

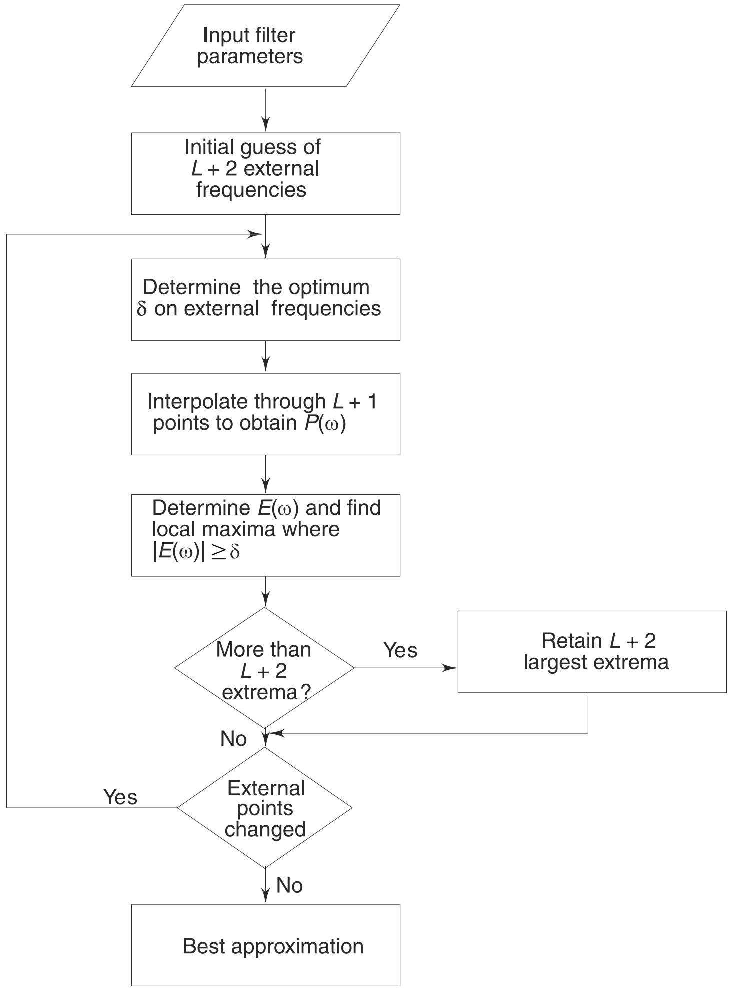


Fig. 7.8 *Flow chart of the Remez Exchange Algorithm*

REVIEW QUESTIONS

7.1 What is an FIR system? Compare an FIR system with an IIR system.

7.2 Define phase delay and group delay.

7.3 What is a linear phase filter? What conditions are to be satisfied by the impulse response of an FIR system in order to have a linear phase?

7.4 The length of an FIR filter is 13. If the filter has a linear phase, show that

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

7.5 The transfer function of an FIR filter ($M = 7$) is $H(z) = \sum_{n=0}^{M-1} h(n)z^{-n}$. Determine the magnitude response and show that the phase and group delays are constant.

7.6 Obtain a general expression for the frequency response of linear phase FIR filters.

7.7 What are the different design techniques available for the FIR filters?

7.8 Explain the Fourier series method of designing an FIR filter.

7.9 Use the Fourier series method to design a high-pass digital filter to approximate the ideal specifications given by

$$H(e^{j\omega}) = \begin{cases} 0, & \text{for} |f| < f_p \\ 1, & f_p \le |f| \le F/2 \end{cases}$$

where f_p is the passband frequency and F is the sampling frequency.

7.10 Illustrate the steps involved in the design of linear phase FIR filter by the frequency sampling method.

7.11 Explain the Type-I Frequency sampling method of designing an FIR filter.

7.12 A low-pass filter has the desired response as shown

$$H_d(e^{j\omega}) = \begin{cases} e^{-j8\omega}, & 0 \le \omega \le \pi/2 \\ 1, & \pi/2 < |\omega| \le \pi \end{cases}$$

Using the frequency sampling (Type-I) technique, determine the filter coefficients. The length of the filter is $M = 17$.

Ans: $h(n) = 0.05 + 0.1176$ {cos $0.3696(8-n)$ + cos $0.7392(8-n)$ + cos $1.1088(8-n)$ + cos $1.478\,(8-n)$}

7.13 Explain the Type-II Frequency sampling method of designing an FIR filter.

7.14 What are the effects of truncating an infinite Fourier series into a finite series?

7.15 Explain Gibb's phenomenon.

7.16 What are the desirable features of the window functions?

7.17 What are the effects of windowing?

7.18 Explain the process of windowing using illustrations.

7.19 Name the different types of window functions. How they are defined?

7.20 What is a rectangular window function? Obtain its frequency-domain characteristics.

7.21 What is a Hamming window function? Obtain its frequency-domain characteristics.

7.22 What is a Hanning window function? Obtain its frequency-domain characteristics.

7.23 Compare the frequency-domain characteristics of the different types of window functions.

7.24 Compare the rectangular window and Hanning window.

7.25 Compare the Hamming window and Blackman window.

7.26 List the three well known methods of design techniques for FIR filters and explain any one.

7.27 The desired frequency response of a low-pass filter is

$$H_d(e^{j\omega}) = \begin{cases} 1, & -\pi/2 \le \omega \le \pi/2 \\ 0, & \pi/2 \le |\omega| < \pi \end{cases}$$

Determine $h_d(n)$. Also determine $h(n)$ using the symmetric rectangular window with window length = 7.

Ans: $h_d(-3) = -0.1061 = h_d(3), h_d(-2) = 0 = h_d(2), h_d(-1) = 0.3183 = h_d(1), h_d(0) = 0.5; h(n) = h_d(n) \cdot w(n) = h_d(n)$ for $-3 \le n \le 3$

7.28 The desired frequency response of a low-pass filter is

$$H_d(e^{j\omega}) = \begin{cases} e^{-j3\omega}, & -3\pi/4 \le \omega \le 3\pi/4 \\ 0, & 3\pi/4 < |\omega| \le \pi \end{cases}$$

Determine $H(e^{j\omega})$ *for* $M = 7$ using a rectangular window.

Ans: $H(e^{j\omega}) = e^{-j3\omega}[0.75 + 0.4502 \cos \omega - 0.3184 \cos 2\omega + 0.15 \cos 3\omega]$

7.29 Design a bandpass filter which approximates the ideal filter with cut-off frequencies at 0.2 rad/sec and 0.3 rad/sec. The filter order is $M = 7$. Use the Hanning window function.

Ans: $h(0) = 0$, $h(1) = 0.0078$, $h(2) = 0.0209$, $h(3) = 0.0232$, $h(4) = 0.0128$, $h(5) = 0.00248$, $h(6) = 0$.

7.30 What is a Kaiser window? Why is it superior to other window functions?

7.31 Explain the procedure for designing an FIR filter using the Kaiser window.

7.32 The desired frequency response of a desired filter is

$$H_d(\omega) = \begin{cases} e^{-j3\omega}, & \dfrac{-\pi}{4} \le \omega \le \dfrac{\pi}{4} \\ 0, & \dfrac{\pi}{4} \le |\omega| \le \pi \end{cases}$$

Determine the filter coefficients if the window function is defined as

$$w(n) = \begin{cases} 1, & 0 \le n \le 5 \\ 0, & \text{otherwise} \end{cases}$$

7.33 Determine the filter coefficients $h(n)$ of length $M = 15$ obtained by sampling its frequency response as

$$H\left[\left(\frac{2\pi}{15}\right)k\right] = \begin{cases} 1, & k = 0, 1, 2, 3, 4 \\ 0.4, & k = 5 \\ 0, & k = 6, 7 \end{cases}$$

using rectangular window.

7.34 Design a digital filter with

$$H_d\left(e^{j\omega}\right) = \begin{cases} 1, & 2 \le |\omega| \le \pi \\ 0, & \text{otherwise} \end{cases}$$

7.35 A bandpass FIR filter of length 7 is required. It is to have lower and upper cut-off frequencies of 3 kHz and 5 kHz respectively and is intended to be used with a sampling frequency of 24 kHz. Determine the filter coefficients using Hanning window. Consider the filter to be causal.
Using Hamming window with $M = 7$, draw the frequency response.

7.36 Explain the Hilbert transformer in detail.

Chapter 8 Infinite Impulse Response (IIR) Filters

INTRODUCTION 8.1

There are several techniques available for the design of digital filters having an infinite duration unit impulse response. The analog filter design is well-developed and the techniques discussed in this chapter are all based on taking an analog filter and converting it into a digital filter. Thus, the design of an IIR filter involves design of a digital filter in the analog domain and transforming the design into the digital domain.

The system function describing an analog filter may be written as

$$H_a(s) = \frac{\sum_{k=0}^{M} b_k s^k}{\sum_{k=0}^{N} a_k s^k} \tag{8.1}$$

where $\{a_k\}$ and $\{b_k\}$ are the filter coefficients. The impulse response of these filter coefficients is related to $H_a(s)$ by the Laplace transform

$$H_a(s) = \int_{-\infty}^{\infty} h(t) e^{-st} \mathrm{d}t \tag{8.2}$$

The analog filter having the rational system function $H_a(s)$ given in Eq. (8.1) can also be described by the linear constant-coefficient differential equation

$$\sum_{k=0}^{N} a_k \frac{\mathrm{d}^k y(t)}{\mathrm{d}t^k} = \sum_{k=0}^{M} b_k \frac{\mathrm{d}^k x(t)}{\mathrm{d}t} \tag{8.3}$$

where $x(t)$ is the input signal and $y(t)$ is the output of the filter.

The above three equivalent characterisation of an analog filter leads to three alternative methods for transforming the filter into the digital domain. The design techniques for IIR filters are presented with the restriction that the filters be realisable and stable. Recall that an analog filter with system function $H(s)$ is stable if all its poles lie in the left-half of the s-plane. As a result, if the conversion techniques are to be effective, the technique should possess the following properties:

(i) The $j\Omega$ axis in the s-plane should map onto the unit circle in the z-plane. This gives a direct relationship between the two frequency variables in the two domains.

(ii) The left-half plane of the s-plane should map into the inside of the unit circle in the z-plane to convert a stable analog filter into a stable digital filter.

The physically realisable and stable IIR filter cannot have a linear phase. For a filter to have a linear phase, the condition is $h(n) = h(M-1-n)$ and the filter would have a mirror image pole outside the unit circle for every pole inside the unit circle. This results in an unstable filter. As a result, a causal and stable IIR filter cannot have a linear phase. In the design of IIR filters, only the desired magnitude response is specified and the phase response that is obtained from the design methodology is accepted.

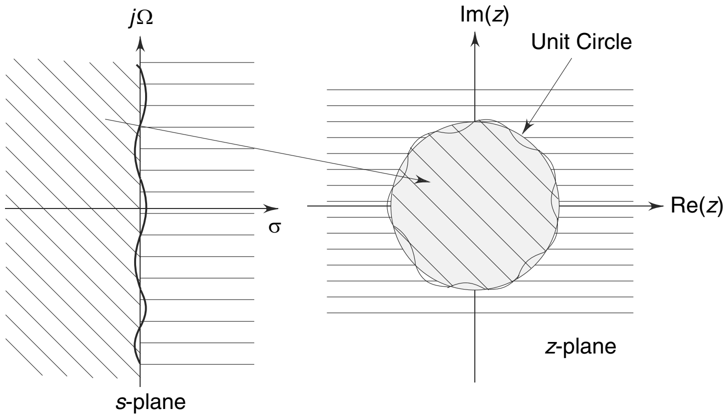


Fig. 8.1 *Mapping from s-plane to z-plane*

IIR FILTER DESIGN BY APPROXIMATION OF DERIVATIVES 8.2

In this method, an analog filter is converted into a digital filter by approximating the differential equation in Eq. (8.3) by an equivalent difference equation. The backward difference formula is substituted for the derivative d$y(t)$/dt at time $t = nT$. Thus,

$$\left.\frac{\mathrm{d}y(t)}{\mathrm{d}t}\right|_{t=nT} = \frac{y(nT)-y(nT-T)}{T}$$
$$= \frac{y(n)-y(n-1)}{T} \tag{8.4}$$

where T is the sampling interval and $y(n) \equiv y(nT)$. The system function of an analog differentiator with an output dy/dt is $H(s) = s$, and the digital system that produces the output $[y(n)-y(n-1)]/T$ has the system function $H(z) = (1-z^{-1})/T$. These two can be compared to get the frequency-domain equivalent for the relationship in Eq. (8.4) as

$$s = \frac{1-z^{-1}}{T} \tag{8.5}$$

The second derivative $\mathrm{d}^2y(t)/\mathrm{d}t^2$ is replaced by the second backward difference,

$$\left.\frac{\mathrm{d}^2y(t)}{\mathrm{d}t^2}\right|_{t=nT} = \frac{\mathrm{d}}{\mathrm{d}T}\left[\frac{\mathrm{d}y(t)}{\mathrm{d}t}\right]_{t=nT}$$
$$= \frac{\{[y(nT)-y(nT-T)]/T\}-\{[y(nT-T)-y(nT-2T]/T\}}{T}$$
$$= \frac{y(n)-2y(n-1)+y(n-2)}{T^2} \tag{8.6}$$

The equivalent to Eq. (8.6) in the frequency-domain is

$$s^2 = \frac{1-2z^{-1}+z^{-2}}{T^2} = \left(\frac{1-z^{-1}}{T}\right)^2 \tag{8.7}$$

The i^{th} derivative of $y(t)$ results in the equivalent frequency-domain relationship

$$s^i = \left(\frac{1-z^{-1}}{T}\right)^i \tag{8.8}$$

As a result, the digital filter's system function can be obtained by the method of approximation of the derivatives as,

$$H(z) = H_a(s)\Big|_{s=\frac{(1-z^{-1})}{T}} \tag{8.9}$$

where $H_a(s)$ is the system function of the analog filter characterised by the differential equation given in Eq. (8.3). The outcomes of the mapping of the z-plane from the s-plane are discussed below. Equation (8.5) can be, equivalently, written as

$$z = \frac{1}{1-sT} \tag{8.10}$$

Substituting $s = j\,\Omega$ in the above equation,

$$z = \frac{1}{1-j\Omega T} = \frac{1}{1+\Omega^2T^2} + j\frac{\Omega T}{1+\Omega^2T^2} \tag{8.11}$$

Varying Ω from $-\infty$ to ∞, the corresponding locus of points in the z-plane is a circle with radius $\frac{1}{2}$ and with centre at $z = \frac{1}{2}$, as shown in Fig. 8.2.

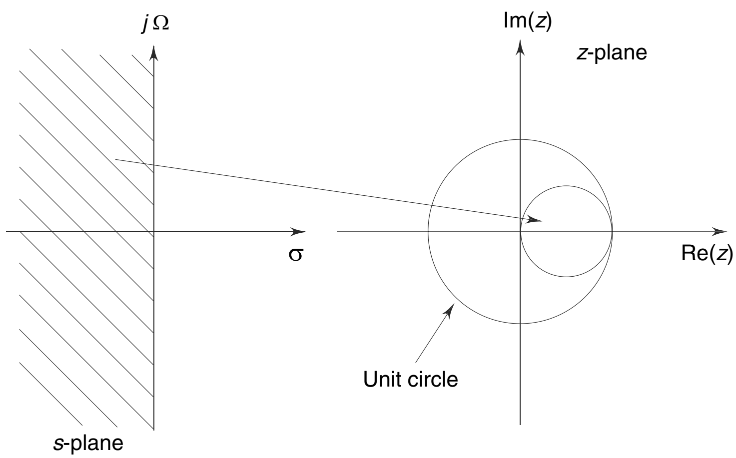


Fig. 8.2 *The Mapping of Eq. (8.5) into the z-plane*

It can be seen that the mapping of Eq. (8.5) takes the left-half plane of s-domain into the corresponding points inside the circle of radius 0.5 and centre at $z = 0.5$, and the right-half of the s-plane is mapped outside the unit circle. As a result, this mapping results in a stable analog filter transformed into a stable digital filter; however, as the locations of poles in the z-domain are confined to smaller frequencies, this design method can be used only for transforming analog low-pass filters and bandpass filters having smaller resonant frequencies. Neither a high-pass filter nor a band reject filter can be realised using this technique.

The forward difference can be substituted for the derivative instead of the backward difference. This gives,

$$\frac{\mathrm{d}y(t)}{\mathrm{d}t} = \frac{y(nT+T)-y(nT)}{T}$$
$$= \frac{y(n+1)-y(n)}{T} \tag{8.12}$$

The transformation formula will be

$$s = \frac{z-1}{T} \tag{8.13}$$

or,

$$z = 1 + sT \tag{8.14}$$

The mapping of Eq. (8.14) is shown in Fig. 8.3. This results in a worse situation than the backward difference substitution for the derivative. When $s = j\Omega$, the mapping of these points in the s-domain results in a straight line in the z-domain with coordinates $(z_{real}, z_{imag}) = (1, \Omega T)$. Consequently, stable analog filters do not always map into stable digital filters.

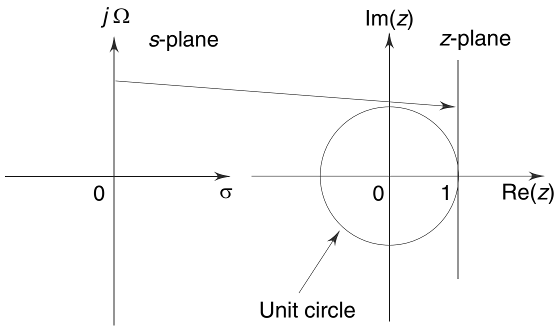


Fig. 8.3 *Mapping of Eq. (8.14) into the z-plane*

The limitations of the two mapping methods discussed above are overcome by using a more complex substitution for the derivatives. An Nth order difference is proposed for the derivative, as shown.

$$\left.\frac{\mathrm{d}y(t)}{\mathrm{d}t}\right|_{t=nT} = \frac{1}{T}\sum_{k=1}^{N} a_k \frac{y(nT+kT)-y(nT-kT)}{T} \tag{8.15}$$

where $\{a_k\}$ are a set of parameters selected so as to optimise the approximation. The transformation from the s-plane to the z-plane is then,

$$s = \frac{1}{T}\sum_{k=1}^{N} a_k (z^k - z^{-k}) \tag{8.16}$$

By choosing proper values for $\{a_k\}$, the $j\Omega$ axis can be mapped into the unit circle and the left-half s-plane can be mapped into points inside the unit circle in the z-plane.

Example 8.1 *Use the backward difference for the derivative to convert the analog low-pass filter with system function*

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

Solution The mapping formula for the backward difference for the derivative is given in Eq. (8.5), i.e.

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

The system response of the digital filter is

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

If $T = 1$ s,

$$H(z) = \frac{1}{3-z^{-1}}$$

Example 8.2 *Use the backward difference for the derivative and convert the analog filter with system function*

$$H(s) = \frac{1}{s^2+16}$$

Solution Using Eq. (8.5),

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

The system response of the digital filter is

$$H(z) = H(s)\Big|_{s=\frac{1-z^{-1}}{T}} = \frac{1}{\left(\frac{1-z^{-1}}{T}\right)^2+16}$$

$$H(z) = \frac{T^2}{1-2z^{-1}+z^{-2}+16T^2}$$

If $T = 1$ s,

$$H(z) = \frac{1}{z^{-2}-2z^{-1}+17}$$

Example 8.3 *An analog filter has the following system function. Convert this filter into a digital filter using backward difference for the derivative.*

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

Solution The system response of the digital filter is

$$H(z) = H(s)\Big|_{s=\frac{1-z^{-1}}{T}} = \frac{1}{\left(\frac{1-z^{-1}}{T}+0.1\right)^2+9}$$

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

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

If $T = 1$ s,

$$H(z)=\frac{0.0979}{1-0.2155z^{-1}+0.09792z^{-2}}$$

IIR FILTER DESIGN BY IMPULSE INVARIANT METHOD 8.3

In this technique, the desired impulse response of the digital filter is obtained by uniformly sampling the impulse response of the equivalent analog filter. That is,

$$h(n) = h_a(nT) \tag{8.17}$$

where T is the sampling interval. The transformation technique can be well understood by first considering a simple distinct pole case for the analog filter's system function, as shown below.

$$H_a(s)=\sum_{i=1}^{M}\frac{A_i}{s-p_i} \tag{8.18}$$

The impulse response of the system specified by Eq. (8.18) can be obtained by taking the inverse Laplace transform and it will be of the form

$$h_a(t)=\sum_{i=1}^{M}A_i e^{p_i t}u_a(t) \tag{8.19}$$

where $u_a(t)$ is the unit step function in continuous time. The impulse response $h(n)$ of the equivalent digital filter is obtained by uniformly sampling $h_a(t)$, i.e. by applying Eq. (8.17)

$$h(n)=h_a(nT)=\sum_{i=1}^{M}A_i e^{p_i nT}u_a(nT) \tag{8.20}$$

The system response of the digital system of Eq. (8.20), can be obtained by taking the z-transform, i.e.

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

Using Eq. (8.20),

$$H(z)=\sum_{n=0}^{\infty}\left[\sum_{i=1}^{M}A_i e^{p_i nT}u_a(nT)\right]z^{-n} \tag{8.21}$$

Interchanging the order of summation,

$$H(z)=\sum_{i=1}^{M}\left[\sum_{n=0}^{\infty}A_i e^{p_i nT}u_a(nT)\right]z^{-n}$$

$$H(z)=\sum_{i=1}^{M}\frac{A_i}{1-e^{p_i T}z^{-1}} \tag{8.22}$$

Now, by comparing Eqs. (8.18) and (8.22), the mapping formula for the impulse invariant transformation is given by

$$\frac{1}{s-p_i} \to \frac{1}{1-e^{p_i T} z^{-1}} \tag{8.23}$$

Equation (8.23) shows that the analog pole at $s = p_i$ is mapped into a digital pole at $z = e^{p_i T}$. Therefore, the analog poles and the digital poles are related by the relation

$$z = e^{sT} \tag{8.24}$$

The general characteristic of the mapping $z = e^{sT}$ can be obtained by substituting $s = \sigma + j\Omega$ and expressing the complex variable z in the polar form as $z = re^{j\omega}$. With these substitutions, Eq. (8.24) becomes

$$re^{j\omega} = e^{\sigma T} e^{j\Omega T}$$

Clearly,

$$r = e^{\sigma T}$$
$$\omega = \Omega T \tag{8.25}$$

Consequently, $\sigma < 0$ implies that $0 < r < 1$ and $\sigma > 0$ implies that $r > 1$. When $\sigma = 0$, we have $r = 1$. Therefore, the left-half of s-plane is mapped inside the unit circle in the z-plane and the right-half of s-plane is mapped into points that fall outside the unit circle in z. This is one of the desirable properties for stability. The $j\Omega$-axis is mapped into the unit circle in z-plane. However, the mapping of the $j\Omega$-axis is not one-to-one. The mapping $\omega = \Omega T$ implies that the interval $-\pi/T \le \Omega \le \pi/T$ maps into the corresponding values of $-\pi \le \omega \le \pi$. Further, the frequency interval $\pi/T \le \Omega \le 3\pi/T$ also maps into the interval $-\pi \le \omega \le \pi$ and, in general, any frequency interval $(2k-1)\pi/T \le \Omega \le (2k+1)\pi/T$, where k is an integer, will also map into the interval $-\pi \le \omega \le \pi$ in the z-plane. Thus the mapping from the analog frequency Ω to the frequency variable ω in the digital domain is many-to-one, which simply reflects the effects of aliasing due to sampling of the impulse response. Figure 8.4 illustrates the mapping from the s-plane to the z-plane.

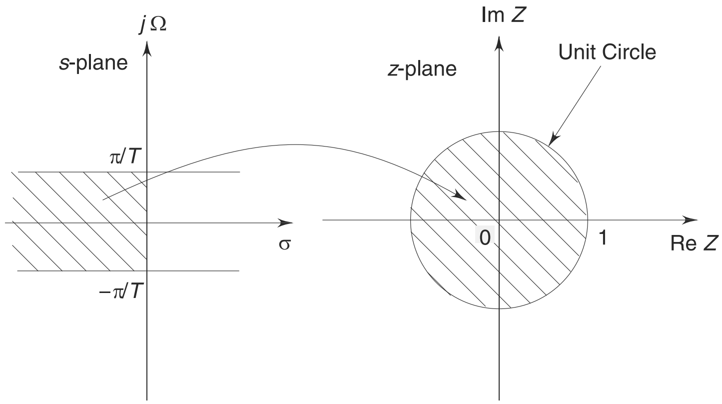


Fig. 8.4 *The Mapping of* $z = e^{sT}$

Some of the properties of the impulse invariant transformation are given below.

$$\frac{1}{(s+s_i)^m} \to \frac{(-1)^{m-1}}{(m-1)!}\frac{d^{m-1}}{ds^{m-1}}\left[\frac{1}{1-e^{-sT}z^{-1}}\right];\; s \to s_i \tag{8.26}$$

$$\frac{s+a}{(s+a)^2+b^2} \rightarrow \frac{1-e^{-aT}(\cos bT)z^{-1}}{1-2e^{-aT}(\cos bT)z^{-1}+e^{-2aT}z^{-2}} \quad (8.27)$$

$$\frac{b}{(s+a)^2+b^2} \rightarrow \frac{e^{-aT}(\sin bT)z^{-1}}{1-2e^{-aT}(\cos bT)z^{-1}+e^{-2aT}z^{-2}} \quad (8.28)$$

Example 8.4 *Convert the analog filter into a digital filter whose system function is*

$$H(s) = \frac{s+0.2}{(s+0.2)^2+9}$$

Use the impulse invariant technique. Assume T = 1s.

Solution The system response of the analog filter is of the standard form

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

where $a = 0.2$ and $b = 3$. The system response of the digital filter can be obtained using Eq. (8.27).

$$H(z) = \frac{1-e^{-aT}(\cos bT)z^{-1}}{1-2e^{-aT}(\cos bT)z^{-1}+e^{-2aT}z^{-2}}$$

$$= \frac{1-e^{-0.2T}(\cos 3T)z^{-1}}{1-2e^{-0.2T}(\cos 3T)z^{-1}+e^{-0.4T}z^{-2}}$$

Taking $T = 1$ s,

$$H(z) = \frac{1-(0.8187)(-0.99)z^{-1}}{1-2(0.8187)(-0.99)z^{-1}+0.6703z^{-2}}$$

That is,

$$H(z) = \frac{1+(0.8105)z^{-1}}{1+1.6210z^{-1}+0.6703z^{-2}}$$

Example 8.5 *For the analog transfer function*

$$H(s) = \frac{1}{(s+1)\,(s+2)}$$

determine H(z) using impulse invariant technique. Assume T = 1 s.

Solution Using partial fractions, $H(s)$ can be written as

$$H(s) = \frac{1}{(s+1)(s+2)} = \frac{A_1}{s+1} + \frac{A_2}{s+2} = \frac{A_1(s+2)+A_2(s+1)}{(s+1)(s+2)}$$

$$1 = A_1(s+2)+A_2(s+1)$$

Letting $s = -2$, we get $A_2 = -1$ and letting, $s = -1$, we get $A_1 = 1$. Therefore,

$$H(s) = \frac{1}{s+1} - \frac{1}{s+2}$$

The system function of the digital filter is obtained by using Eq. (8.23).

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

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

Since $T = 1$ s,

$$H(z) = \frac{0.2326z^{-1}}{1-0.5032z^{-1}+0.0498z^{-2}}$$

Example 8.6 *Determine H(z) using the impulse invariant technique for the analog system function*

$$H(s) = \frac{1}{(s+0.5)(s^2+0.5s+2)}$$

Solution Using partial fractions, $H(s)$ can be written as

$$H(s) = \frac{1}{(s+0.5)(s^2+0.5s+2)} = \frac{A_1}{s+0.5} + \frac{A_2 s + A_3}{s^2+0.5s+2}$$

Therefore,

$$A_1(s^2 + 0.5s + 2) + (A_2 s + A_3)(s + 0.5) = 1$$

Comparing the coefficients of s^2, s and the constants on either side of the above expression, we get

$$A_1 + A_2 = 0$$

$$0.5A_1 + 0.5A_2 + A_3 = 0$$

$$2A_1 + 0.5A_3 = 1$$

Solving the above simultaneous equations, we get $A_1 = 0.5$, $A_2 = -0.5$ and $A_3 = 0$. The system response can be written as,

$$H(s) = \frac{0.5}{s+0.5} - \frac{0.5s}{s^2+0.5s+2}$$

$$= \frac{0.5}{s+0.5} - 0.5\left(\frac{s}{(s+0.25)^2+(1.3919)^2}\right)$$

$$= \frac{0.5}{s+0.5} - 0.5\left(\frac{s+0.25}{(s+0.25)^2+(1.3919)^2} - \frac{0.25}{(s+0.25)^2+(1.3919)^2}\right)$$

$$= \frac{0.5}{s+0.5} - 0.5\left(\frac{s+0.25}{(s+0.25)^2+(1.3919)^2}\right) + 0.0898\left(\frac{1.3919}{(s+0.25)^2+(1.3919)^2}\right)$$

Using Eqs. (8.27) and (8.28),

$$H(z) = \frac{0.5}{1-e^{-0.5T}z^{-1}} - 0.5\left[\frac{1-e^{-0.25T}(\cos 1.3919T)z^{-1}}{1-2e^{-0.25T}(\cos 1.3919T)z^{-1}+e^{-0.5T}z^{-2}}\right]$$

$$+0.0898\left[\frac{e^{-0.25T}(\sin 1.3919T)z^{-1}}{1-2e^{-0.25T}(\cos 1.3919T)z^{-1}+e^{-0.5T}z^{-2}}\right]$$

Letting $T = 1$ s,

$$H(z) = \frac{0.5}{1-0.6065\,z^{-1}} - 0.5\left(\frac{1-0.1385\,z^{-1}}{(1+0.277\,z^{-1}+0.606\,z^{-2}}\right) + 0.0898\left[\frac{0.7663\,z^{-1}}{1-0.277\,z^{-1}+0.606\,z^{-2}}\right]$$

IIR FILTER DESIGN BY THE BILINEAR TRANSFORMATION 8.4

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

$$H(s) = \frac{b}{s+a} \tag{8.29}$$

The differential equation describing the analog filter can be obtained from Eq. (8.29) as shown below.

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

$$sY(s) + aY(s) = bX(s) \tag{8.30}$$

Taking inverse Laplace transform,

$$\frac{\mathrm{d}y(t)}{\mathrm{d}t} + ay(t) = bx(t) \tag{8.31}$$

Equation (8.31) is integrated between the limits $(nT - T)$ and nT

$$\int_{nT-T}^{nT} \frac{\mathrm{d}y(t)}{\mathrm{d}t}\mathrm{d}t + a\int_{nT-T}^{nT} y(t)\mathrm{d}t = b\int_{nT-T}^{nT} x(t)\mathrm{d}t \tag{8.32}$$

The trapezoidal rule for numeric integration is given by

$$\int_{nT-T}^{nT} a(t)\mathrm{d}t = \frac{T}{2}[a(nT) + a(nT-T)] \tag{8.33}$$

Applying Eq. (8.33) in Eq. (8.32), 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

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b}{\dfrac{2}{T}\left(\dfrac{1-z^{-1}}{1+z^{-1}}\right) + a} \tag{8.34}$$

Comparing Eqs. (8.29) and (8.34), we get

$$s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right) = \frac{2}{T}\left(\frac{z-1}{z+1}\right) \tag{8.35}$$

The general characteristic of the mapping $z = e^{sT}$ can be obtained by substituting $s = \sigma + j\,\Omega$ and expressing the complex variable z in the polar form as $z = re^{j\omega}$ in Eq. (8.35).

$$s = \frac{2}{T}\left(\frac{z-1}{z+1}\right) = \frac{2}{T}\left(\frac{re^{j\omega}-1}{re^{j\omega}+1}\right) \tag{8.36}$$

Substituting $e^{j\omega} = \cos\omega - j\sin\omega$ and simplifying, we get

$$s = \frac{2}{T}\left(\frac{r^2-1}{1+r^2+2r\cos\omega} + j\frac{2r\sin\omega}{1+r^2+2r\cos\omega}\right)$$

Therefore,

$$\sigma = \frac{2}{T}\left(\frac{r^2-1}{1+r^2+2r\cos\omega}\right) \tag{8.37}$$

$$\Omega = \frac{2}{T}\left(\frac{2r\sin\omega}{1+r^2+2r\cos\omega}\right) \tag{8.38}$$

From Eq. (8.37), it can be noted that if $r < 1$, then $\sigma < 0$, and if $r > 1$, then $\sigma > 0$. Thus, the left-half of the s-plane maps onto the points inside the unit circle in the z-plane and the transformation results in a stable digital system. Consider Eq. (8.38), for unity magnitude ($r = 1$), σ is zero. In this case,

$$= \frac{2}{T}\left(\frac{\sin\omega}{1+\cos\omega}\right)$$

$$= \frac{2}{T}\left(\frac{2\sin(\omega/2)\cos(\omega/2)}{\cos^2(\omega/2)+\sin^2(\omega/2)+\cos^2(\omega/2)-\sin^2(\omega/2)}\right)$$

$$\Omega = \frac{2}{T}\tan\frac{\omega}{2} \tag{8.39}$$

or equivalently,

$$\omega = 2\tan^{-1}\frac{\Omega T}{2} \tag{8.40}$$

Pre-Warping Equation (8.40) gives the relationship between the frequencies in the two domains and this is shown in Fig. 8.5. It can be noted that the entire range in Ω is mapped only once into the range $-\pi \le \omega \le \pi$. However, as seen in Fig. 8.5, the mapping is non-linear and the lower frequencies in analog domain are expanded in the digital domain, whereas the higher frequencies are compressed. The distortion introduced in the frequency scale of the digital filter to that of the analog filter due to the non linearity of the arctangent function and this effect of the bilinear transform is usually called *frequency warping* the filter design.

The analog filter is designed to compensate for the frequency warping by setting Eqn. (8.39) for every frequency specification so that the corner frequency or center frequency is controlled. This is called *pre-warping* the filter design. When a digital filter is designed as an approximation of an analog filter, the frequency response of the digital filter can be made to match the frequency response of the analog filter by considering the following:

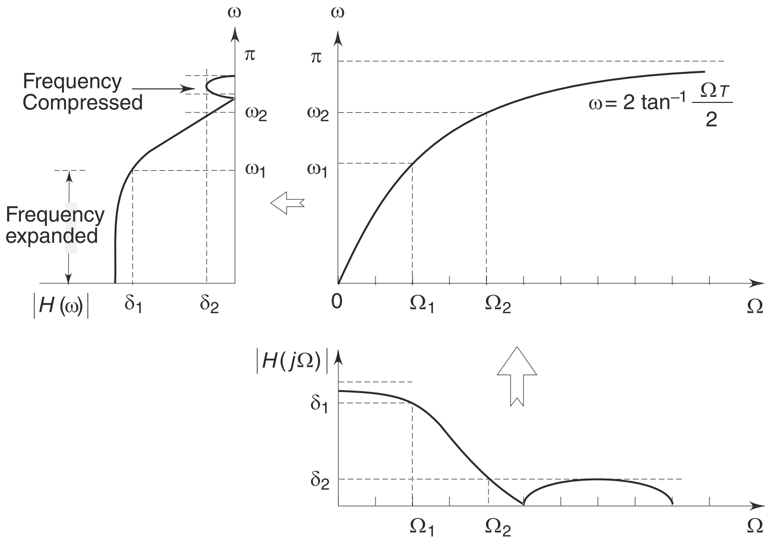


Fig. 8.5 *Relationship between ω and Ω as given in Eq. (8.40)*

The pre-warped frequencies corresponding to f_p and f_s are

$$\Omega_p = \frac{2}{T}\tan\left(\frac{\omega_p}{2}\right), \text{ and } \Omega_s = \frac{2}{T}\tan\left(\frac{\omega_s}{2}\right)$$

For bilinear transformation, $s = \dfrac{2}{T}\left(\dfrac{z-1}{z+1}\right)$

The designer shall use $\Omega_1 = \Omega_p = \tan\left(\dfrac{\omega_p}{2}\right)$, $\Omega_2 = \Omega_s = \tan\left(\dfrac{\omega_s}{2}\right)$ and $s = \left(\dfrac{z-1}{z+1}\right)$ because the factor $\dfrac{2}{T}$ cancels out at numerator and denominator while calculating the order N of filter and $H(z)$.

The warping phenomenon will eliminate the aliasing distortion of the frequency response characteristic such as observed with impulse invariance. It is essential to compensate for the frequency warping by pre-warping the given frequency specifications of the continuous-time system. These pre-warped specifications are used in the bilinear transform to obtain the desired discrete-time system.

Example 8.7 *The transfer function of an analog LPF is*

$$H(s) = \frac{1}{s+1}$$

with a bandwidth of 1 rad/s. Use bilinear transform to design a digital filter with a bandwidth of 20 Hz at a sampling frequency of 60 Hz.

Solution Given $H(s) = \dfrac{1}{s+1}$, $\Omega_p = 1$ rad/s

$$f_p' = 20 \text{ Hz}$$

$$\Omega_p' = 2\pi f_p' = 2\pi \times 20 = 125.66 \text{ rad/s}$$

Analog lowpass filter to lowpass filter transformation is obtained by replacing $s \to \dfrac{\Omega_p}{\Omega_{p'}} . s$

$$H(s) = \frac{1}{\frac{s}{125.66}+1} = \frac{125.66}{s+125.66}$$

The sampling frequency is given by $f_s = 60$ Hz

$$T = \frac{1}{f_s} = \frac{1}{60} = 0.0167\,\text{s}$$

Using bilinear transformation, we get

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

$$H(z) = \frac{125.66}{s+125.66}\Bigg|_{s=\frac{2}{0.0167}\frac{1-z^{-1}}{1+z^{-1}}}$$

$$= \frac{125.66}{120\frac{(1-z^{-1})}{(1+z^{-1})}+125.66} = \frac{125.66(1+z^{-1})}{120(1-z^{-1})+125.66(1+z^{-1})}$$

$$= \frac{125.66(1+z^{-1})}{120-120z^{-1}+125.66+125.66z^{-1}} = \frac{125.66(1+z^{-1})}{245.66+5.66z^{-1}} = \frac{125.66(1+z^{-1})}{245.66(1+0.023z^{-1})}$$

$$H(z) = \frac{0.512(1+z^{-1})}{1+0.023z^{-1}}$$

Example 8.8 *Convert the analog filter with system function*

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

into a digital IIR filter using bilinear transformation. The digital filter should have a resonant frequency of $\omega_r = \frac{\pi}{4}$.

Solution From the system function, we note that $\Omega_c = 3$. The sampling period T can be determined using Eq. (8.39), i.e.

$$\Omega_c = \frac{2}{T}\tan\frac{\omega_r}{2}$$

The sampling period is obtained from the above equation using

$$T = \frac{2}{\Omega_c}\tan\frac{\omega_r}{2} = \frac{2}{3}\tan\frac{\pi}{8} = 0.276\,\text{s}$$

Using bilinear transformation,

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

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

Substituting $T = 0.276$ s,

$$H(z) = \frac{1 + 0.027\,z^{-1} - 0.973\,z^{-2}}{8.572 - 11.84\,z^{-1} + 8.177\,z^{-2}}$$

Example 8.9 *Apply bilinear transformation to*

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

with $T = 0.1$ s.

Solution For bilinear transformation,

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

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

Using $T = 0.1$ s,

$$H(z) = \frac{2}{\left(20\frac{(z-1)}{(z+1)} + 1\right)\left(20\frac{(z-1)}{(z+1)} + 3\right)} = \frac{2(z+1)^2}{(21z-19)(23z-17)}$$

Simplifying further,

$$H(z) = \frac{0.0041(1 + z^{-1})^2}{1 - 1.644\,z^{-1} + 0.668\,z^{-2}}$$

Example 8.10 *A digital filter with a 3 dB bandwidth of* 0.25π *is to be designed from the analog filter whose system response is*

$$H(s) = \frac{\Omega_c}{s + \Omega_c}$$

Use bilinear transformation and obtain H(z).

Solution Using Eq. (8.39),

$$\Omega_c = \frac{2}{T}\tan\frac{\omega_r}{2} = \frac{2}{T}\tan 0.125\,\pi = 0.828/T$$

The system response of the digital filter is given by

$$H(z) = H(s)\Big|_{s=\frac{2}{T}\frac{(z-1)}{(z+1)}} = \frac{\Omega_c}{\frac{2}{T}\frac{(z-1)}{(z+1)} + \Omega_c} = \frac{\frac{0.828}{T}}{\frac{2}{T}\frac{(z-1)}{(z+1)} + \frac{0.828}{T}}$$

$$= \frac{0.828(z+1)}{2(z-1) + 0.828(z+1)}$$

Simplifying, we get further

$$H(z) = \frac{1 + z^{-1}}{3.414 - 1.414\,z^{-1}}$$

Example 8.11 *Using bilinear transformation obtain H(z) if*

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

and $T = 0.1$ s.

Solution For the bilinear transformation,

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

Substituting $T = 0.1$ s,

$$H(z) = \frac{1}{\left(20\frac{(z-1)}{(z+1)}+1\right)^2} = \frac{(z+1)^2}{(21z-19)^2}$$

Further simplifying,

$$H(z) = \frac{0.0476(1+z^{-1})^2}{(1-0.9048\,z^{-1})^2}$$

BUTTERWORTH FILTERS 8.5

The Butterworth low-pass filter has a magnitude response given by

$$|H(j\Omega)| = \frac{A}{\left[1+(\Omega/\Omega_c)^{2N}\right]^{0.5}} \tag{8.41}$$

where A is the filter gain and Ω_c is the 3 dB cut-off frequency and N is the order of the filter. The magnitude response of the Butterworth filter is shown in Fig. 8.6(a). The magnitude response has a maximally flat passband and stopband. It can be seen that by increasing the filter order N, the Butterworth response approximates the ideal response. However, the phase response of the Butterworth filter becomes more non-linear with increasing.

The analog magnitude response of the Butterworth filter with the design parameters is shown in Fig. 8.6(b). Here, $\Omega_1 = \Omega_p$ and $\Omega_2 = \Omega_s$.

The design parameters of the Butterworth filter are obtained by considering the low-pass filter with the desired specifications as given below.

$$\delta_1 \le |H(j\Omega)| \le 1, \quad 0 \le \Omega \le \Omega_1 \tag{8.42a}$$

$$|H(j\Omega)| \le \delta_2, \quad \Omega_2 \le \Omega \le \pi \tag{8.42b}$$

From Fig. 8.6(b),

$$\alpha_p = -20 \log \delta_1$$

$$\alpha_s = -20 \log \delta_2$$

and

$$\delta_1 = \frac{1}{\sqrt{1+\varepsilon^2}}$$

$$\delta_2 = \frac{1}{\sqrt{1+\lambda^2}}$$

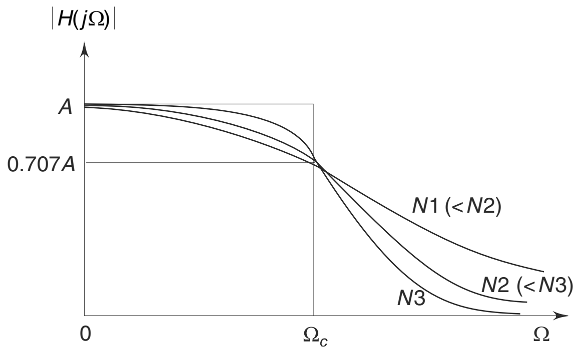


Fig. 8.6(a) *Magnitude Response of a Butterworth Low-pass Filter*

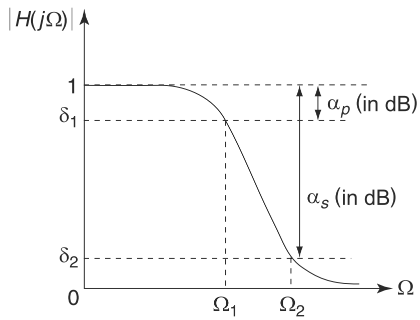


Fig. 8.6(b) *Magnitude Response of Butterworth LPF with Design specifications*

where ε and δ_1 are the parameters specifying allowable passband, and λ and δ_2 are the parameters specifying allowable stopband.

The corresponding analog magnitude response is to be obtained in the design process. Using Eq. (8.41) in Eq. (8.42) and if $A = 1$, we get

$$\delta_1^2 \le \frac{1}{1+(\Omega_1/\Omega_c)^{2N}} \le 12 \tag{8.43a}$$

$$\frac{1}{1+(\Omega_2/\Omega_c)^{2N}} \le \delta_2^2 \tag{8.43b}$$

Equation (8.43) can be written in the form

$$(\Omega_1/\Omega_c)^{2N} \le \frac{1}{\delta_1^2}-1 \tag{8.44a}$$

$$(\Omega_2/\Omega_c)^{2N} \ge \frac{1}{\delta_2^2}-1 \tag{8.44b}$$

Equality is assumed in Eq. (8.44) in order to obtain the filter order N and the 3 dB cut-off frequency Ω_c. Dividing Eq. (8.44b) by Eq. (8.44a)

$$(\Omega_2/\Omega_1)^{2N} = ((1/\delta_2^2)-1)/((1/\delta_1^2)-1) \tag{8.45}$$

From Eq. (8.45), the order of the filter N is given by

$$N = \frac{1}{2}\frac{\log\{((1/\delta_2^2)-1)/((1/\delta_1^2)-1)\}}{\log(\Omega_2/\Omega_1)} \tag{8.46}$$

The value of N is chosen to be next nearest integer to the value of N as given by Eq. (8.46). Using Eq. (8.44), we get

$$\Omega_c = \frac{\Omega_1}{\left[(1/\delta_1^2)-1\right]^{1/2N}} \tag{8.47}$$

The values of Ω_1 and Ω_2 are obtained using the bilinear transformation or impulse invariant transformation techniques, $\Omega = \frac{2}{T}\tan\left(\frac{\omega}{2}\right)$ for bilinear transformation or $\Omega = \frac{\omega}{T}$ for the impulse invariant transformation.

The transfer function of the Butterworth filter is usually written in the factored form as given below.

$$H(s) = \prod_{k=1}^{N/2} \frac{B_k \Omega_c^2}{s^2 + b_k \Omega_c s + c_k \Omega_c^2} \quad N = 2, 4, 6, \ldots \tag{8.48}$$

or

$$H(s) = \frac{B_0 \Omega_c}{s + c_0 \Omega_c} \prod_{k=1}^{(N-1)/2} \frac{B_k \Omega_c^2}{s^2 + b_k \Omega_c s + c_k \Omega_c^2} \quad N = 3, 5, 7, \ldots. \tag{8.49}$$

The coefficients b_k and c_k are given by

$$b_k = 2 \sin [(2k-1)\pi/2N] \text{ and } c_k = 1 \tag{8.50}$$

The parameter B_k can be obtained from

$$A = \prod_{k=2}^{N/2} B_k, \quad \text{for even } N$$

and (8.51)

$$A = \prod_{k=1}^{(N-1)/2} B_k, \quad \text{for odd } N$$

The system function of the equivalent digital filter is obtained from $H(s)$ (Eq. (8.48) or Eq. (8.49)) using the specified transformation technique, viz. impulse invariant technique or bilinear transformation.

For bilinear transformation, $s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right) = \frac{2}{T}\left(\frac{z-1}{z+1}\right)$, and for impulse invariant transformation, $\frac{1}{s-p_i} \rightarrow \frac{1}{1-e^{p_i T} z^{-1}}$ or Eqs. (8.26), (8.27) or Eq. (8.28) may be used.

Poles of a Normalised Butterworth Filter

The Butterworth low-pass filter has a magnitude squared response given by

$$|H(j\Omega)|^2 = \frac{1}{1 + (\Omega / \Omega_c)^{2N}}$$

For a normalised filter, $\Omega_c = 1$. Thus,

$$|H(j\Omega)|^2 = \frac{1}{1 + \Omega^{2N}}$$

The normalised poles in the s-domain can be obtained by substituting $\Omega = s/j$ and equating the denominator polynomial to zero, i.e.

$$1 + \left(\frac{s}{j}\right)^{2N} = 0, \text{ or } 1 + (-s^2)^N = 0$$

The solution to the above expression gives us the poles of the filter. The above expression can be written as

$$(-1)^N s^{2N} = -1$$

Expressing -1 in the polar form,

$$(-1)^N s^{2N} = e^{j(2n-1)\pi}, \qquad n = 1, 2, \ldots, N$$

The poles in the left-half of the s-plane are given by,

$$s_n = \sigma_n + j\,\Omega_n = e^{j(2n+N-1)\pi/2N} = je^{j(2n-1)\pi/2N}$$

Using the polar to rectangular conversion, $e^{j\theta} = \cos\theta + j\sin\theta$, the normalised poles are obtained as

$$s_n = -\sin\left(\frac{2n-1}{2N}\right)\pi + j\cos\left(\frac{2n-1}{2N}\right)\pi$$

where

$$n = \begin{cases} 1, 2, \ldots, (N+1)/2, & \text{for } N \text{ odd} \\ 1, 2, \ldots, N/2, & \text{for } N \text{ even} \end{cases}$$

The unnormalised poles, s'_n, can also be obtained from the normalised poles as shown below,

$$s'_n = s_n\,(\Omega_c)^{-1/N}$$

The normalised poles lie on the unit circle spaced π/N apart.

Example 8.12 *Obtain the system functions of normalised Butterworth filters for order* $N = 1$, 2, 3 *and* 4.

Solution The poles in the left half of the s-plane are given by

$$s_n = je^{j(2n-1)\pi/2N}, \; n = 1, 2, \ldots N$$

The denominator polynomial $D(s)$ is determined by

$$D(s) = (s - s_1)(s - s_2)\ldots(s - s_n)$$

Numerator polynomial $N(s)$ is determined by $D(s)\big|_{s=0}$

Therefore, the system function is $H(s) = \dfrac{N(s)}{D(s)}$

For $N = 1$: Here, $n = 1$

$$s_1 = je^{j(2-1)\frac{\pi}{2}} = j.e^{j\frac{\pi}{2}} = j.\left[\cos\left(\frac{\pi}{2}\right) + j\sin\left(\frac{\pi}{2}\right)\right] = j[j] = -1$$

Denominator polynomial is $D_1(s) = (s - s_1) = s + 1$

Numerator polynomial is $N_1(s) = D_1(s)\big|_{s=0} = 1$

Therefore, the system function is $H_1(s) = \dfrac{N_1(s)}{D_1(s)} = \dfrac{1}{s+1}$

For $N = 2$: Here, $n = 1, 2$

$$s_n = j.e^{j(2n-1)\frac{\pi}{2\times 2}} = j.e^{j(2n-1)\frac{\pi}{4}}$$

$$s_1 = j.e^{j\frac{\pi}{4}} = j(0.707 + j0.707) = (-0.707 + j0.707)$$

$$s_2 = j.e^{j\frac{3\pi}{4}} = j(-0.707 + j0.707) = (-0.707 - j0.707)$$

Denominator polynomial is

$$\begin{aligned} D_2(s) &= (s - s_1)(s - s_2) = [s - (-0.707 + j0.707)][s - (-0.707 - j0.707)] \\ &= [s + 0.707) - j0.707]\cdot[(s + 0.707) + j0.707] \\ &= (s + 0.707)^2 + (0.707)^2 = s^2 + \sqrt{2}s + 1 \end{aligned}$$

Numerator polynomial is $N_2(s) = D_2(s)\big|_{s=0} = 1$

Therefore, the system function is $H_2(s) = \dfrac{N_2(s)}{D_2(s)} = \dfrac{1}{s^2 + \sqrt{2}s + 1}$

For N = 3: Here, n = 1, 2, 3

$$s_n = j.e^{j(2n-1)\frac{\pi}{6}}, \quad n = 1,2,3$$

$$s_1 = j.e^{j\left(\frac{\pi}{6}\right)} = j(0.866 + j0.5) = (-0.5 + j0.866)$$

$$s_2 = j.e^{j\frac{3\pi}{6}} = j \cdot (j) = -1$$

$$s_3 = j.e^{j\frac{5\pi}{6}} = j(0.866 + j0.5) = (-0.5 + j0.866)$$

The denominator polynomial is

$$\begin{aligned} D_3(s) &= (s - s_1)(s - s_2)(s - s_3) \\ &= [s - (-0.5 + j0.866)][s - (-1)][s - (-0.5 + j0.866)] \\ &= [s - (0.5 - j0.866)][s + 1][s + (0.5 - j0.866)] \\ &= [(s + 0.5)^2 + (0.866)^2][s + 1] \\ &= (s^2 + s + 1)(s + 1) \end{aligned}$$

The numerator polynomial is $N_3(s) = D_3(s)\big|_{s=0} = 1$

Therefore, the system function is $H_3(s) = \dfrac{N_3(s)}{D_3(s)} = \dfrac{1}{(s+1)(s^2 + s + 1)}$

For N = 4: Here, n = 1, 2, 3, 4

$$s_n = j.e^{j(2n-1)\frac{\pi}{8}}, n = 1,2,3,4$$

$$s_1 = j.e^{j\left(\frac{\pi}{8}\right)} = j(0.923 + j0.382) = (-0.382 + j0.923)$$

$$s_2 = j.e^{j\frac{3\pi}{8}} = j\ (0.382 + j0.923) = -0.923 + j0.382$$

$$s_3 = j.e^{j\frac{5\pi}{8}} = j(-0.382 + j0.923) = -0.923 - j0.382$$

$$s_4 = j.e^{j\frac{7\pi}{8}} = j(-0.923 + j0.382) = -0.382 - j0.923$$

The denominator polynomial is

$$\begin{aligned} D_4(s) &= (s - s_1)(s - s_2)(s - s_3)(s - s_4) \\ &= [s - (-0.382 + j0.923)][s - (-0.923 + j0.382)] \\ &\quad [s - (-0.923 - j0.382)][s - (-0.382 - j0.923)] \\ &= [(s + 0.382) - j0.923][(s + 0.923) - j0.382] \\ &\quad [(s + 0.923) + j0.382][(s + 0.382) + j0.923] \\ &= [(s + 0.382)^2 + (0.923)^2][(s + 0.923)^2 + (0.382)^2] \\ &= (s^2 + 0.764s + 0.1459 + 0.851)(s^2 + 1.846s + 0.8519 + 0.145) \\ &= (s^2 + 0.764s + 1)[s^2 + 1.846s + 1] \end{aligned}$$

The numerator polynomial is $N_4(s) = D_4(s)|_{s=0} = 1$

Therefore, the system function is

$$H_4(s) = \frac{1}{(s^2 + 0.764s + 1)(s^2 + 1.846s + 1)}$$

Note: Similarly, we can find the denominator polynomials for Butterworth Filters for $N = 5, 6, 7\ldots10$. The denominator polynomials in factorised form for Butterworth Filters are given in Appendix C.

Example 8.13 *Use bilinear transform to design a first-order Butterworth LPF with 3 dB cut-off frequency of* $0.2\,\pi$.

Solution Given 3 dB cut-off frequency $\omega_p = 0.2\,\pi$

The analog transfer function for first order Butterworth filter is

$$H_a(s) = \frac{\Omega_c}{s + \Omega_c}$$

$$\Omega_c = \frac{2}{T}\tan\left(\frac{\omega_p}{2}\right) = 2\tan\left(\frac{0.2\pi}{2}\right) = 0.6498$$

$$H_a(s) = \frac{0.6498}{s + 0.6498}$$

Using bilinear transformation, we get

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

Assuming $T = 1$ s, we have

$$H(z) = \frac{0.6498}{s + 0.6498}\Big|_{s=2\left(\frac{1-z^{-1}}{1+z^{-1}}\right)} = \frac{0.6498}{\frac{2\left(1-z^{-1}\right)}{\left(1+z^{-1}\right)} + 0.6498}$$

$$= \frac{0.6498\left(1+z^{-1}\right)}{2\left(1-z^{-1}\right) + 0.6498\left(1+z^{-1}\right)} = \frac{0.6498\left(1+z^{-1}\right)}{2 - 2z^{-1} + 0.6498 + 0.6498z^{-1}}$$

$$= \frac{0.6498\left(1+z^{-1}\right)}{2.6498 - 1.3502z^{-1}} = \frac{0.6498\left(1+z^{-1}\right)}{2.6498\left(1 - 0.5095z^{-1}\right)}$$

$$H(z) = \frac{0.2452\left(1+z^{-1}\right)}{1 - 0.5095z^{-1}}$$

Example 8.14 *Determine the order of the butterworth filter for the specifications* $\alpha_p = 1$ *dB*, $\alpha_s = 30$ *dB*, $\Omega_p = 200$ rad/s and $\Omega_s = 600$ *rad/s*.

Solution Given $\alpha_p = -20\log\delta_1$

$1 = -20\log\delta_1$

Therefore, $\delta_1 = 0.8912$

Given $\alpha_s = -20\log\delta_2$

$30 = -20\log\delta_2$

Therefore, $\delta_2 = 0.0316$

Given $\Omega_1 = \Omega_p = 200$ rad/s and $\Omega_2 = \Omega_s = 600$ rad/s

The order of the filter is

$$N \geq \frac{1}{2}\frac{\log\left\{\dfrac{\left(\dfrac{1}{\delta_2^2}\right)-1}{\left(\dfrac{1}{\delta_1^2}\right)-1}\right\}}{\log\left(\dfrac{\Omega_2}{\Omega_1}\right)} \geq \frac{1}{2}\frac{\log\left(\dfrac{999}{0.259}\right)}{\log\left(\dfrac{600}{200}\right)} \geq 3.758$$

Hence, $N = 4$

Example 8.15 *Determine the order and poles of lowpass Butterworth filter that has a 3 dB attenuation at 500 Hz and 40 dB attenuation at 1000 Hz.*

Solution Given

$$\Omega_p = \Omega_1 = 2\pi \times 500 = 1000\pi \text{ rad/s}$$
$$\Omega_s = \Omega_2 = 2\pi \times 1000 = 2000\pi \text{ rad/s}$$
$$\alpha_p = 3 = -20 \log \delta_1$$

Therefore, $\delta_1 = 0.707$

$$\alpha_s = 40 = -20 \log \delta_2$$

Therefore, $\delta_2 = 0.01$

$$N \geq \frac{1}{2}\frac{\log\left\{\dfrac{\left(\dfrac{1}{\delta_2^2}\right)-1}{\left(\dfrac{1}{\delta_1^2}\right)-1}\right\}}{\log\left(\dfrac{\Omega_2}{\Omega_1}\right)} \geq \frac{1}{2}\frac{\log\left(\dfrac{9999}{1}\right)}{\log\left(\dfrac{2000\pi}{1000\pi}\right)} \geq 6.644$$

Hence, $N = 7$

The normalised poles are obtained as

$$s_n = je^{j(2n-1)\frac{\pi}{2N}} \quad n = 1, 2, \ldots N$$

Substituting $n = 1, 2, 3, 4, 5, 6, 7$, we get

$$s_1 = je^{j\pi/14} = -0.2225 + j0.975$$
$$s_2 = je^{j3\pi/14} = -0.6234 + j0.7818$$
$$s_3 = je^{j5\pi/14} = -0.9009 + j0.4339$$
$$s_4 = je^{j7\pi/14} = -1$$
$$s_5 = je^{j9\pi/14} = -0.9009 - j0.4339$$
$$s_6 = je^{j11\pi/14} = -0.6234 - j0.7818$$
$$s_7 = je^{j13\pi/14} = -0.2225 - j0.975$$

Example 8.16 *Design an analog Butterworth filter that has a –2 dB passband attenuation at a frequency of 20 rad/s and at least 10 dB stopband attenuation at 30 rad/s.*

Solution Given

$$\Omega_1 = \Omega_p = 20 \text{ rad/s}$$

$$\Omega_2 = \Omega_s = 30 \text{ rad/s}$$

$$\alpha_p = 2 = -20 \log \delta_1$$

Therefore, $\delta_1 = 0.794$

$$\alpha_s = 10 = -20 \log \delta_2$$

Therefore, $\delta_2 = 0.316$

The order of the filter is

$$N \geq \frac{1}{2}\frac{\log\left\{\dfrac{\left(\dfrac{1}{\delta_2^2}\right)-1}{\left(\dfrac{1}{\delta_1^2}\right)-1}\right\}}{\log\left(\dfrac{\Omega_2}{\Omega_1}\right)} \geq \frac{1}{2}\frac{\log\left(\dfrac{9.014}{0.586}\right)}{\log\left(\dfrac{30}{20}\right)} \geq 3.37$$

Hence, $N = 4$

The cut-off frequency is

$$\Omega_c = \frac{\Omega_1}{\left[\left(\dfrac{1}{\delta_1^2}\right)-1\right]^{\frac{1}{2N}}} = \frac{20}{\left[\left(\dfrac{1}{0.794^2}\right)-1\right]^{\frac{1}{8}}} = 21.387$$

The system transfer function is

$$H(s) = \prod_{K=1}^{N/2} \frac{B_k\Omega_c^2}{s^2 + b_k\Omega_c s + C_k\Omega_c}$$

$$\left[\frac{B_1\Omega_c^2}{s^2 + b_1\Omega_c s + C_1\Omega_c^2}\right]\left[\frac{B_2\Omega_c^2}{s^2 + b_2\Omega_c s + C_2\Omega_c^2}\right]$$

$$b_1 = 2\sin\frac{\pi}{8} = 0.7654, C_1 = 1$$

$$b_2 = 2\sin\frac{3\pi}{8} = 1.8478, \; C_2 = 1$$

$$B_1B_2 = 1$$

Therefore, $B_1 = B_2 = 1$

$$H(s) = \left[\frac{457.4}{\left(s^2 + 16.3688s + 457.4\right)}\right]\left[\frac{457.4}{s^2 + 39.5180s + 457.4}\right]$$

Example 8.17 *Design a highpass filter for the given specifications, α_p = 3 dB, α_s = 15 dB, Ω_p = 1000 rad/s and Ω_s = 500 rad/s.*

Solution For the given highpass filter $\Omega_p = 1000$ rad/s and $\Omega_s = 500$ rad/s

For equivalent lowpass filter $\Omega_1 = \Omega_p = 500$ rad/s and $\Omega_2 = \Omega_s = 1000$ rad/s

$$\alpha_p = -20 \log \delta_1 = 3 \text{ dB}$$

Therefore, $\delta_1 = 0.708$

$$\alpha_s = -20 \log \delta_2 = 15 \text{ dB}$$

Therefore, $\delta_2 = 0.178$

The order of the filter is

$$N \geq \frac{1}{2}\frac{\left(\dfrac{\left(1/\delta_2^2\right)-1}{\left(1/\delta_1^2\right)-1}\right)}{\log\left(\dfrac{\Omega_2}{\Omega_1}\right)} \geq \frac{1}{2}\frac{\log\left(\dfrac{30.56}{0.995}\right)}{\log\left(\dfrac{1000}{500}\right)} \geq 2.458$$

Hence, $N = 3$

The transfer function for order is

$$H(s) = \frac{1}{(s+1)\left(s^2+s+1\right)}$$

To obtain the transfer function for the highpass filter,

$$s \to \frac{\Omega_{c(HPF)}}{s} = \frac{1000}{3}$$

Therefore, $H_{HPF}(s) = \dfrac{s^3}{(s+1000)\left(s^2+1000s+1000^2\right)}$

Example 8.18 *Design a third order Butterworth digital filter using an impulse invariant technique.*

Solution The transfer function of a normalised third-order Butterworth filter is

$$H(s) = \frac{1}{(s+1)\left(s^2+s+1\right)} = \frac{A_1}{s+1} + \frac{A_2}{s+0.5+j0.866} + \frac{A_3}{s+0.5-j0.866}$$

$$1 = A_1\left(s^2+s+1\right) + A_2(s+1)(s+0.5-j0.866) + A_3(s+1)(s+0.5+j0.866)$$

Substituting $s = -1$, we get $A_1 = 1$

Substituting $s = 0$ and $s = 1$, we can find A_2 and A_3.

$$A_2 = -0.5 + j\,0.288$$

$$A_3 = -0.5 - j\,0.288$$

$$H(s) = \frac{1}{s+1} + \frac{-0.5+j0.288}{s+0.5+j0.866} + \frac{-0.5-j0.288}{s+0.5-j0.866}$$

Using the transformation,

$$\frac{1}{s-p_i} \to \frac{1}{1-e^{p_i T} z^{-1}}$$

$$H(z) = \frac{1}{1-e^{-1}z^{-1}} + \frac{-0.5+j0.288}{1-e^{-0.5-j0.866}z^{-1}} + \frac{-0.5-j0.288}{1-e^{-0.5+j0.866}z^{-1}}$$

$$= \frac{1}{1-0.368z^{-1}} - \frac{1-0.66z^{-1}}{1-0.786z^{-1}+0.368z^{-2}}$$

Example 8.19 *Determine H(z) for a Butterworth filter satisfying the following constraints*

$$\sqrt{0.5} \le |H(e^{j\omega})| \le 1 \qquad 0 \le \omega \le \pi/2$$

$$|H(e^{j\omega})| \le 0.2 \qquad 3\pi/4 \le \omega \le \pi$$

with T = 1 s. Apply impulse invariant transformation.

Solution Given $\delta_1 = \sqrt{0.5} = 0.707$, $\delta_2 = 0.2$, $\omega_1 = \pi/2$ and $\omega_2 = 3\pi/4$.

Step (i) Determination of the analog filter's edge frequencies:

From Eq. (8.25),

$$\Omega_1 = \frac{\omega_1}{T} = \frac{\pi}{2} \quad \text{and} \quad \Omega_2 = \frac{\omega_2}{T} = \frac{3\pi}{4}$$

Therefore, $\Omega_2/\Omega_1 = 1.5$

Step (ii) Determination of the order of the filter:

From Eq. (8.46),

$$N \ge \frac{1}{2}\frac{\log\{((1/\delta_2^2)-1)/((1/\delta_1^2)-1)\}}{\log(\Omega_2/\Omega_1)} = \frac{1}{2}\frac{\log\{24/1\}}{\log(1.5)} = 3.91$$

Let $N = 4$.

Step (iii) Determination of −3 dB cut-off frequency:

From Eq. (8.47),

$$\Omega_c = \frac{\Omega_1}{[(1/\delta_1^2)-1]^{1/2N}} = \frac{\pi/2}{[(1/0.707^2)-1]^{1/8}} = \frac{\pi}{2}$$

Step (iv) Determination of $H_a(s)$:

From Eq. (8.48),

$$H(s) = \prod_{k=1}^{N/2} \frac{B_k \Omega_c^2}{s^2 + b_k \Omega_c s + c_k \Omega_c^2}$$

$$= \left(\frac{B_1 \Omega_c^2}{s^2 + b_1 \Omega_c s + c_1 \Omega_c^2}\right)\left(\frac{B_2 \Omega_c^2}{s^2 + b_2 \Omega_c s + c_2 \Omega_c^2}\right)$$

From Eq. (8.50),

$$b_1 = 2\sin\frac{\pi}{8} = 0.76536,\ c_1 = 1$$

$$b_2 = 2\sin\frac{3\pi}{8} = 1.84776,\ c_2 = 1$$

$B_1 B_2 = 1$. Therefore $B_1 = B_2 = 1$.

Therefore,

$$H(s)=\left(\frac{2.467}{s^2+1.2022s+2.467}\right)\left(\frac{2.467}{s^2+2.9025+2.467}\right)$$

Using partial fractions,

$$H(s)=\left(\frac{A_1s+A_2}{s^2+1.2022s+2.467}\right)+\left(\frac{A_3s+A_4}{s^2+2.9025+2.467}\right)$$

Comparing the above two expressions, we get

$$6.086 = (s^2 + 2.9025s + 2.467)(A_1s + A_2) + (s^2 + 1.2022s + 2.467)(A_3s + A_4)$$

Comparing the coefficients of s^3, s^2, s and the constants, we get a set of simultaneous equations.

$$A_1 + A_3 = 0$$
$$2.9025\, A_1 + A_2 + 1.2022\, A_3 + A_4 = 0$$
$$2.467\, A_1 + 2.9025\, A_2 + 2.467\, A_3 + 1.2022\, A_4 = 0$$
$$A_2 + A_4 = 2.467$$

Solving, we get $A_1 = -1.4509$, $A_2 = -1.7443$, $A_3 = 1.4509$ and $A_4 = 4.2113$.
Therefore,

$$H(s)=-\left(\frac{1.4509s+1.7443}{s^2+1.2022s+2.467}\right)+\left(\frac{1.4509s+4.2113}{s^2+2.9025s+2.467}\right)$$

Let $H(s) = H_1(s) + H_2(s)$. Therefore,

$$H_1(s)=-\left(\frac{1.4509s+1.7443}{s^2+1.2022s+2.467}\right) \quad \text{and } H_2(s)=\left(\frac{1.4509s+4.2113}{s^2+2.9025s+2.467}\right)$$

Rearranging $H_1(s)$ into the standard form,

$$H(s)=-\left(\frac{1.4509s+1.7443}{s^2+1.2022s+2.467}\right)$$
$$=-1.4509\left(\frac{s+1.2022}{(s+0.601)^2+1.451^2}\right)$$
$$=(-1.4509)\left[\frac{s+0.601}{(s+0.601)^2+1.451^2}+\frac{0.601}{(s+0.601)^2+1.451^2}\right]$$
$$=(-1.4509)\left(\frac{s+0.601}{(s+0.601)^2+1.451^2}\right)-(0.601)\left(\frac{1.451}{(s+0.601)^2+1.451^2}\right)$$

Similarly, $H_2(s)$ can be written as

$$H_2(s)=(1.4509)\left(\frac{s+1.45}{(s+1.45)^2+0.604^2}\right)+(3.4903)\left(\frac{0.604}{(s+1.45)^2+0.604^2}\right)$$

Step (*v*) Determination of $H(z)$: Using Eqs. (8.27) and (8.28),

$$H_1(z)=(-1.4509)\frac{1-e^{-0.601T}(\cos 1.451T)z^{-1}}{1-2e^{-0.601T}(\cos 1.451T)z^{-1}+e^{-1.202T}z^{-2}}$$
$$-(0.601)\frac{e^{-0.601T}(\sin 1.451T)z^{-1}}{1-2e^{-0.601T}(\cos 1.451T)z^{-1}+e^{-1.202T}z^{-2}}$$

and

$$H_2(z) = (1.4509)\frac{1 - e^{-1.45T}(\cos 0.604T)z^{-1}}{1 - 2e^{-1.45T}(\cos 0.604T)z^{-1} + e^{-2.9T}z^{-2}}$$

$$+(3.4903)\frac{e^{-1.45T}(\sin 0.604T)z^{-1}}{1 - 2e^{-1.45T}(\cos 0.604T)z^{-1} + e^{-2.9T}z^{-2}}$$

where $H(z) = H_1(z) + H_2(z)$.

Upon simplifying, we get

$$H(z) = \frac{-1.4509 - 0.2321z^{-1}}{1 - 01310z^{-1} + 0.3006z^{-2}} + \frac{1.4509 + 0.1848z^{-1}}{1 - 0.3862z^{-1} + 0.055z^{-2}}$$

Example 8.20 *Design a digital Butterworth filter that satisfies the following constraint using bilinear transformation. Assume T* = 1 s.

$$0.9 \le |H(e^{j\omega})| \le 1 \qquad 0 \le \omega \le \pi/2$$

$$|H(e^{j\omega})| \le 0.2 \qquad 3\pi/4 \le \omega \le \pi$$

Solution Given $\delta_1 = 0.9$, $\delta_2 = 0.2$, $\omega_1 = \pi/2$ and $\omega_2 = 3\pi/4$.

Step (*i*) Determination of the analog filter's edge frequencies:

From Eq. (8.39),

$$\Omega_1 = \frac{2}{T}\tan\frac{\omega_1}{2} = 2\tan\frac{\pi}{4} = 2 \text{ and } \Omega_2 = \frac{2}{T}\tan\frac{\omega_2}{2} = 2\tan\frac{3\pi}{8} = 4.828$$

Therefore, $\Omega_2/\Omega_1 = 2.414$

Step (*ii*) Determination of the order of the filter:

From Eq. (8.46),

$$N \ge \frac{1}{2}\frac{\log\{((1/\delta_2^2) - 1)/((1/\delta_1^2) - 1)\}}{\log(\Omega_2/\Omega_1)} = \frac{1}{2}\frac{\log\{24/0.2346\}}{\log(2.414)} = 2.626$$

Let $N = 3$.

Step (*iii*) Determination of −3 dB cut-off frequency:

From Eq. (8.47),

$$\Omega_c = \frac{\Omega_1}{[(1/\delta_1^2) - 1]^{1/2N}} = \frac{2}{[(1/0.9^2) - 1]^{1/6}} = 2.5467$$

Step (*iv*) Determination of $H_a(s)$:

From Eq. (8.49),

$$H(s) = \frac{B_0\Omega_c}{s + c_0\Omega_c}\prod_{k=1}^{(N-1)/2}\frac{B_k\Omega_c^2}{s^2 + b_k\Omega_c s + c_k\Omega_c^2}$$

$$= \left(\frac{B_0\Omega_c}{s + c_0\Omega_c}\right)\left(\frac{B_1\Omega_c^2}{s^2 + b_1\Omega_c s + c_1\Omega_c^2}\right)$$

From Eq. (8.50),

$b_1 = 2\sin\frac{\pi}{6} = 1$, $c_0 = 1$ and $c_1 = 1$

$B_0 B_1 = 1$. Therefore, $B_0 = B_1 = 1$.

Therefore,

$$H(s) = \left(\frac{2.5467}{s+2.5467}\right)\left(\frac{6.4857}{s^2+2.5467s+6.4857}\right)$$

Step (*v*) Determination of $H(z)$:

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

That is,

$$H(z) = \left(\frac{2.5467}{2\dfrac{(z-1)}{(z+1)}+2.5467}\right)\left(\frac{6.4857}{\left[2\dfrac{(z-1)}{(z+1)}\right]^2+2.5467s+6.4857}\right)$$

Simplifying, we get

$$H(z) = \frac{16.5171(z+1)^3}{70.83z^3+31.1205z^2+27.2351z+2.948} \text{ or } H(z) = \frac{0.2332(1+z^{-1})^3}{1+0.4394z^{-1}+0.3845z^{-2}+0.0416z^{-3}}$$

Example 8.21 *Using the bilinear transformation, design a highpass filter, monotonic in passband with cut-off frequency of 1000 Hz and down 10 dB at 350 Hz. The sampling frequency is 5000 Hz.*

Solution Given $\alpha_p = 3 = -20 \log \delta_1$

Therefore, $\delta_1 = 0.7079$

Given $\alpha_s = 10 = -20 \log \delta_2$

Therefore, $\delta_2 = 0.3162$

$$\omega_p = 2000\pi$$

$$\omega_s = 700\pi$$

$$F = 5000 \text{ Hz}$$

$$T = \frac{1}{5000} = 2\times10^{-4} \text{ s}$$

For a lowpass filter,

$$\omega_p = 700\pi$$

$$\omega_s = 2000\pi$$

$$\Omega_1 = \Omega_p = \frac{2}{T}\tan\frac{\omega_p T}{2} = 2235 \text{ rad/s}$$

$$\Omega_2 = \Omega_s = \frac{2}{T}\tan\frac{\omega_s T}{2} = 7265 \text{ rad/s}$$

The order of the filter is given by

$$N \ge \frac{1}{2}\frac{\log\left\{\dfrac{\left(\dfrac{1}{\delta_2^2}\right)-1}{\left(\dfrac{1}{\delta_1^2}\right)-1}\right\}}{\log\left(\dfrac{\Omega_2}{\Omega_1}\right)} \ge \frac{1}{2}\frac{\log\left(\dfrac{9.0017}{0.9955}\right)}{\log\left(\dfrac{7265}{2235}\right)} \ge 0.9339$$

Hence, $N = 1$

The first-order Butterworth filter for $\Omega_c = 1$ rad/s is

$$H(s) = \frac{1}{s+1}$$

For a highpass filter, the 3 dB cut-off frequency is

$$\Omega_c = \Omega_p = 7265 \text{ rad/s}$$

Using the transformation from lowpass to highpass filter, i.e. $s \rightarrow \Omega_c/s$,

$$H_{HPF}(s) = \frac{s}{s+7265}$$

Using bilinear transformation,

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

$$= \frac{\dfrac{2}{2\times10^{-4}}\left(\dfrac{z-1}{z+1}\right)}{\dfrac{2}{2\times10^{-4}}\left(\dfrac{z-1}{z+1}\right)+7265} = \frac{0.5792\left(1-z^{-1}\right)}{1-0.1584z^{-1}}$$

Example 8.22 *Design a Butterworth digital filter using bilinear transformation that satisfies the following specifications.*

$$0.89 \le |H(\omega)| \le 1.0\ ; 0 \le \omega \le 0.2\pi$$
$$|H(\omega)| \le 0.18; 0.3\pi \le \omega \le \pi$$

Solution Given $\delta_1 = 0.89$, $\delta_2 = 0.18$, $\omega_p = 0.2\pi$ and $\omega_s = 0.3\pi$

Assume $T = 1$ s

The edge frequencies of analog filter are

$$\Omega_1 = \frac{2}{T}\tan\frac{\omega_p T}{2} = 0.6498$$

$$\Omega_2 = \frac{2}{T}\tan\frac{\omega_s T}{2} = 1.019$$

The order of the filter is

$$N \ge \frac{1}{2}\frac{\log\left\{\dfrac{\left(\dfrac{1}{\delta_2^2}\right)-1}{\left(\dfrac{1}{\delta_1^2}\right)-1}\right\}}{\log\left(\dfrac{\Omega_2}{\Omega_1}\right)} \ge \frac{1}{2}\frac{\log\left(\dfrac{29.8642}{0.2625}\right)}{\log\left(\dfrac{1.019}{0.6498}\right)} \ge 5.26$$

Hence, $N = 6$

The 3 dB cut-off frequency is

$$\Omega_c = \frac{\Omega_1}{\left[\left(1/\delta_1^2\right)-1\right]^{1/2N}} = \frac{0.6498}{\left[\left(1/(0.89)^2\right)-1\right]^{1/12}} = 0.7264 \text{ rad/s}$$

The transfer function of the Butterworth filter is

$$H(s) = \prod_{k=1}^{N/2} \frac{B_k \Omega_c^2}{s^2 + b_k \Omega_c s + C_k \Omega_c^2} = \prod_{k=1}^{3} \frac{B_k \Omega_c^2}{s^2 + b_k \Omega_c s + C_k \Omega_c^2}$$

$$b_k = 2\sin\left[(2k-1)\frac{\pi}{2N}\right] \quad \text{and} \quad C_k = 1$$

$$b_1 = 2\sin\frac{\pi}{12} = 0.5176$$

$$b_2 = 2\sin\frac{3\pi}{12} = 1.414$$

$$b_3 = 2\sin\frac{5\pi}{12} = 1.9318$$

$$A = \prod_{k=1}^{N/2} B_k = B_1 B_2 B_3 = 1$$

Therefore,

$$H(s) = \frac{(0.7264)^6}{\left(s^2 + 0.376s + (0.7264)^2\right)\left(s^2 + 1.0271s + (0.7264)^2\right)\left(s^2 + 1.403s + (0.7264)^2\right)}$$

Example 8.23 *Using bilinear transformation, design a digital bandpass Butterworth filter for the following specifications:*

Sampling frequency = 8 kHz

α_p = 2 dB in 800 Hz $\le f \le$ 1000 Hz

α_s = 20 dB in $0 \le f \le$ 400 Hz and $f \ge$ 2000 Hz

Solution Given $\alpha_p = \delta_1 = 2$ dB and $\alpha_s = \delta_2 = 20$ dB

Hence,

$$T = \frac{1}{f_s} = \frac{1}{8000}$$

$$\frac{\omega_1 T}{2} = \frac{2\times\pi\times 400}{2\times 8000} = \frac{\pi}{20}$$

$$\frac{\omega_\ell T}{2} = \frac{2\times\pi\times 800}{2\times 8000} = \frac{\pi}{10}$$

$$\frac{\omega_u T}{2} = \frac{2\pi\times 1000}{2\times 8000} = \frac{\pi}{8}$$

$$\frac{\omega_2 T}{2} = \frac{2\pi\times 2000}{2\times 8000} = \frac{\pi}{4}$$

Prewarped analog frequencies are given by

$$\frac{\Omega_1 T}{2} = \tan\frac{\omega_1 T}{2} = \tan\frac{\pi}{20} = 0.1584$$

$$\frac{\Omega_l T}{2} = \tan\frac{\omega_l T}{2} = \tan\frac{\pi}{10} = 0.325$$

$$\frac{\Omega_u T}{2} = \tan\frac{\omega_u T}{2} = \tan\frac{\pi}{8} = 0.4142$$

$$\frac{\Omega_2 T}{2} = \tan\frac{\omega_2 T}{2} = \tan\frac{\pi}{4} = 1$$

Therefore,

$$A = \frac{-\Omega_1^2 + \Omega_l\Omega_u}{\Omega_1\left(\Omega_u - \Omega_l\right)} = \frac{-(0.1584)^2 + (0.325)(0.4142)}{0.1584(0.4142 - 0.325)} = 11.303$$

$$B = \frac{\Omega_2^2 - \Omega_l\Omega_u}{\Omega_2\left(\Omega_u - \Omega_l\right)} = \frac{1 - (0.4142)(0.325)}{1.(0.4142 - 0.325)} = 9.7016$$

Hence, $\frac{\Omega_s}{\Omega_p} = \min\{|A|, |B|\} = 9.7016$

Order, $N \geq \frac{\log_{10}\left[\frac{10^{0.1\delta_2} - 1}{10^{0.1\delta_1} - 1}\right]}{2\log_{10}\left(\frac{\Omega_s}{\Omega_p}\right)} \geq \frac{\log_{10}\sqrt{\frac{10^2 - 1}{10^{0.2} - 1}}}{\log_{10}(9.7016)} \geq 1.1290$

Hence, $N = 2$

The second-order normalised Butterworth lowpass filter transfer function is given by

$$H(s) = \frac{1}{s^2 + 1.4142s + 1}$$

The transformation for the bandpass filter is

$$s \to \frac{s^2 + \Omega_l\Omega_u}{s\left(\Omega_u - \Omega_l\right)} = \frac{s^2 + 0.1346}{s(0.0892)}$$

$$H(s) = \left.\frac{1}{s^2 + 1.4142s + 1}\right|_{s = \frac{s^2 + 0.1346}{(0.0892)s}}$$

$$= \frac{1}{\left(\frac{s^2 + 0.1346}{0.0892s}\right)^2 + 1.4142\left(\frac{s^2 + 0.1346}{0.0892s}\right) + 1}$$

$$= \frac{0.0079s^2}{s^4 + 0.2692s^2 + 0.0181 + 1.4142(0.0892)s\left(s^2 + 0.1346\right) + 0.0079s^2}$$

$$= \frac{0.0079s^2}{s^4 + 0.2692s^2 + 0.0181 + 0.126s\left(s^2 + 0.1346\right) + 0.0079s^2}$$

$$= \frac{0.0079s^2}{s^4 + 0.2692s^2 + 0.0181 + 0.126s^3 + 0.0169s + 0.0079s^2}$$

$$H(s) = \frac{0.0079s^4}{s^4 + 0.126s^3 + 0.2771s^2 + 0.0169s + 0.0181}$$

$$H(z) = H(s)\Big|_{s=\frac{1-z^{-1}}{1+z^{-1}}}$$

$$= \frac{0.0079\left(\frac{1-z^{-1}}{1+z^{-1}}\right)^2}{\left(\frac{1-z^{-1}}{1+z^{-1}}\right)^4 + 0.126\left(\frac{1-z^{-1}}{1+z^{-1}}\right)^3 + 0.277\left(\frac{1-z^{-1}}{1+z^{-1}}\right)^2 + 0.169\left(\frac{1-z^{-1}}{1+z^{-1}}\right) + 0.0181}$$

$$= \frac{0.0079\left(1-z^{-1}\right)^2}{\left(1-z^{-1}\right)^4 + 0.126\left(1-z^{-1}\right)^3\left(1+z^{-1}\right) + 0.2692\left(1-z^{-1}\right)^2\left(1+z^{-1}\right) + 0.169\left(1-z^{-1}\right)\left(1+z^{-1}\right)^3 + 0.0181\left(1+z^{-1}\right)^4}$$

$$= \frac{0.0079\left(1-z^{-2}\right)^2}{1-4z^{-1}+6z^{-2}-4z^{-3}+z^{-4}+0.126\left(1-2z^{-1}+2z^{-3}-z^{-4}\right)+0.2677\left(1-2z^{-2}+z^{-4}\right) + 0.169\left(1+2z^{-1}-2z^{-3}-z^{-4}\right)+0.0181\left(1+4z^{-1}+6z^{-2}+4z^{-3}+z^{-4}\right)}$$

$$= \frac{0.0079\left(1-z^{-2}\right)^2}{7.1871-3.8416z^{-1}+0.644z^{-2}-4.086z^{-3}+0.9908z^{-4}}$$

Example 8.24 *Convert the analog filter with system function* $H(s) = \frac{1}{s}$ *into a digital filter using bilinear transformation. Assume* $T = 2s$.

(a) *Find the difference equation for the digital filter relating the input x(n) to the output y(n).*
(b) *Sketch the magnitude* $|H(j\Omega)|$ *and* $\theta(\Omega)$ *for the analog filter.*
(c) *Sketch* $|H(\omega)|$ *and* $\theta(\omega)$ *for the digital filter.*
(d) *Compare the magnitude and phase characteristics obtained in parts (ii) and (iii).*

Solution (a) We know that $H(z) = H(s)\Big|_{s=\frac{2}{T}\frac{(1-z^{-1})}{(1+z^{-1})}}$

Given $T = 2$ seconds

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

$$\frac{Y(z)}{X(z)} = \frac{1+z^{-1}}{1-z^{-1}}$$

$$Y(z) - z^{-1}Y(z) = X(z) + z^{-1}X(z)$$

Taking inverse z-Transform, we get

$$y(n) - y(n-1) = x(n) + x(n-1)$$

Hence, $y(n) = y(n-1) + x(n) + x(n-1)$

(b) The frequency response of the analog integrator is obtained by assuming $s = j\Omega$ in $H(s)$.

Hence, $H(j\Omega) = \dfrac{1}{j\Omega}$

$$|H(j\Omega)| = \frac{1}{|\Omega|} \text{ and } \theta(\Omega) = \angle H(j\Omega) = \begin{cases} -\dfrac{\pi}{2}, & \Omega > 0 \\ +\dfrac{\pi}{2}, & \Omega < 0 \end{cases}$$

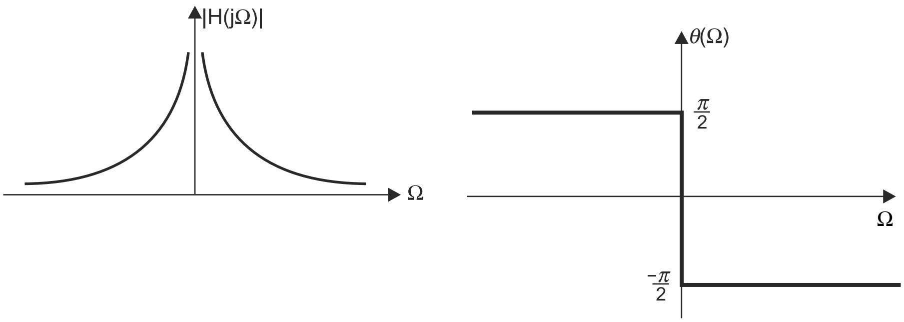


Fig. E8.24(a) *Magnitude and phase responses of analog filter*

(c) We know that
$$H(z) = H(s)\Big|_{s=\frac{1-z^{-1}}{1+z^{-1}}}$$
$$H(z) = \frac{1+z^{-1}}{1-z^{-1}}$$

The frequency response of the digital filter is obtained by assuming $z = e^{j\omega}$ in $H(z)$.

That is,
$$H\left(e^{j\omega}\right) = H(\omega) = \frac{1+e^{-j\omega}}{1-e^{-j\omega}}$$
$$= \frac{e^{-j\omega/2}\left[e^{j\omega/2} + e^{-j\omega/2}\right]}{e^{-j\omega/2}\left[e^{-j\omega/2} - e^{-j\omega/2}\right]} = -j\cot\frac{\omega}{2}$$

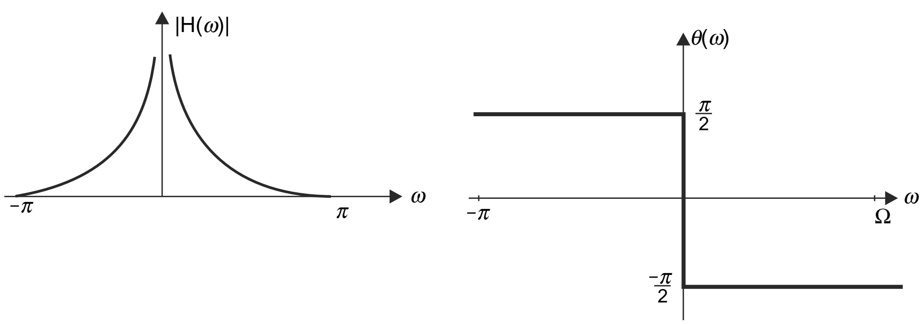


Fig. E8.24(b) *Magnitude and phase responses of digital filter*

Hence, its magnitude, $|H(\omega)| = \left|\cot\left(\frac{\omega}{2}\right)\right|$

and its phase, $$\theta(\omega) = \angle H(\omega) = \begin{cases} -\frac{\pi}{2}; & \omega > 0 \\ +\frac{\pi}{2}; & \omega < 0 \end{cases}$$

(d) From the figures in parts (ii) and (iii), the magnitude and phase characteristics of the digital filter and analog filter are identical. This comparison is made for Ω varying from – π to + π.

Example 8.25 *A second-order Butterworth lowpass analog filter H(s) with a half-power frequency of 1 rad/s is converted to a digital filter H(z), using the bilinear transformation at a sampling period, T* = 1 s.

(a) Find the transfer function H(s) of the analog filter.
(b) Determine the transfer function H(z) of the digital filter.
(c) Verify whether the dc gains of H(z) and H(s) are identical.
(d) Verify whether the gains of H(z) and H(s) at their respective half-power frequencies are identical.

Solution (a) The transfer function of the second-order normalised lowpass Butterworth filter is

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

(b) The transfer function $H(z)$ of the digital filter is

$$H(z) = H_2(s)\Big|_{s=\frac{2}{1}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)}$$

$$= \frac{(z+1)^2}{7.8284z^2 - 6z + 2.1716} = \frac{(1+z^{-1})^2}{2.1716z^{-2} - 6z^{-1} + 7.8284}$$

(c) The dc gain of $H(s)$ is obtained by assuming $s = 0$ in $H(s)$ and dc gain of $H(z)$ is obtained by assuming $z = 1$ in $H(z)$.

Therefore,

The dc gain of $$H(s) = \frac{1}{s^2 + \sqrt{2s} + 1}\Big|_{s=0} = 1$$

The dc gain of $$H(z) = \frac{(1+z^{-1})^2}{2.1716z^{-1} - 6z^{-1} + 7.8284}\Big|_{z=1} = 1$$

It shows that the dc gains of $H(s)$ and $H(z)$ are identical.

(d) Gain of $H(s)$ at Ω = 1 rad/s is obtained by assuming $s = j$ in $H(s)$

$$H(j) = \frac{1}{s^2 + \sqrt{2s} + 1}\Big|_{s=j}$$

$$= \frac{1}{-1 + \sqrt{2j} + 1} = \frac{1}{j\sqrt{2}}$$

Therefore, the magnitude of gain $\left|H(j)\right| = \frac{1}{\sqrt{2}} = 0.707$

Gain of $H(z)$ at $\omega = 1$ rad is obtained by assuming $z = e^{j}$ in $H(z)$.

$$H\left(e^{j}\right) = \frac{\left(1+e^{-j}\right)^2}{2.1716e^{-j2} - 6e^{-j} + 7.8284}$$

Therefore, the magnitude of gain $|H(e^{j})| = 0.6421$

Therefore, in this case, the gains of $H(z)$ and $H(s)$ at their respective power frequencies are not identical.

CHEBYSHEV FILTERS 8.6

The Chebyshev low-pass filter has a magnitude response given by

$$\left|H(j\Omega)\right| = \frac{A}{[1+\varepsilon^2 C_N^2(\Omega/\Omega_c)]^{0.5}} \tag{8.52}$$

where A is the filter gain, ε is a constant and Ω_c is the 3 dB cut-off frequency. The Chebyshev polynomial of the I kind of Nth order, $C_N(x)$ is given by

$$C_N(x) = \begin{cases} \cos(N\cos^{-1}x), & \text{for } |x| \le 1 \\ \cos(N\cos\text{h}^{-1}x), & \text{for } |x| \ge 1 \end{cases} \tag{8.53}$$

The magnitude response of the Chebyshev filter is shown in Fig. 8.7. The magnitude response has equiripple passband and maximally flat stopband. It can be seen that by increasing the filter order N, the Chebyshev response approximates the ideal response. The phase response of the Chebyshev filter is more non-linear than the Butterworth filter for a given filter length N.

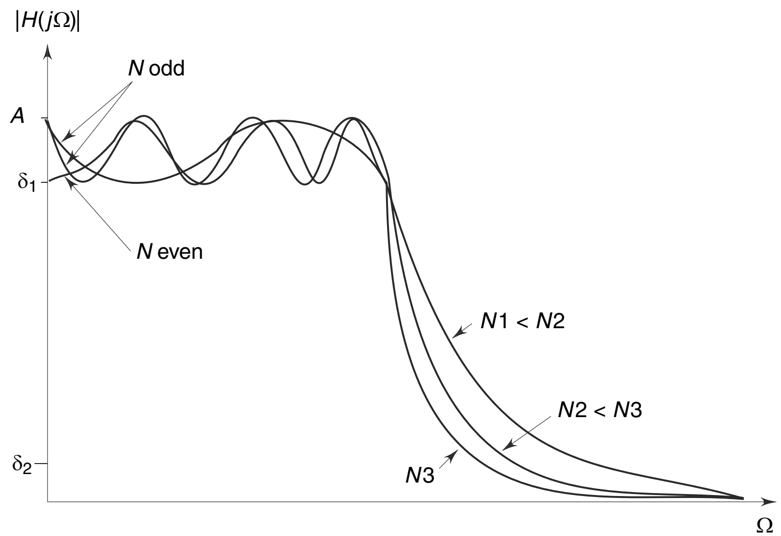


Fig. 8.7 *Magnitude Response of a Low-pass Chebyshev Filter*

The design parameters of the Chebyshev filter are obtained by considering the low-pass filter with the desired specifications as below.

$$\delta_1 \le |H(e^{j\omega})| \le 1 \qquad 0 \le \omega \le \omega_1 \tag{8.54a}$$

$$|H(e^{j\omega})| \le \delta_2 \qquad \omega_2 \le \omega \le \pi \tag{8.54b}$$

The corresponding analog magnitude response is to be obtained in the design process. Using Eqs. (8.52) in (8.54) and if $A = 1$, we get

$$\delta_1^2 \le \frac{1}{1+\varepsilon^2 C_N^2(\Omega_1/\Omega_c)} \le 1 \tag{8.55a}$$

$$\frac{1}{1+\varepsilon^2 C_N^2(\Omega_1/\Omega_c)} \le \delta_2^2 \tag{8.55b}$$

Assuming $\Omega_c = \Omega_1$, we will have $C_N(\Omega_c/\Omega_c) = C_N(1) = 1$. Therefore, Eq. (8.55a) can be written as

$$\delta_1^2 \le \frac{1}{1+\varepsilon^2}$$

Assuming equality in the above equation, the expression for ε is

$$\varepsilon = \left[\frac{1}{\delta_1^2} - 1\right]^{0.5} \tag{8.56}$$

The order of the analog filter N can be determined from Eq. (8.55b). Assuming $\Omega_c = \Omega_1$,

$$C_N(\Omega_2/\Omega_1) \ge \frac{1}{\varepsilon}\left[\frac{1}{\delta_2^2} - 1\right]^{0.5}$$

Since $\Omega_2 > \Omega_1$,

$$\cosh[N\cosh^{-1}(\Omega_2/\Omega_1)] \ge \frac{1}{\varepsilon}\left[\frac{1}{\delta_2^2} - 1\right]^{0.5}$$

or

$$N \ge \frac{\cosh^{-1}\left\{\frac{1}{\varepsilon}\left[\frac{1}{\delta_2^2} - 1\right]^{0.5}\right\}}{\cosh^{-1}(\Omega_2/\Omega_1)} \tag{8.57}$$

Choose N to be next nearest integer to the value given by Eq. (8.57). The values of Ω_1 and Ω_2 are obtained using the bilinear transformation or impulse invariant transformation techniques; $\Omega = \frac{2}{T}\tan\left(\frac{\omega}{2}\right)$ for bilinear transformation or $\Omega = \frac{\omega}{T}$ for the impulse invariant transformation.

The transfer function of Chebyshev filters are usually written in the factored form as given in Eqs. (8.48) or (8.49). The coefficients b_k and c_k are given by

$$b_k = 2y_N \sin[(2k-1)\pi/2N]$$

$$c_k = y_N^2 + \cos^2\frac{(2k-1)\pi}{2N} \tag{8.58}$$

$$c_0 = y_N$$

The parameter y_N is given by

$$y_N = \frac{1}{2}\left\{\left[\left(\frac{1}{\varepsilon^2}+1\right)^{0.5}+\frac{1}{\varepsilon}\right]^{\frac{1}{N}} - \left[\left(\frac{1}{\varepsilon^2}+1\right)^{0.5}+\frac{1}{\varepsilon}\right]^{-\frac{1}{N}}\right\} \quad (8.59)$$

The parameter B_k can be obtained from

$$\frac{A}{(1+\varepsilon^2)^{0.5}} = \prod_{k=1}^{N/2}\frac{B_k}{c_k}, \quad \text{for } N \text{ even} \quad (8.60)$$

and

$$A = \prod_{k=0}^{\frac{N-1}{2}}\frac{B_k}{c_k} \text{ for } N \text{ odd}$$

The system function of the equivalent digital filter is obtained from $H(s)$ (Eq. (8.48) or Eq. (8.49)) using the specified transformation technique, viz. impulse invariant technique or Bilinear transformation.

Poles of a Normalised Chebyshev Filter

The Chebyshev low-pass filter has a magnitude squared response given by

$$|H(j\Omega)|^2 = \frac{1}{1+\varepsilon^2 C_N^2(\Omega/\Omega_c)}$$

For a normalised filter, $\Omega_c = 1$. Thus,

$$|H(j\Omega)|^2 = \frac{1}{1+\varepsilon^2 C_N^2(\Omega)}$$

The normalised poles in the s-domain can be obtained by substituting $\Omega = \frac{s}{j} = -js$ and equating the denominator polynomial to zero, i.e.

$$1 + \varepsilon^2 C_N^2(-js) = 0$$

The solution to the above expression gives us the poles of the filter. The above expression can be written as

$$C_N(-js) = \pm\frac{j}{\varepsilon} = \cos[N\cos^{-1}(-js)]$$

The cosine term in the above equation has a complex argument. Using trigonometric identities for the imaginary terms and with minor manipulations, the poles of the normalised low-pass analog Chebyshev filter is given by

$$s_n = -\sin x \sinh y + j\cos x \cosh y = \sigma_n + j\Omega_n$$

where

$$n = \begin{cases} 1,2,\ldots,(N-1)/2 & \text{for } N \text{ odd} \\ 1,2,\ldots,N/2 & \text{for } N \text{ even}\end{cases}$$

and

$$x = (2n-1)\frac{\pi}{2N} \quad n = 1,2,\ldots,N$$

$$y = \pm\frac{1}{N}\sinh^{-1}\left(\frac{1}{\varepsilon}\right)$$

The unnormalised poles, s'_n, can also be obtained from the normalised poles as shown below,

$$s'_n = s_n\,\Omega_c$$

The normalised poles lie on an ellipse in the s-plane and the equation of the ellipse is given by

$$\frac{\sigma_n^2}{\sinh^2 y} + \frac{\Omega_n^2}{\cosh^2 y} = 1$$

Example 8.26 *Discuss the method of determination of Chebyshev Polynomials and their properties.*

Solution

Determination of Chebyshev Polynomials

The Chebyshev polynomials of the I kind of Nth order, $C_N(x)$ is given by

$$C_N(x) = \begin{cases} \cos(N\cos^{-1} x) & \text{for } |x| \le 1 \\ \cosh(N\cosh^{-1} x) & \text{for } |x| < 1 \end{cases}$$

Chebyshev Polynomial is obtained as follows.

$$x = \frac{\Omega}{\Omega_c};$$

Since Ω_c is the normalised frequency, i.e. $\Omega_c = 1$, $x = \Omega$

(*i*) ***For*** $0 \le \Omega \le 1$, ***frequency below cut-off***

In this region, the Chebyshev Polynomial is given by

$$C_N(\Omega) = \cos\,(N\cos^{-1}(\Omega)$$

Let $\cos^{-1}\Omega = \theta$ (or) $\Omega = \cos\theta$

$$C_N(\Omega) = \cos\,(N\cos^{-1}(\cos\theta)) = \cos\,(N\theta)$$

For $N = 0$,

$$C_0(\Omega) = \cos\,(0 \times \theta) = \cos\,(0) = 1$$

For $N = 1$,

$$C_1(\Omega) = \cos\,(\theta) = \Omega$$

For $N = 2$,

$$C_2(\Omega) = \cos\,(2\theta) = 2\cos^2\theta - 1 = 2\Omega^2 - 1$$

For $N = 3$,

$$C_3(\Omega) = \cos\,(3\theta) = 4\cos^3\theta - 3\cos\theta = 4\Omega^3 - \Omega$$

For $N = 4$,

$$C_4(\Omega) = \cos\,(4\theta) = 2(\cos 2\theta)^2 - 1 = 2\,(2\Omega^2 - 1)^2 - 1$$
$$= 2(4\Omega^4 - 4\Omega^2 + 1) - 1 = 8\Omega^4 - 8\Omega^2 + 1$$

Similarly the Chebyshev Polynomials for $N = 5, 6\ldots$ can be computed. The Chebyshev Polynomials for $N = 0$ to 10 are listed in Appendix C.

(*ii*) ***For*** $\Omega \ge 1$, ***frequency above cut off frequency***

The Chebyshev Polynomial $C_N(x)$ for this region is given by

$$C_N(\Omega) = \cosh\,(N\cosh^{-1}\Omega)$$

Let $\cosh^{-1}\Omega = y$

$$\Omega = \cosh y = \frac{e^{y} + e^{-y}}{2}$$

For $y >> 1$,

$$\Omega = \cosh y \cong \frac{e^{y}}{2} \quad \text{(or)} \quad e^{y} = 2\Omega$$

Taking natural logarithm, we get

$$y = \ln 2\Omega = \cosh^{-1}\Omega$$

Therefore,

$$C_N(\Omega) = \cosh(N \ln 2\Omega)$$
$$= \cosh(\ln (2\Omega)^N)$$

$$C_N(\Omega) = \frac{e^{\ln(2\Omega)^N} + e^{-\ln(2\Omega)^N}}{2}$$

For $\Omega >> 1$, $C_N(\Omega)$ becomes

$$C_N(\Omega) = \frac{e^{\ln(2\Omega)^N}}{2} = \frac{1}{2}(2\Omega)^N$$

$$C_N(\Omega) = \frac{1}{2}(2\Omega)^N$$

For $N = 0$,

$$C_0(\Omega) = \frac{1}{2}(2\Omega)^0 = \frac{1}{2}$$

For $N = 1$,

$$C_1(\Omega) = \frac{1}{2}(2\Omega)^1 = \Omega$$

For $N = 2$,

$$C_2(\Omega) = \frac{1}{2}(2\Omega)^2 = \frac{1}{2} \times 2^2\Omega^2 = 2\Omega^2$$

For $N = 3$,

$$C_3(\Omega) = \frac{1}{2}(2\Omega)^3 = \frac{1}{2} \times 2^3\Omega^3 = 4\Omega^3$$

Properties of Chebyshev Polynomial

(i) $C_N(x) | \leq 1$ for all $| x | \leq 1$

(ii) $C_N(1) = 1$ for all N

(iii) The roots of the polynomials $C_N(x)$ occur in the interval $-1 \leq x \leq 1$.

Example 8.27 *Design a Chebyshev filter with a maximum passband attenuation of 2.5 dB at $\Omega_p = 20$ rad/s and the stopband attenuation of 30 dB at $\Omega_s = 50$ rad/s.*

Solution

Given $\Omega_1 = \Omega_p = 20$ rad/s and $\Omega_2 = \Omega_s = 50$ rad/s

$$\alpha_p = 2.5 = -20 \log \delta_1$$

Therefore, $\delta_1 = 0.7499$

$$\alpha_s = 30 = -20 \log \delta_2$$

Therefore, $\delta_2 = 0.0316$

The order of the filter is

$$N \ge \frac{\cos h^{-1}\left\{\frac{1}{\varepsilon}\left[\frac{1}{\delta_2^2}-1\right]^{0.5}\right\}}{\cos h^{-1}\left(\frac{\Omega_2}{\Omega_1}\right)}$$

where $$\varepsilon = \left[\frac{1}{\delta_1^2}-1\right]^{0.5} = \left[\frac{1}{(0.7499)^2}-1\right]^{0.5} = 0.882$$

Therefore, $$N \ge \frac{\cos h^{-1}\left\{\left(\frac{1}{0.882}\right)\left(\frac{1}{(0.0316)^2}-1\right)^{0.5}\right\}}{\cos h^{-1}\left(\frac{50}{20}\right)} \ge \frac{\cos h^{-1}(35.8614)}{\cos h^{-1}(2.5)} \ge 2.727$$

Hence, $N = 3$

$$\beta = \left[\frac{\sqrt{1+\varepsilon^2}+1}{\varepsilon}\right]^{\frac{1}{N}} = \left[\frac{\sqrt{1+(0.882)^2}+1}{0.882}\right]^{\frac{1}{3}} = 1.3826$$

$$r_1 = \Omega_p\left[\frac{\beta^2+1}{2\beta}\right] = 21.06$$

$$r_2 = \Omega_p\left[\frac{\beta^2-1}{2\beta}\right] = 6.60$$

$$\theta_k = \frac{\pi}{2}+(2k+1)\frac{\pi}{2N},\ k = 0,1,....(N-1)$$

$$= \frac{\pi}{2}+(2k+1)\frac{\pi}{6},\ k = 0,1,2 = \frac{2\pi}{3},\ \pi,\ \frac{4\pi}{3}$$

$s_k = r_2 \cos\theta_k + jr_1 \sin\theta_k$

$s_1 = -3.3 + j\,18.23$

$s_2 = -6.6$

$s_3 = -3.3 + j\,18.23$

Denominator of $H(s) = (s + 6.6)(s^2 + 6.6s + 343.2)$

Numerator of $H(s) = 2265.27$

Hence, the transfer function $H(s) = \dfrac{2265.27}{(s+6.6)(s^2+6.6s+343.2)}$.

Example 8.28 *Design a digital Chebyshev filter to satisfy the constraints*

$$0.707 \le |H(e^{j\omega})| \le 1, \qquad 0 \le \omega \le 0.2\pi$$

$$|H(e^{j\omega})| \le 0.1, \qquad 0.5\pi \le \omega \le \pi$$

using bilinear transformation and assuming $T = 1$ s.

Solution Given $\delta_1 = 0.707$, $\delta_2 = 0.1$, $\omega_1 = 0.2\pi$ and $\omega_2 = 0.5\pi$.

Step (i) Determination of the analog filter's edge frequencies:

From Eq. (8.39),

$$\Omega_c = \Omega_1 = \frac{2}{T}\tan\frac{\omega_1}{2} = 2\tan 0.1\pi = 0.6498$$

$$\Omega_2 = \frac{2}{T}\tan\frac{\omega_2}{2} = 2\tan 0.25\pi = 2$$

Therefore, $\Omega_2/\Omega_1 = 3.0779$

Step (ii) Determination of the order of the filter:

From Eq. (8.56),

$$\varepsilon = \left[\frac{1}{\delta_1^2} - 1\right]^{0.5} = \left[\frac{1}{0.707^2} - 1\right]^{0.5} = 1$$

From Eq. (8.57),

$$N \geq \frac{\cosh^{-1}\left\{\frac{1}{\varepsilon}\left[\frac{1}{\delta_2^2} - 1\right]^{0.5}\right\}}{\cosh^{-1}(\Omega_2/\Omega_1)} = \frac{\cosh^{-1}\left\{1\left[\frac{1}{0.1^2} - 1\right]^{0.5}\right\}}{\cosh^{-1}(3.0779)} = 1.669$$

Let $N = 2$.

Step (iii) Determination of $H(s)$:

From Eq. (8.48),

$$H(s) = \prod_{k=1}^{N/2} \frac{B_k \Omega_c^2}{s^2 + b_k \Omega_c s + c_k \Omega_c^2} = \frac{B_1 \Omega_c^2}{s^2 + b_1 \Omega_c s + c_1 \Omega_c^2}$$

From Eqs. (8.58), (8.59) and (8.60),

$$y_N = \frac{1}{2}\left\{\left[\left(\frac{1}{\varepsilon^2} + 1\right)^{0.5} + \frac{1}{\varepsilon}\right]^{\frac{1}{N}} - \left[\left(\frac{1}{\varepsilon^2} + 1\right)^{0.5} + \frac{1}{\varepsilon}\right]^{-\frac{1}{N}}\right\}$$

$$= \frac{1}{2}\left\{[2.414]^{\frac{1}{2}} - [2.414]^{-\frac{1}{2}}\right\} = 0.455$$

$$b_1 = 2y_2 \sin[(2k-1)\pi/2N] = 0.6435$$

$$c_1 = y_2^2 + \cos^2\frac{(2k-1)\pi}{2N} = 0.707$$

For N even,

$$\prod_{k=1}^{N/2} \frac{B_k}{c_k} = \frac{A}{(1+\varepsilon^2)^{0.5}} = 0.707$$

That is,

$$\frac{B_1}{c_1} = 0.707, \text{ and hence } B_1 = 0.5$$

The system function is

$$H(s) = \frac{0.5(0.6498)^2}{s^2 + (0.6435)(0.6498)s + (0.707)(0.6498)^2}$$

On simplifying, we get

$$H(s) = \frac{0.2111}{s^2 + 0.4181s + 0.2985}$$

Step (*iv*) Determination of $H(z)$. Using bilinear transformation:

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

That is,

$$H(z) = \frac{0.2111}{\left(2\frac{(z-1)}{(z+1)}\right)^2 + 0.4181\left(2\frac{(z-1)}{(z+1)}\right) + 0.2985}$$

$$= \frac{0.2111(z+1)^2}{5.1347z^2 - 7.403z - 3.4623}$$

Rearranging,

$$H(z) = \frac{0.041(1+z^{-1})^2}{1 - 1.4418z^{-1} + 0.6743z^{-2}}$$

Example 8.29 *The specification of the desired lowpass digital filter is*

$$0.9 \le |H(\omega)| \le 1.0, \qquad 0 \le \omega \le 0.25\pi$$

$$|H(\omega)| \le 0.24, \qquad 0.5\pi \le \omega \le \pi$$

Design a Chebyshev digital filter using impulse invariant transformation.

Solution Given $\delta_1 = 0.9$, $\delta_2 = 0.24$, $\omega_p = 0.25\pi$ and $\omega_s = 0.5\pi$

Therefore, $\dfrac{\Omega_2}{\Omega_1} = \dfrac{\omega_s T}{\omega_p T} = \dfrac{0.5\pi T}{0.25\pi T} = 2$

where, $\varepsilon = \left[\dfrac{1}{\delta_1^2} - 1\right]^{\frac{1}{2}} = \left[\dfrac{1}{0.9^2} - 1\right]^{\frac{1}{2}} = 0.484$

The order of the filter is

$$N \ge \frac{\cosh^{-1}\left\{\frac{1}{\varepsilon}\left[\frac{1}{\delta_2^2} - 1\right]^{0.5}\right\}}{\cosh^{-1}\left(\frac{\Omega_2}{\Omega_1}\right)} \ge \frac{\cosh^{-1}\left\{\frac{1}{0.484}\left[\frac{1}{(0.24)^2} - 1\right]^{0.5}\right\}}{\cosh^{-1}(2)} \ge 2.136$$

Therefore, $N = 3$

$$\beta = \left[\frac{\sqrt{1+\varepsilon^2} + 1}{\varepsilon}\right]^{\frac{1}{N}} = \left[\frac{\sqrt{1+(0.484)^2} + 1}{0.484}\right]^{\frac{1}{3}} = 1.6337$$

$$r_1 = \Omega_p\left[\frac{\beta^2+1}{2\beta}\right] = 0.25\pi\left[\frac{(1.6337)^2+1}{2(1.6337)}\right] = 0.882$$

$$r_2 = \Omega_p\left[\frac{\beta^2-1}{2\beta}\right] = 0.25\pi\left[\frac{(1.6337)^2-1}{2(1.6337)}\right] = 04012$$

The poles are

$$s_k = r_2\cos\theta_k + jr_1\sin\theta_k$$

$$\theta_k = \frac{\pi}{2} + (2k+1)\frac{\pi}{2N}, \qquad k = 0,1,\ldots, (N-1)$$

$$= \frac{2\pi}{3}, \pi, \frac{4\pi}{3}$$

$$s_1 = (0.4012)\cos\left(\frac{2\pi}{3}\right) + j0.882\sin\left(\frac{2\pi}{3}\right) = -0.2 + j0.764$$

$$s_2 = -0.4012$$

$$s_3 = -0.2 - j0.764$$

Denominator of the transfer function $H(s)$ is

$$(s+0.4012)(s+0.2-j0.764)(s+0.2+j0.764)$$

Numerator of $H(s) = 0.25$

Hence, the transfer function $H(s)$ is

$$H(s) = \frac{0.25}{(s+0.4012)(s+0.2-j0.764)(s+0.2+j0.764)}$$

$$= \frac{A_1}{s+0.4012} + \frac{A_2}{s+0.2-j0.764} + \frac{A_3}{s+0.2+j0.764}$$

$$A_1 = H(s)\times(s+0.4012)\big|_{s=0.4012} = 0.4$$

$$A_2 = H(s)\times(s+0.2-j0.764)\big|_{s=-0.2+j0.764}$$

$$= \frac{0.25}{(-0.2+j0.764+0.4012)(-0.2+j0.764+0.2+j0.764)} = -0.138 + j0.5242$$

$$A_3 = A_2^* = -0.138 - j0.5242$$

$$H(s) = \frac{0.4}{s+0.4012} + \frac{-0.138+j0.5242}{s+0.2-j0.764} + \frac{-0.138-j0.5242}{s+0.2+j0.764}$$

Using the transformation,

$$\frac{1}{s-p_i} \rightarrow \frac{1}{1-e^{p_iT}z^{-1}}$$

$$H(z) = \frac{0.4}{1-e^{-0.4012}z^{-1}} + \frac{-0.138+j0.5242}{1-e^{-0.2}e^{+j0.764}z^{-1}} - \frac{0.138+j0.5242}{1-e^{-0.2}e^{-j0.764}z^{-1}}$$

Hence, $H(z) = \dfrac{0.4}{1-0.6695z^{-1}} - \dfrac{0.138-j0.5242}{1-(0.5912+j0.5664)z^{-1}} - \dfrac{0.138+j0.5242}{1-(0.5912-j0.5664)z^{-1}}.$

Example 8.30 *Find the transfer function of a Chebyshev lowpass discrete time filter to meet the following specifications: passband:* 0 *to* 0.5 MHz *with* 0.2 dB *ripple stopband edge:* 1 MHz *with attenuation of at least* 50 dB.

Solution Given $\frac{\Omega_s}{\Omega_p} = \frac{2\pi \times 1 \times 10^6}{2\pi \times 0.5 \times 10^6} = 2$, $\delta_1 = 0.2$ dB and $\delta_2 = 50$ dB

To find *H(z)*: Chebyshev Lowpass digital filter design:

Step I: To find the order of the filter (N):

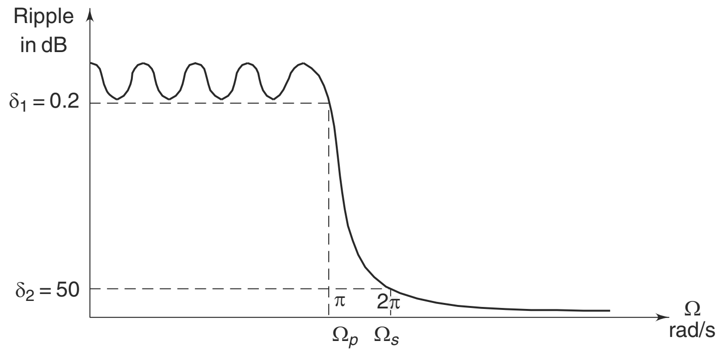


Fig. E8.30

$$N = \frac{\cosh^{-1}\sqrt{\frac{10^{0.1\delta_2}-1}{10^{0.1\delta_1}-1}}}{\cosh^{-1}\left(\frac{\Omega_s}{\Omega_p}\right)} = \frac{\cosh^{-1}\sqrt{\frac{10^5-1}{10^{0.02}-1}}}{\cosh^{-1}(2)} = 6.06$$

Therefore, $N \approx 7$

Step II:

$$r_1 = \Omega_p\left(\frac{\beta^2+1}{2\beta}\right)$$

$$r_2 = \Omega_p\left(\frac{\beta^2+1}{2\beta}\right)$$

$$\beta = \left[\frac{\sqrt{1+\varepsilon^2}+1}{\varepsilon}\right]^{\frac{1}{N}}$$

Here, $\varepsilon = \sqrt{10^{0.1\delta_1}-1} = \sqrt{10^{0.02}} = 0.2171$

Therefore, $\beta = \left[\frac{\sqrt{1+(0.2171)^2}+1}{0.2171}\right]^{\frac{1}{7}} = 1.38$

$$r_1 = \pi \times 10^6\left(\frac{(1.38)^2+1}{2(1.38)}\right) = 3.31 \times 10^6$$

$$r_2 = \pi \times 10^6(0.289) = 1.03 \times 10^6$$

Step III: To find poles:

$$\varphi_k = \frac{\pi}{2} + (2k+1)\frac{\pi}{2N} \quad k = 0,1,...., N-1 = \frac{\pi}{2} + (2k+1)\frac{\pi}{14}$$

Therefore, $\varphi_0 = \frac{8\pi}{14}, \varphi_1 = \frac{10\pi}{14}, \varphi_2 = \frac{12\pi}{14}, \varphi_3 = \frac{14\pi}{14}, \varphi_4 = \frac{16\pi}{14},$

$$\varphi_5 = \frac{18\pi}{14} \text{ and } \varphi_6 = \frac{20\pi}{14}$$

$$x_k = r_2 \cos\varphi_k = 1.03\times10^6 \cos\varphi_k$$

$$y_k = r_1 \sin\varphi_k = 3.31\times10^6 \sin\varphi_k$$

$$s_k = x_k + jy_k = 1.03\times10^6 \cos\varphi_k + j3.31\times10^6 \sin\varphi_k$$

Step IV: To find $H_a(s)$:

$$H_a(s) = \frac{1}{(s-s_0)\cdots(s-s_6)}$$

$$s_0 = -2.3\times10^5 + j3.23\times10^6 \qquad s_1 = -6.4\times10^5 + j2.6\times10^6$$

$$s_2 = -9.3\times10^5 + j1.4\times10^6 \qquad s_3 = -1.03\times10^6$$

$$s_4 = -9.3\times10^5 - j1.4\times10^6 \qquad s_5 = -6.4\times10^5 - j2.6\times10^6$$

$$s_6 = -2.3\times10^5 - j3.21\times10^6$$

Here, $s_0 = s_6^*;\ s_1 = s_5^*;\ s_2 = s_4^*$

Hence, $H_a(s) = \frac{1}{(s-s_0)(s-s_0^*)(s-s_1)(s-s_1^*)(s-s_2)(s-s_2^*)(s-s_3)}$

$$= \frac{1}{(s^2 - 2(-2.3\times10^5)s + 1.05\times10^{13})(s^2 - 2(-6.4\times10^5)s + 7.2\times10^{12})(s^2 - 2(-9.3\times10^5)s + 2.8\times10^{12})(s + 1.03\times10^6)}$$

$$= \frac{1}{(s^2 - 4.6\times10^5 s + 1.05\times10^{13})(s^2 + 12.8\times10^5 s + 7.2\times10^{12})(s^2 + 18.6\times10^5 s + 2.8\times10^{12})(s + 1.03\times10^6)}$$

Example 8.31 *Design a Chebyshev filter with a maximum passband attenuation of 2.5* dB *at* $\Omega_p = 20$ *rad/s and the stopband attenuation of 30* dB *at* $\Omega_s = 50$ *rad/s.*

Solution

Given $\Omega_p = \Omega_1 = 20$ rad/s

$$-20 \log \delta_1 = 2.5$$

$$\delta_1 = 0.749$$

Given $\Omega_s = \Omega_2 = 50$ rad/s

$$-20 \log \delta_2 = 30$$

$$\delta_2 = 0.0316$$

The attenuation constant $\varepsilon = \left(\frac{1}{\delta_1^2} - 1\right)^{\frac{1}{2}}$

$$= \left(\frac{1}{(0.749)^2} - 1\right)^{\frac{1}{2}} = 0.8819$$

$$\frac{\Omega_s}{\Omega_p} = \frac{50}{20} = 2.5$$

$$N \ge \frac{\cosh^{-1}\left\{\frac{1}{\varepsilon}\left[\frac{1}{\delta_2^2} - 1\right]^{\frac{1}{2}}\right\}}{\cosh^{-1}\left(\frac{\Omega_2}{\Omega_1}\right)} \ge \frac{\cosh^{-1}\left\{\frac{1}{0.8819}\left[\frac{1}{(0.0316)^2} - 1\right]^{\frac{1}{2}}\right\}}{\cosh^{-1}(2.5)}$$

$$\ge \frac{\cosh^{-1}(35.8654)}{\cosh^{-1}(2.5)} \ge \frac{4.2727}{1.5667} \ge 2.72$$

$$N = 3$$

The analog cut-off frequency

$$\Omega_c \approx \Omega_p = 20 \text{ rad/s}$$

For N is odd,

$$H_a(s) = \frac{B_0\Omega_c}{s + c_0\Omega_c} \prod_{k=1}^{\frac{N-1}{2}} \frac{B_k\Omega_c^2}{s^2 + b_k\Omega_c s + c_k\Omega_c^2}$$

$$= \frac{B_0 \times 20}{s + (c_0 \times 20)} \prod_{k=1}^{\frac{3-1}{2}} \frac{B_k (20)^2}{s^2 + b_k 20s + c_k (20)^2}$$

$$= \frac{20B_0}{s + 20c_0} \times \frac{20B_1}{s^2 + b_1 20s + (20)^2 c_1}$$

$$b_k = 2Y_N \sin\left(\frac{(2k-1)\pi}{2N}\right)$$

$$Y_N = \frac{1}{2}\left\{\left[\left(\frac{1}{\varepsilon^2} + 1\right)^{\frac{1}{2}} + \frac{1}{\varepsilon}\right]^{\frac{1}{N}} - \left[\left(\frac{1}{\varepsilon^2} + 1\right)^{\frac{1}{2}} + \frac{1}{\varepsilon}\right]^{-\left(\frac{1}{N}\right)}\right\}$$

$$= \frac{1}{2}\left\{\left[\left(\frac{1}{(0.8819)^2} + 1\right)^{\frac{1}{2}} + \frac{1}{0.8819}\right]^{\frac{1}{3}} - \left[\left(\frac{1}{(0.8819)^2} + 1\right)^{\frac{1}{2}} + \frac{1}{0.8819}\right]^{-\frac{1}{3}}\right\}$$

$$= \frac{1}{2}\left\{[1.51187+1.134]^{\frac{1}{3}} - [1.51187+1.134]^{-\frac{1}{3}}\right\} = \frac{1}{2}\{1.3831-0.723\}$$

$$Y_N = 0.33$$

$$b_1 = 2Y_N \sin\left(\frac{(2-1)\pi}{2\times 3}\right) = 2\times 0.33 \sin\left[\frac{\pi}{6}\right]$$

$$b_1 = 0.33$$

$$c_0 = Y_N = 0.33$$

$$c_k = Y_n^2 + \cos^2\left(\frac{(2k-1)\pi}{2N}\right)$$

$$c_1 = (0.33)^2 + \cos^2\left[\frac{(2-1)\pi}{2\times 3}\right] = (0.33)^2 + \cos^2\left[\frac{\pi}{6}\right] = (0.33)^2 + 0.75 = 0.8589$$

$$H_a(s) = \frac{20B_0}{s+(20\times 0.33)} \times \frac{20^2 B_1}{s^2 + (0.33)(20)s + \left((20)^2 \times 0.8589\right)}$$

$$H_a(s) = \frac{20B_0}{s+6.6} \times \frac{20^2 B_1}{s^2 + 6.6s + 343.56}$$

For N = odd, B_k can be calculated from

$$H_a(s)\big|_{s=0} = 1, \left[B_0 = B_1 = B_2 \cdots B_N\right]$$

$$\frac{20B_0}{6.6} \times \frac{20^2 B_1}{343.56} = 1$$

$$\frac{8000B_0^2}{2267.496} = 1$$

$$B_0^2 = 0.2834$$

$$B_0 = 0.532$$

$$B_0 = B_1 = 0.532$$

$$H_a(s) = \frac{20\times 0.532}{s+6.6} \times \frac{20^2 \times 0.532}{s^2 + 6.6s + 343.56} = \frac{10.64}{s+6.6} \cdot \frac{212.8}{s^2 + 6.6s + 343.56}$$

$$H_a(s) = \frac{2265.76}{(s+6.6)(s^2 + 6.6s + 343.56)}$$

Assume T =1 second

$$H(z) = H_a(s)\Big|_{s=\frac{2}{T}\frac{(z-1)}{(z+1)}}$$

$$= \frac{2265.76}{\left(\frac{2(z-1)}{z+1} + 6.6\right)\left[\left(\frac{2(z-1)}{z+1}\right)^2 + 6.6\times\frac{2(z-1)}{(z+1)} + 343.56\right]}$$

$$= \frac{2265.76}{\left(\frac{2(z-1)+6.6(z+1)}{(z+1)}\right)\left(\frac{4(z-1)^2+13.2(z-1)(z+1)+343.56(z+1)^2}{(z+1)^2}\right)}$$

$$= \frac{2265.76(z+1)^3}{[2(z-1)+6.6(z+1)]\left[4\left(z^2-2z+1\right)+13.2\left(z^2-1\right)+343.56\left(z^2+2z+1\right)\right]}$$

$$= \frac{2265.76(z+1)^2}{(8.6z+4.6)\left(4z^2-8z+2+13.2z^2-13.2+343.56z^2+687.12z+343.56\right)}$$

$$H(z) = \frac{2265.76(z+1)^2}{(8.6z+4.6)\left(360.76z^2+687.72z+332.36\right)}$$

Comparison between IIR and FIR Filters

IIR Filter	FIR Filter
All infinite samples of impulse response are considered.	Only N samples of impulse response are considered.
The impulse response cannot be directly converted to digital filter transfer function.	The impulse response can be directly converted to digital filter transfer function.
The design involves design of analog filter and then transforming analog filter to digital filter.	The digital filter can be directly designed to achieve the desired specification.
The specifications include the desired characteristics for magnitude response only.	The specifications include the desired characteristics for both magnitude and phase response.
Linear phase characteristics cannot be achieved.	Linear phase filter can be easily designed.

INVERSE CHEBYSHEV FILTERS 8.7

Inverse Chebyshev filters are also called Type-II Chebyshev filters. A low-pass inverse Chebyshev filter has a magnitude response given by

$$|H(j\Omega)| = \frac{\varepsilon C_N(\Omega_2/\Omega_c)}{[1+\varepsilon^2 C_N^2(\Omega_2/\Omega)]^{0.5}} \tag{8.61}$$

where ε is a constant and Ω_c is the 3 dB cut-off frequency. The Chebyshev polynomial, $C_N(x)$ is given by Eq. (8.53).

The magnitude response of the inverse Chebyshev filter is shown in Fig. 8.8. The magnitude response has maximally flat passband and equiripple stopband, just the opposite of the Chebyshev filter's response. That is why the Type-II Chebyshev filters are called the inverse Chebyshev filters.

The parameters of the inverse Chebyshev filter are obtained by considering the low-pass filter with the desired specifications as given below.

$$0.707 \le |H(j\Omega)| \le 1, \qquad 0 \le \Omega \le \Omega_c \tag{8.62a}$$

$$|H(j\Omega)| \le \delta_2, \qquad \Omega \ge \Omega_2 \tag{8.62b}$$

Using Eq. (8.61) in Eq. (8.62), we get

$$0.707^2 \le \frac{\varepsilon^2 C_N^2(\Omega_2/\Omega_c)}{1+\varepsilon^2 C_N^2(\Omega_2/\Omega)} \le 1, \quad 0 \le \Omega \le \Omega_c \tag{8.63a}$$

$$\frac{\varepsilon^2 C_N^2(\Omega_2/\Omega_c)}{1+\varepsilon^2 C_N^2(\Omega_2/\Omega)} \le \delta_2^2, \quad \Omega \ge \Omega_2 \tag{8.63b}$$

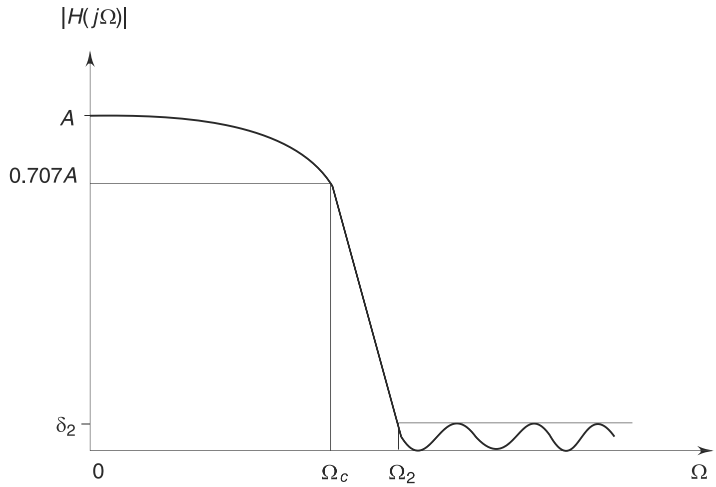


Fig. 8.8 *Magnitude Response of the Low-pass Inverse Chebyshev Filter*

When $\Omega = \Omega_2 = \Omega_c$, Eq. (8.63b) becomes

$$\delta_2^2 = \frac{\varepsilon^2}{1+\varepsilon^2}$$

Rearranging,

$$\varepsilon = \frac{\delta_2}{(1-\delta_2^2)^{0.5}} \tag{8.64}$$

When $\Omega = \Omega_c$, Eq. (8.63a) becomes

$$0.5 = \frac{\varepsilon^2 C_N^2(\Omega_2/\Omega_c)}{1+\varepsilon^2 C_N^2(\Omega_2/\Omega_c)}$$

$$0.5 + 0.5\varepsilon^2 C_N^2(\Omega_2/\Omega_c) = \varepsilon^2 C_N^2(\Omega_2/\Omega_c)$$

or simplifying,

$$C_N(\Omega_2/\Omega_c) = \frac{1}{\varepsilon} \tag{8.65}$$

Using Eq. (8.53),

$$\cosh\left[N \cosh^{-1}(\Omega_2/\Omega_c)\right] = \frac{1}{\varepsilon} \tag{8.66}$$

From Eqs. (8.66) and (8.64), we can get the order of the filter, N.

$$N = \frac{\cosh^{-1}(1/\varepsilon)}{\cosh^{-1}(\Omega_2/\Omega_c)} = \frac{\cosh^{-1}\left[\frac{1}{\delta_2^2} - 1\right]^{0.5}}{\cosh^{-1}(\Omega_2/\Omega_c)} \tag{8.67}$$

The value of N is chosen to be the nearest integer greater than the value given by Eq. (8.67).

ELLIPTIC FILTERS 8.8

The elliptic filter is sometimes called the Cauer filter. This filter has equiripple passband and stopband. Among the filter types discussed so far, for a given filter order, passband and stopband deviations, elliptic filters have the minimum transition bandwidth. The magnitude response of an odd ordered elliptic filter is shown in Fig. 8.9. The magnitude squared response is given by

$$\left|H(j\Omega)\right|^2 = \frac{1}{1+\varepsilon^2 U_N(\Omega/\Omega_c)} \tag{8.68}$$

where $U_N(x)$ is the Jacobian elliptic function of order N and ε is a constant related to the passband ripple.

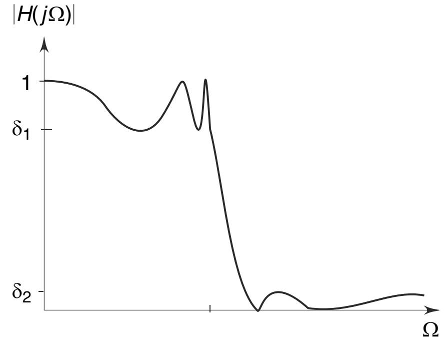


Fig. 8.9 *Magnitude Response of a Low-pass Elliptic Filter*

FREQUENCY TRANSFORMATION 8.9

There are basically four types of frequency selective filters, viz. low-pass, highpass, bandpass and bandstop. In the design techniques discussed so far we have considered only lowpass filters. This low-pass filter can be considered as a prototype filter and its system function can be obtained. Then, if a highpass or bandpass or bandstop filter is to be designed, it can be easily obtained by using frequency transformation. Frequency transformation can be accomplished in two ways. In the analog frequency transformation, the analog system function $H_p(s)$ of the prototype filter is converted into another analog system function $H(s)$ of the desired filter. Then using any of the mapping techniques, it is converted into the digital filter having a system function $H(z)$. In the digital frequency transformation, the analog prototype filter is first transformed to the digital domain, to have a system function $H_p(z)$. Then using frequency transformation, it can be converted into the desired digital filter.

8.9.1 Analog Frequency Transformation

The frequency transformation formulae used to convert a prototype lowpass filter into a lowpass (with a different cut-off frequency), highpass, bandpass or bandstop are given below.

(i) Low-pass with cut-off frequency Ω_c to low-pass with a new cut-off frequency Ω_c^*

$$s \to \frac{\Omega_c}{\Omega_c^*} s \tag{8.69}$$

Thus, if the system response of the prototype filter is $H_p(s)$, the system response of the new low-pass filter will be

$$H(s) = H_p\left(\frac{\Omega_c}{\Omega_c^*} s\right) \tag{8.70}$$

(ii) Low-pass with cut-off frequency Ω_c to high-pass with cut-off frequency Ω_c^*

$$s \to \frac{\Omega_c \Omega_c^*}{s} \tag{8.71}$$

The system function of the high-pass filter is then,

$$H(s) = H_p\left(\frac{\Omega_c \Omega_c^*}{s}\right) \tag{8.72}$$

(iii) Low-pass with cut-off frequency Ω_c to band-pass with lower cut-off frequency Ω_1 and higher cut-off frequency Ω_2

$$s \to \Omega_c \frac{s^2 + \Omega_1 \Omega_2}{s(\Omega_2 - \Omega_1)} \tag{8.73}$$

The system function of the high-pass filter is then

$$H(s) = H_p\left(\Omega_c \frac{s^2 + \Omega_1 \Omega_2}{s(\Omega_2 - \Omega_1)}\right) \tag{8.74}$$

(iv) Low-pass with cut-off frequency Ω_c to bandstop with lower cut-off frequency Ω_1 and higher cut-off frequency Ω_2

$$s \to \Omega_c \frac{s^2(\Omega_2 - \Omega_1)}{s^2 + \Omega_1 \Omega_2} \tag{8.75}$$

The system function of the bandstop filter is then,

$$H(s) = H_p\left(\Omega_c \frac{s(\Omega_2 - \Omega_1)}{s^2 + \Omega_1 \Omega_2}\right) \tag{8.76}$$

Table 8.1 *Analog frequency transformation*

Type	*Transformation*
Low-pass	$s \to \frac{\Omega_c}{\Omega_c^*} s$
High-pass	$s \to \frac{\Omega_c \Omega_c^*}{s}$
Bandpass	$s \to \Omega_c \frac{s^2 + \Omega_1 \Omega_2}{s(\Omega_2 - \Omega_1)}$
Bandstop	$s \to \Omega_c \frac{s(\Omega_2 - \Omega_1)}{s^2 + \Omega_1 \Omega_2}$

Example 8.32 *A prototype low-pass filter has the system response* $H(s) = \frac{1}{s^2 + 2s + 1}$. *Obtain a bandpass filter with* $\Omega_0 = 2$ *rad/s and* $Q = 10$. $\Omega_0^2 = \Omega_1 \cdot \Omega_2$ *and* $Q = \frac{\Omega_0}{\Omega_2 - \Omega_1}$.

Solution From Table 8.1, the required transformation is

$$s \to \Omega_c \frac{s^2 + \Omega_1\Omega_2}{s(\Omega_2 - \Omega_1)}, \text{ i.e.}$$

$$s = \Omega_c \frac{s^2 + \Omega_0^2}{s(\Omega_0 / Q)} = \Omega_c \frac{s^2 + 2^2}{s(2/10)} = 5\Omega_c \left(\frac{s^2 + 4}{s}\right)$$

Therefore,

$$H(s) = H(s)\Big|_{s = 5\Omega_c\left(\frac{s^2+4}{s}\right)}$$

$$H(s) = \frac{0.04s^2}{\Omega_c^2 s^4 + 0.4\Omega_c s^3 + (8\Omega_c^2 + 0.01)s^2 + 1.6\Omega_c s + 16\Omega_c^2}$$

Example 8.33 *Transform the prototype low-pass filter with system function*

$$H(s) = \frac{\Omega_c}{s + \Omega_c}$$

into a high-pass filter with cut-off frequency Ω_c^*.

Solution From Table 8.1, the desired transformation is,

$$s \to \frac{\Omega_c \Omega_c^*}{s}$$

Thus we have,

$$H_{hpf}(s) = \frac{\Omega_c}{\left(\frac{\Omega_c \Omega_c^*}{s}\right) + \Omega_c} = \frac{s}{s + \Omega_c^*}$$

Example 8.34 *Design an analog BPF to satisfy the following specifications:*

(*i*) 3 *dB upper and lower cut-off frequencies are* 100 *Hz and* 3.8 *kHz.*
(*ii*) *Stopband attenuation of* 20 *dB at* 20 *Hz and* 8 *kHz.*
(*iii*) *No ripple with both passband and stopband.*

Solution

Design of Butterworth filter (bandpass filter)

Given $\Omega_1 = 2\pi \times 20$ rad/sec, $\Omega_2 = 2\pi \times 8000$ rad/sec, $\Omega_l = 2\pi \times 100$ rad/sec,
$\Omega_u = 2\pi \times 3800$ rad/sec,
$\delta_1 = 3$ dB and $\delta_2 = 20$ dB

***To find Ha* (*s*)**

Step I: To find the ratio Ω_s/Ω_p

We know that

$$\frac{\Omega_s}{\Omega_p} = \min(|A|, |B|)$$

where

$$A = \frac{-\Omega_1^2 + \Omega_l\Omega_u}{\Omega_1(\Omega_u - \Omega_l)} = 5.129$$

$$B = \frac{-\Omega_2^2 + \Omega_l\Omega_u}{\Omega_2(\Omega_u - \Omega_l)} = 2.149$$

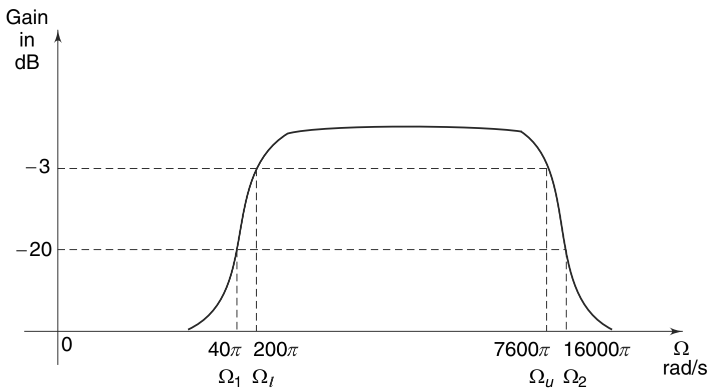


Fig. E8.34 *Butterworth BPF*

Therefore, $\dfrac{\Omega_s}{\Omega_p} = 2.149$

Step II: To find the order of filter (N):

$$N = \frac{\log_{10}\left(\dfrac{10^{0.1\delta_2}-1}{10^{0.1\delta_1}-1}\right)}{2\log_{10}\left(\dfrac{\Omega_s}{\Omega_p}\right)} = \frac{\log_{10}\left(\dfrac{10^2-1}{10^{0.3}-1}\right)}{2\log_{10}(2.149)} = 3.006$$

Therefore, the order of the filter is selected as $N = 4$.

Step III: Cut-off frequency Ω_c:

$$\Omega_c = 7600\pi \text{ rad/s}$$

Step IV: To find poles:

$$s_k = \Omega_c e^{j(\pi/2+(2k+1)\pi/2N)} \quad \text{where } k = 0, 1, \ldots, N-1$$

Here, $s_k = 7600 \quad e^{j(\pi/2+(2k+1)\pi/8)}, k = 0,1,2$

$$s_0 = -9136.99 + j22058.6$$
$$s_1 = -22058.6 + j9136.99$$
$$s_2 = -22058.6 - j9136.99$$
$$s_3 = -9136.99 - j22058.6$$

Step V: To find $H_a(s)$:

$$H_a(s) = \frac{1}{(s-s_0)(s-s_1)(s-s_2)(s-s_3)}$$

Here, $s_2 = s_1^*$ and $s_3 = s_0^*$

Therefore, $H_a(s) = \dfrac{1}{(s-s_0)(s-s_0^*)(s-s_1)(s-s_1^*)}$

Let $s_0 = a + jb$ and $s_1 = c + jd$

Therefore, $H_a(s) = \dfrac{1}{(s^2-2as+a^2+b^2)(s^2-2cs+c^2+d^2)}$

Here, $a = -9136.99$, $b = 22058.6$, $c = -22058.6$ and $d = 9136.99$

$$H_a(s) = \frac{1}{(s^2 - 2(-9136.99)s + 5.7\times10^8)(s^2 - 2(-22058.6)s + 5.7\times10^8)}$$

$$\frac{1}{(s^2 + 1.83\times10^4 s) + 5.7\times10^8)(s^2 + 4.41\times10^4 s + 5.7\times10^8)}$$

We know that

$$(s^2 + as + b)(s^2 + cs + b) = s^4 + s^3(a+c) + s^2(2b + ac) + sb(a+c) + b^2$$

Therefore,

$$H_a(s) = \frac{1}{s^4 + 6.2\times10^4 s^3 + 1.947\times10^9 s^2 + 3.56\times10^{13} s + 3.25\times10^{17}}$$

Step VI: To transform LPF to BPF:

$$s \Rightarrow \frac{s^2 + \Omega_l\Omega_u}{s(\Omega_u - \Omega_l)}$$

Here, $\Omega_2 = 200\,\pi$ rad/s

$\Omega_u = 7600\,\pi$ rad/s

$$s \Rightarrow \frac{s^2 + 1.5\times10^7}{s(2.32\times10^4)}$$

Let $x = 1.5 \times 10^7$ and $y = 2.32 \times 10^4$

Therefore, $$s \to \frac{s^2 + x}{sy}$$

$$H_a(s) = \frac{1}{s^4 + ms^3 + ns^2 + os + p}$$

where $m = 6.24 \times 10^4$

$n = 1.947 \times 10^9$

$o = 3.56 \times 10^{13}$

$p = 3.25 \times 10^{17}$

Therefore,

$$H_a(s) = \frac{1}{\left(\frac{s^2 + x}{sy}\right)^4 + m\left(\frac{s^2 + x}{sy}\right)^3 + n\left(\frac{s^2 + x}{sy}\right)^2 + o\left(\frac{s^2 + x}{sy}\right) + p}$$

$$H_a(s) = \frac{(sy)^4}{(s^2 + x)^4 + m(s^2 + x)^3 sy + ns^2y^2(s^2 + x)^2 + os^3y^3(s^2 + x) + ps^4y^4}$$

$$= \frac{s^4y^4}{(s^8 + 4s^6x + 4s^2x^3 + 6s^4x^2 + x^4) + msy(s^6 + 3s^4x + 3s^2x^2 + x^3) + ns^2y^2(s^4 + 2s^2x + x^2) + os^5y^3 + os^3y^3x + ps^4y^4}$$

$$= \frac{s^4y^4}{(s^8 + 4s^6x + 4s^2x^3 + 6s^4x^2 + x^4) + ms^7y + 3ms^5xy + 3ms^3yx^2 + msyx^3 + ns^6y^2 + 2ns^4xy^2 + ns^2y^2x^2 + os^3y^3x + ps^4y^4 + os^5y^3}$$

$$== \frac{s^4y^4}{s^8 + ms^7y + 4s^6x + ns^6y^2 + 3ms^5xy + os^5y^3 + 6s^4x^2 + 2ns^4xy^2 + ps^4y^4 + 3ms^3yx^2 + os^3y^3x + 4s^2x^3 + ns^2y^2x^2 + msyx^3 + x^4}$$

$$= \frac{s^4y^4}{s^8 + (my)s^7 + (4x + ny^2)s^6 + (3mxy + oy^3)s^5 + (6x^2 + 2nxy^2 + py^4)s^4 + (3myx^2 + oy^3x)s^3 + (4x^3 + ny^2x^2)s^2 + myx^3s + x^4}$$

$$H_a(s) = \frac{2.9\times10^{17}s^4}{s^8 + 1.45\times10^9 s^7 + 1.048\times10^{18}s^6 + 4.45\times10^{26}s^5 + 9.492\times10^{34}s^4 \; 6.67\times10^{33}s^3 + 2.358\times10^{32}s^2 + 4.89\times10^{30}s + 5.06\times10^{28}}$$

Example 8.35 *A discrete-time-second-order bandpass Butterworth filter is to be designed to the following specifications using bilinear transform method:*
(*a*) *Lower cut-off frequency* = 10 *Hz*
(*b*) *Upper cut-off frequency* = 20 *Hz*
(*c*) *Sampling frequency* = 100 *samples/s*

Solution To find $H(z)$ of the Butterworth filter:
Given $N = 2$

Step I: To find cutoff frequency Ω_c:

$$\Omega_c = 40\,\pi \text{ rad/s}$$

Step II: To find poles:

$$s_k = \Omega_c e^{j(\pi/2+(2k+1)\pi/2N)} \quad k = 0,1$$

i.e. $$s_k = 40\pi\, e^{j(\pi/2+(2k+1)\pi/4)} \quad k = 0,1$$

$$s_0 = -88.86 + j88.86$$

$$s_1 = -88.86 - j88.86$$

Step III: To find analog transfer function:

$$H_a(s) = \frac{1}{(s-s_0)(s-s_1)}$$

Here, $s_0 = s_1^*$

$$H_a(s) = \frac{1}{s^2 - 2as + a^2 + b^2}$$

$$H_a(s) = \frac{1}{s^2 + 177.72s + 1.58\times10^4}$$

Step IV: To transform LPF to BPF:

$$s \Rightarrow \frac{s^2 + \Omega_l \cdot \Omega_u}{s[\Omega_u - \Omega_l]}$$

Therefore, $$s \Rightarrow \frac{s^2 + 7895.7}{s[20\pi]}$$

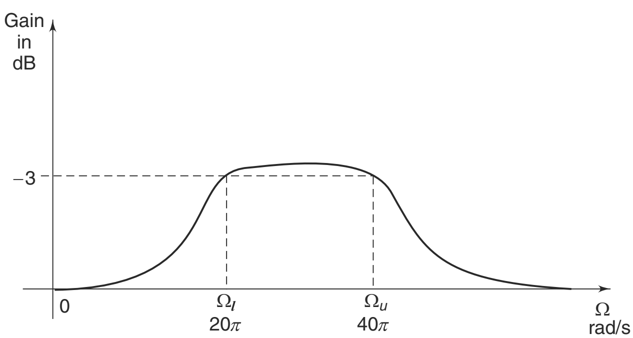


Fig. E8.35 *Butterworth BPF*

Let $x = 7895.7$ and $y = 20\pi$.

Therefore, $s \Rightarrow \dfrac{s^2 + x}{y}$

$$H_a(s) = \frac{1}{\left(\dfrac{s^2+x}{sy}\right)^2 + 177.72\left(\dfrac{s^2+x}{sy}\right) + 1.58\times 10^4}$$

$$= \frac{1}{\dfrac{s^4+2xs^2+x^2}{s^2y^2} + 177.72\dfrac{(s^2+x)}{sy} + 1.58\times 10^4}$$

$$= \frac{s^2y^2}{s^4+2xs^2+x^2+sy(s^2+x)177.72+1.58\times 10^4(s^2y^2)}$$

$$= \frac{s^2y^2}{s^4+2xs^2+x^2+s^3y177.72+syx177.72+s^2y^2 1.58\times 10^4}$$

$$= \frac{s^2y^2}{s^4+y177.72s^3+s^2(2x)+yx177.72s+s^2y^2 1.58\times 10^4+x^2}$$

$$= \frac{s^2y^2}{\begin{array}{l}s^4+(20\pi)177.72s^3+s^2[(2\times 7895.7)+(20\pi)^2\times 1.58\times 10^4] \\ +20\pi\times 7895.7\times 177.2s+(7895.7)^2\end{array}}$$

$$H_a(s) = \frac{3947.8s^2}{s^4+(1.12\times 10^4)s^3+(6.24\times 10^7)s^2+8.82\times 10^7 s+6.23\times 10^7}$$

Step V: To convert analog transfer function to digital transfer function using bilinear transformation:

$$\Omega = \frac{2}{T}\tan(\omega/2)$$

$$s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}$$

where $\dfrac{1}{T} = f_s = 100$ Hz

Therefore, $s = 2f_s\dfrac{1-z^{-1}}{1+z^{-1}} = 200\dfrac{1-z^{-1}}{1+z^{-1}}$

Let $a = 200$

$$s = a\frac{1-z^{-1}}{1+z^{-1}}$$

Therefore,

$$H(z)=\frac{3947.8\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)^2}{\left\{\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)^4+1.12\times10^4\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)^3+6.24\times10^7\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)^2+8.82\times10^7\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)+6.23\times10^7\right\}}$$

$$=\frac{m\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)^2}{\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)^4+n\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)^3+o\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)^2+\left(a\cdot\frac{1-z^{-1}}{1+z^{-1}}\right)p+q}$$

Let $m = 3947.8$, $n = 1.12 \times 10^4$, $o = 6.24 \times 10^7$, $p = 8.82 \times 10^7$, $q = 6.23 \times 10^7$

Therefore,

$$H(z)=\frac{ma^2(1-z^{-1})^2(1+z^{-1})^2}{a^4(1-z^{-1})^4+na^3(1-z^{-1})^3(1+z^{-1})+oa^2(1-z^{-1})^2(1+z^{-1})^2+pa(1-z^{-1})(1+z^{-1})^3+q(1+z^{-1})^4}$$

$$=\frac{ma^2(1-z^{-2})(1-z^{-2})}{a^4[1-4z^{-1}+6z^{-2}-4z^{-3}+z^{-4}]+na^3[1-3z^{-1}+3z^{-2}-z^{-3}](1+z^{-1})+oa^2[1-2z^{-1}+z^{-2}][1+2z^{-1}+z^{-2}]+pa[1+3z^{-2}+3z^{-1}+z^{-3}](1-z^{-1})+q[1+4z^{-1}+6z^{-2}-4z^{-3}+z^{-3}]}$$

$$=\frac{ma^2[1-2z^{-2}+z^{-4}]}{z^{-4}[a^4-na^3+oa^2-pa+q]+z^{-3}[-4a^4+2na^3-2pa+4q]+z^{-2}[6a^4-2oa^2+6q]z^{-1}[-4a^4-2na^3+2pa+4q]+[a^4+na^3+oa^2+pa+q]}$$

Substituting the values of *m, n, o, p* and *q*, we get

$$H(z)=\frac{1.58\times10^8(1-2z^{-1}+z^{-4})}{(2.4\times10^{12})z^{-4}+(1.13\times10^{11})z^{-3}-(4.98\times10^{12})z^{-2}-(1.5\times10^{11})z^{-1}+2.6\times10^{12}}$$

8.9.2 Digital Frequency Transformation

As in the analog domain, frequency transformation is possible in the digital domain also. The frequency transformation is done in the digital domain by replacing the variable z^{-1} by a function of z^{-1}, i.e. $f(z^{-1})$. This mapping must take into account the stability criterion. All the poles lying within the unit circle must map onto itself and the unit circle must also map onto itself. For the unit circle to map onto itself, the implication is that for $r = 1$,

$$e^{-j\omega} = f(e^{-j\omega})$$
$$= \left|f(e^{-j\omega})\right| e^{j\arg\left[f(e^{j\omega})\right]}$$

Hence, we must have $|f(e^{-j\omega})| = 1$ for all frequencies. So, the mapping is that of an all pass filter and of the form

$$f(z^{-1}) = \pm\prod_{k=1}^{n} \frac{z^{-1} - a_k}{1 - a_k z^{-1}} \tag{8.77}$$

To get a stable filter from the stable prototype filter, we must have $|a_k| < 1$. The transformation formulae can be obtained from Eq. (8.77) for converting the prototype low-pass digital filter into a digital low-pass, high-pass, bandpass or bandstop filter. Table 8.2 gives these transformations.

The frequency transformation may be accomplished in any of the available two techniques, however, caution must be taken to which technique to use. For example, the impulse invariant transformation is not suitable for high-pass or bandpass filters whose resonant frequencies are higher. In such a case, suppose a low-pass prototype filter is converted into a high-pass filter using ***analog frequency transformation*** and transformed later to a digital filter using impulse invariant technique. This will result in alaising problems. However, if the same prototype low-pass filter is first transformed into a digital filter using impulse-invariant technique and later converted into a high-pass filter using ***digital***

Table 8.2 *Digital frequency transformation*

Type	*Transformation*	*Design Parameter*
Low–pass	$z^{-1} \to \dfrac{z^{-1} - a}{1 - a z^{-1}}$	$a = \dfrac{\sin[(\omega_c - \omega_c^*)/2]}{\sin[(\omega_c + \omega_c^*)/2]}$
High-pass	$z^{-1} \to -\dfrac{z^{-1} + a}{1 + a z^{-1}}$	$a = \dfrac{\cos[(\omega_c - \omega_c^*)/2]}{\cos[(\omega_c + \omega_c^*)/2]}$
Bandpass	$z^{-1} \to -\dfrac{z^{-2} - a_1 z^{-1} + a^2}{a_2 z^{-2} - a_1 z^{-1} + 1}$	$a_1 = -2\alpha K/(K+1)$ $a_2 = (K-1)/(K+1)$ $\alpha = \dfrac{\cos[(\omega_2 + \omega_1)/2]}{\cos[(\omega_2 - \omega_1)/2]}$ $K = \cot\left(\dfrac{\omega_2 - \omega_1}{2}\right)\tan\left(\dfrac{\omega_c}{2}\right)$
Bandstop	$z^{-1} \to -\dfrac{z^{-2} - a_1 z^{-1} + a_2}{a_2 z^{-2} - a_1 z^{-1} + 1}$	$a_1 = -2\alpha/(K+1)$ $a_2 = (1-K)/(1+K)$ $\alpha = \dfrac{\cos[(\omega_2 + \omega_1)/2]}{\cos[(\omega_2 - \omega_1)/2]}$ $K = \tan\left(\dfrac{\omega_2 - \omega_1}{2}\right)\tan\left(\dfrac{\omega_c}{2}\right)$

frequency transformation, it will not have any alaising problem. Whenever the bilinear transformation is used, it is of no significance whether analog frequency transformation is used or digital frequency transformation. In this case, both analog and digital frequency transformation techniques will give the same result.

REVIEW QUESTIONS

8.1 What is an IIR filter? Compare its characteristics with an FIR filter.

8.2 What are the requirements for converting a stable analog filter into a stable digital filter?

8.3 What are the different design techniques available for IIR filters?

8.4 Obtain the mapping formula for the approximation of derivatives method using backward difference.

8.5 Comment on the stability of backward difference approximation for the derivative method of transformation.

8.6 Why is the forward difference formula for the approximation of derivatives not used?

8.7 Use the backward difference for the derivative to convert the analog low-pass filter with the following system function, using impulse invariant transformation

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

8.8 Transform the analog filter with the transfer function shown below into a digital filter, using backward difference for the derivative.

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

8.9 What is the impulse invariant technique?

8.10 Discuss the stability of the impulse invariant mapping technique.

8.11 Obtain the mapping formula for the impulse invariant transformation.

8.12 An analog filter has the following system function. Convert this filter into a digital filter using the impulse invariant technique.

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

8.13 Convert the analog filter to digital filter whose system function is

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

8.14 Convert the analog filter to digital filter whose system function is

$$H(s) = \frac{36}{(s+0.1)^2+36}$$

The digital filter should have a resonant frequency of $\omega_r = 0.2\pi$.
Use impulse invariant mapping.

8.15 What is bilinear transformation?

8.16 Compare bilinear transformation with other transformations based on their stability.

8.17 Obtain the transformation formula for the bilinear transformation.

8.18 An analog filter has the following system function. Convert this filter into a digital filter using bilinear transformation.

$$H(s) = \frac{1}{(s+0.2)^2 + 16}$$

8.19 Convert the analog filter to a digital filter whose system function is

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

using bilinear transformation.

8.20 Convert the analog filter to a digital filter whose system function is

$$H(s) = \frac{36}{(s+0.1)^2 + 36}$$

The digital filter should have a resonant frequency of $\omega_r = 0.2\pi$.
Use bilinear transformation.

8.21 What is meant by frequency warping? What is the cause of this effect?

8.22 Describe Butterworth filters?

8.23 Comment on the passband and stopband characteristics of Butterworth filters.

8.24 Describe Chebyshev filters?

8.25 Describe inverse Chebyshev filters?

8.26 Describe elliptic filters?

8.27 Compare the passband and stopband characteristics of the major types of analog filters.

8.28 Why is frequency transformation needed?

8.29 What are the different types of frequency transformations?

8.30 Design a digital Chebyshev filter to meet the constraint

$$0.8 \le |H(e^{j\omega})| \le 1, \qquad 0 \le \omega \le 0.2\pi$$
$$|H(e^{j\omega})| \le 0.2, \qquad 0.6\pi \le \omega \le \pi$$

using (i) bilinear transformation and (ii) impulse invariant transformation.

8.31 Design a digital Butterworth filter to meet the constraint

$$0.8 \le |H(e^{j\omega})| \le 1, \qquad 0 \le \omega \le 0.2\pi$$
$$|H(e^{j\omega})| \le 0.2, \qquad 0.26\pi \le \omega \le \pi$$

using (i) bilinear transformation and (ii) impulse invariant transformation.

8.32 Design a digital Butterworth filter to meet the constraint

$$0.9 \le |H(e^{j\omega})| \le 1, \qquad 0 \le \omega \le 0.25\pi$$
$$|H(e^{j\omega})| \le 0.2, \qquad 0.6\pi \le \omega \le \pi$$

using (i) bilinear transformation and (ii) impulse invariant transformation.

8.33 Design and realise a digital LPF using bilinear transformation to satisfy the following requirements
(a) monotonic stopband and passband
(b) −3 dB cut-off frequency at 0.6π radians, and
(c) magnitude down at 15 dB at 0.75π radians.

8.34 Determine the normalised low-pass Butterworth analog poles for $N = 10$.

8.35 Determine the normalised Chebyshev analog low-pass poles for $N = 6$.

8.36 Mention the advantages and disadvantages of FIR and IIR filters.

Chapter 9 Realisation of Digital Linear Systems

INTRODUCTION 9.1

For the design of digital filters, the system function $H(z)$ or the impulse response $h(n)$ must be specified. Then the digital filter structure can be implemented or synthesised in hardware/software form by its difference equation obtained directly from $H(z)$ or $h(n)$. Each difference equation or computational algorithm can be implemented by using a digital computer or special purpose digital hardware or special programmable integrated circuit.

In order to implement the specified difference equation of the system, the required basic operations are addition, delay and multiplication by a constant. A chosen structure determines a computational algorithm. Generally, different structures give different results. The most common methods for realising digital linear systems are direct, cascade and parallel forms, and state variable realisations.

In this chapter, direct-forms, cascade and parallel forms, ladder form and state variable realisations of IIR filters, direct-form and cascade realisations of FIR filters, and realisations of linear phase FIR systems are discussed.

BASIC REALISATION BLOCK DIAGRAM AND THE SIGNAL-FLOW GRAPH 9.2

The output of a finite order linear time invariant system at time n can be expressed as a linear combination of the inputs and outputs,

$$y(n) = \sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \tag{9.1}$$

where a_k and b_k are constants with $a_0 \neq 0$ and $M \leq N$.

The current output $y(n)$ is equal to the sum of past outputs, form $y(n-1)$ to $y(n-N)$, which are scaled by the *delay-dependent feedback coefficient* a_k, plus the sum of future, present and past inputs, which are scaled by the *delay-dependent feed forward coefficient* b_k.

Taking the z-transform of the output sequence $y(n)$ given in Eq. (9.1), we get

$$Y(z) = \sum_{n=-\infty}^{\infty} y(n) z^{-n}$$

$$= \sum_{n=-\infty}^{\infty} \left(\sum_{k=1}^{N} a_k\, y(n-k + \sum_{k=0}^{M} b_k\, x(n-k) \right) z^{-n}$$

Changing the order of summation

$$Y(z)=\sum_{k=1}^{N}a_k\left\{\sum_{n=-\infty}^{\infty}y(n-k)z^{-n}\right\}+\sum_{k=0}^{M}b_k\left\{\sum_{n=-\infty}^{\infty}x(n-k)z^{-n}\right\}$$

$$=\sum_{k=1}^{N}a_k z^{-k}\,Y(z)+\sum_{k=0}^{M}b_k z^{-k}\,X(z)$$

$$Y(z)\left\{1-\sum_{k=1}^{N}a_k z^{-k}\right\}=\sum_{k=0}^{M}b_k z^{-k}\,X(z)$$

Therefore,

$$H(z)=\frac{Y(z)}{X(z)}=\frac{\sum_{k=0}^{M}b_k z^{-k}}{1-\sum_{k=1}^{N}a_k z^{-k}} \qquad (9.2)$$

The computational algorithm of an LTI digital filter can be conveniently represented as a block diagram using basic building blocks representing the unit delay (z^{-1}) or storage element, the multiplier and the adder. These basic building blocks are shown in Fig. 9.1.

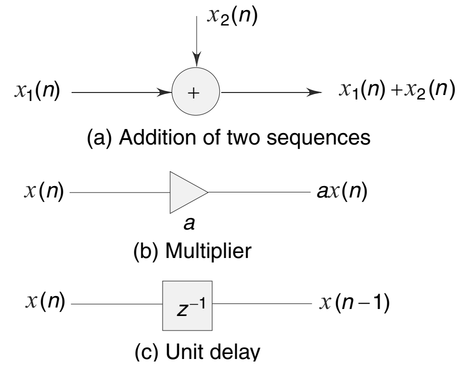


Fig. 9.1 *Basic Building Blocks*

For example, the difference equation of the first-order digital system may be written as

$$Y(n) = a_1 y(n-1) + x(n) + b_1 x(n-1)$$

The basic realisation block diagram for this equation and the corresponding structure of the signal flow graph are shown in Figs. 9.2 (a) and (b). Here, it is clear that there is direct correspondence between branches in the digital realisation structure and branches in the signal flow graph. But in the signal flow graph, the nodes represent both branch points and adders in the digital realisation block diagram.

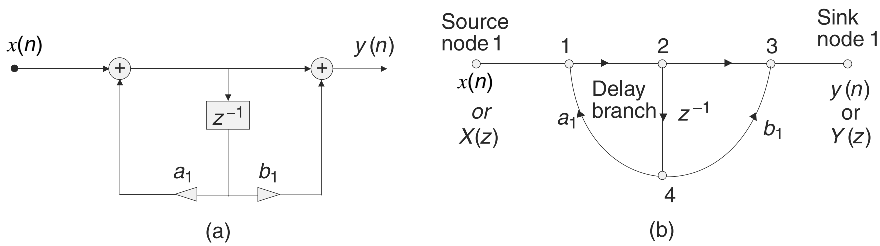


Fig. 9.2 *(a) Basic Realisation Block Diagram Representing a First-order Digital System and (b) Its Corresponding Signal Flow Graph*

Advantages of Representing the Digital System in Block Diagram Form

(i) Just by inspection, the computation algorithm can be easily written
(ii) The hardware requirements can be easily determined
(iii) A variety of equivalent block diagram representations can be easily developed from the transfer function
(iv) The relationship between the output and the input can be determined.

9.2.1 Canonic and Non-Canonic Structures

If the number of delays in the realisation block diagram is equal to the order of the difference equation or the order of the transfer function of a digital filter, then the realisation structure is called **canonic**. Otherwise, it is a **non-canonic** structure.

BASIC STRUCTURES FOR IIR SYSTEMS 9.3

Causal IIR systems are characterised by the constant coefficient difference equation of Eq. (9.1) or equivalently, by the real rational transfer function of Eq. (9.2). From these equations, it can be seen that the realisation of infinite duration impulse response (IIR) systems involves a recursive computational algorithm. In this section, the most important filter structures namely direct Forms I and II, cascade and parallel realisations for IIR systems are discussed.

9.3.1 Direct Form Realisation of IIR Systems

Equation 9.2 is the standard form of the system transfer function. By inspection of this equation, the block diagram representation can be drawn directly for the **direct form realisation**. The multipliers in the feed forward paths are the numerator coefficients and the multipliers in the feedback paths are the negatives of the denominator coefficients. Since the multiplier coefficients in the structures are exactly the coefficients of the transfer function, they are called direct form structures.

Direct Form I

The digital system structure determined directly from either Eq. (9.1) or Eq. (9.2) is called the direct form I. In this case, the system function is divided into two parts connected in cascade, the first part containing only the zeros, followed by the part containing only the poles. An intermediate sequence $w(n)$ is introduced. A possible IIR system direct form I realisation is shown in Fig. 9.3, in which $w(n)$ represents the output of the first part and input to the second.

$$w(n) = \sum_{k=0}^{M} b_k\, x(n-k)$$

and

$$y(n) = \sum_{k=1}^{N} a_k\, y(n-k) + w(n)$$

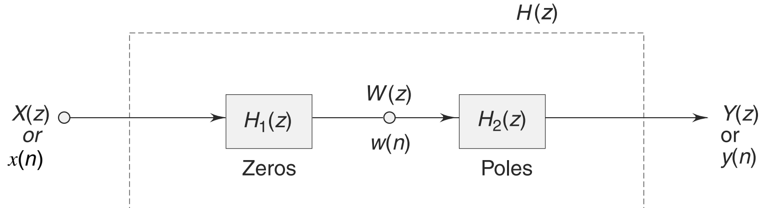


Fig. 9.3 *A Possible IIR System Direct-Form I Realisation*

Taking z-transform, we get

$$W(z) = X(z)\sum_{k=0}^{M} b_k\, z^{-k} \quad \text{and} \quad Y(z) = Y(z)\sum_{k=1}^{N} a_k z^{-k} + W(z)$$

Therefore,

$$Y(z) = \frac{W(z)}{1 - \sum_{k=1}^{N} a_k\, z^{-k}}$$

The direct form I realisation is shown in Fig. 9.4. The direct form I realisation requires $M + N$ storage elements. For our convenience, we have assumed that $M = N$, for drawing the network.

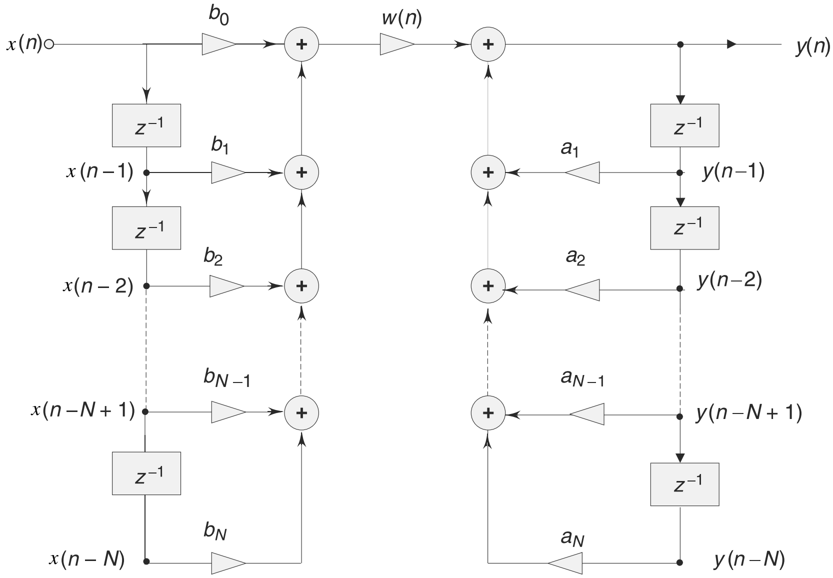


Fig. 9.4 *Direct-Form I Realisation of an N^{th} Order Difference*

Example 9.1 *Develop a direct form I realisation of the difference equation*

$y(n) = b_0x(n) + b_1x(n - 1) + b_2x(n - 2) + b_3\,x(n - 3) - a_1\,y(n - 1) - a_2\,y(n - 2) - a_3\,y(n - 3)$

Solution Taking z-transform of the given difference equation and simplifying, we get the transfer function

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + b_3z^{-3}}{1 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3}}$$

The corresponding direct form I realisation structure is illustrated in Fig. E9.1.

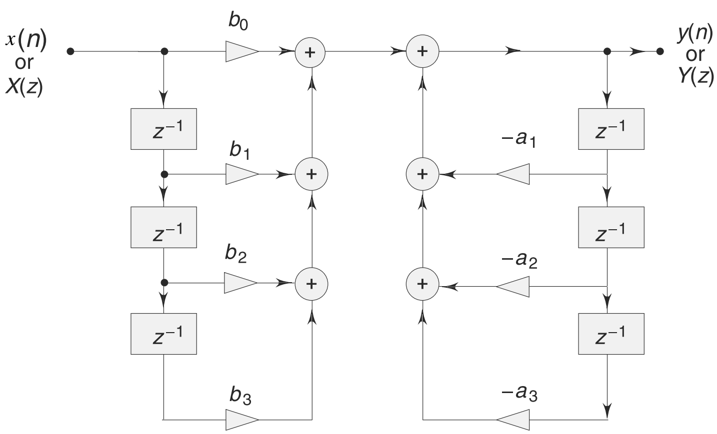


Fig. E9.1

Direct Form II

Since we are dealing with linear systems, the order of these parts can be interchanged. This property yields a second direct form realisation. In direct Form II, the poles of $H(z)$ are realised first and the zeros second. Here, the transfer function $H(z)$ is broken into a product of two transfer functions $H_1(z)$ and $H_2(z)$, where $H_1(z)$ has only poles and $H_2(z)$ contains only the zeros as given below:

$$H(z) = H_1(z).H_2(z)$$

where

$$H_1(z) = \frac{1}{\left(1 - \sum_{k=1}^{N} a_k z^{-k}\right)} \quad \text{and} \quad H_2(z) = \sum_{k=0}^{M} b_k z^{-k}$$

The decomposition for direct Form II realisation is shown in Fig. 9.5. An intermediate sequence $w(n)$ is introduced to obtain the output of the filter.

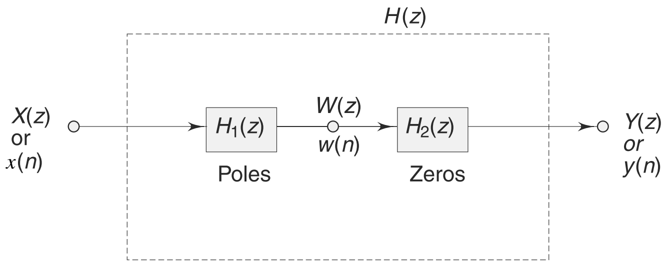


Fig. 9.5 *The Decomposition for Direct Form II Realisation*

$$w(n) = \sum_{k=1}^{N} a_k w(n-k) + x(n) \quad \text{and} \quad y(n) = \sum_{k=0}^{M} b_k w(n-k)$$

Taking z-transform of the above equations, we get

$$W(z) = \frac{X(z)}{1 - \sum_{k=1}^{M} a_k z^{-k}} \quad \text{and} \quad Y(z) = W(z) \sum_{k=0}^{M} b_k z^{-k}$$

The direct Form II realisation requires only the larger of M or N storage elements. When compared to direct Form I realisation, the direct Form II uses the minimum number of storage elements and hence said to be a canonic structure. However, when the addition is performed sequentially, the direct Form II needs two adders instead of one adder required for the direct Form I.

The Direct Form II realisation network structures are shown in Figs. 9.6 and 9.7.

Though the direct Form I and II are commonly employed, they have two drawbacks, viz. (i) they lack hardware flexibility and (ii) due to finite precision arithmetic, as to be discussed in Chapter 10, the sensitivity of the coefficients to quantisation effects increases with the order of the filter. This sensitivity may change the coefficient values and hence the frequency response, thereby causing the filter to become unstable. To overcome these effects, the cascade and parallel realisations can be implemented.

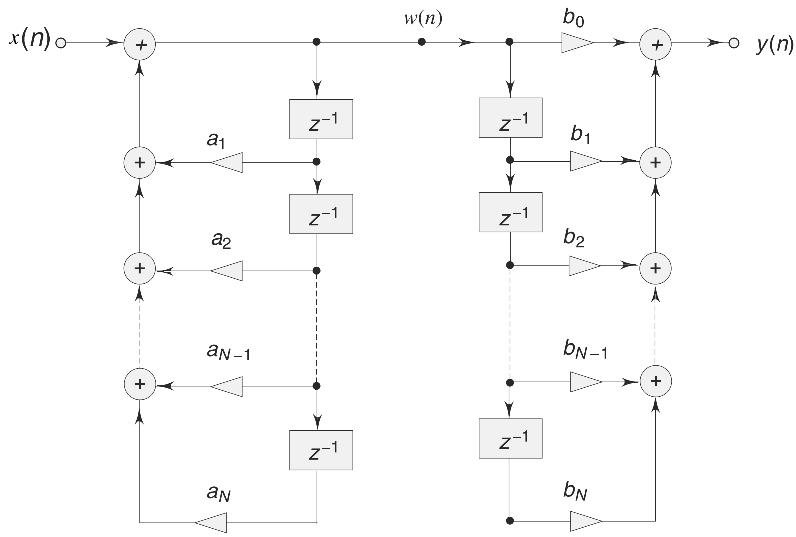


Fig. 9.6 *Direct Form II—Realisation Network of Fig. 9.4 with Poles and Zeros Interchanged*

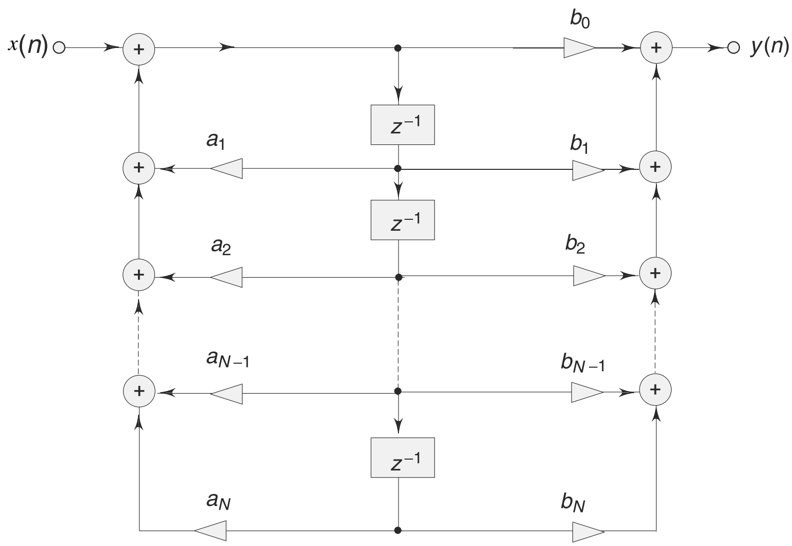


Fig. 9.7 *Direct Form II—Realisation Network of Fig. 9.6 with Combined Common Delays*

Example 9.2 *Determine the direct Forms I and II realisations for a third-order IIR transfer function.*

$$H(z) = \frac{0.28z^2 + 0.319z + 0.04}{0.5z^3 + 0.3z^2 + 0.17z - 0.2}$$

Solution Multiplying the transfer function numerator and denominator by $2z^{-3}$, we obtain the standard form of the transfer function.

$$H(z) = \frac{0.56z^{-1} + 0.638z^{-2} + 0.08z^{-3}}{1 + 0.6z^{-1} + 0.34z^{-2} - 0.4z^{-3}}$$

The direct Forms I and II realisations of the above transfer function are shown in Fig. E9.2(a) and (b) respectively.

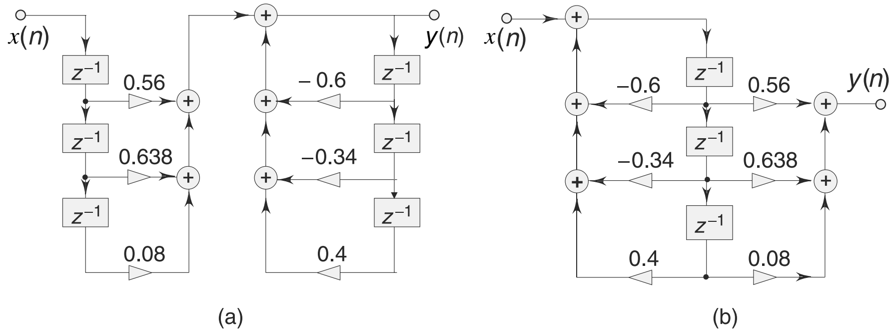


Fig. E9.2

Example 9.3 *Determine the direct Forms I and II for the second-order filter given by* $y(n) = 2b \cos \omega_0 y(n-1) - b^2 y(n-2) + x(n) - b \cos \omega_0 x(n-1)$

Solution Taking z-transform for the given function, we get

$$Y(z) = 2b\cos\omega_0 z^{-1}Y(z) - b^2 z^{-2}Y(z) + X(z) - b\cos\omega_0 z^{-1}X(z)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 - b\cos\omega_0 z^{-1}}{1 - 2b\cos\omega_0 z^{-1} + b^2 z^{-2}}$$

Direct Form I

$$H(z) = H_1(z)\, H_2(z)$$

Therefore, $Y(z) = H_1(z)\, H_2(z) X(z)$

In this form, the intermediate sequence $w(n)$ is introduced between $H_1(n)$ and $H_2(n)$

Let $H_1(z) = 1 - b\cos\omega_0 z^{-1} = \frac{W(z)}{X(z)}$

Therefore, $X(z)\,(1 - b\cos\omega_0 z^{-1}) = W(z)$

$$x(n) - b\cos\omega_0\, x(n-1) = w(n)$$

and

$$H_2(z) = (1 - 2b\cos\omega_0 z^{-1} + b^2 z^{-2})^{-1} = \frac{Y(z)}{X(z)}$$

$$Y(z) = \frac{W(z)}{(1 - 2b\cos\omega_0 z^{-1} + b^2 z^{-2})}$$

$$Y(z)[1 - 2b\cos\omega_0 z^{-1} + b^2 z^{-2}] = W(z)$$

The inverse z-transform of this function is

$$y(n) - 2\, b\cos\omega_0\, y(n-1) + b^2\, y(n-2) = w(n)$$

The direct Form I realisation structure of the above function is shown in Fig. E9.3(a).

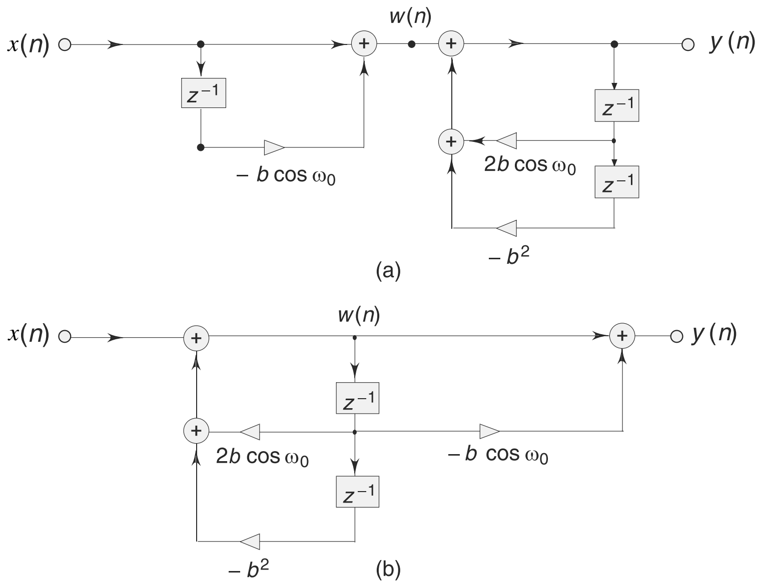


Fig. E9.3

Direct Form II

$$Y(z) = H_2(z)\, H_1(z)\, X(z)$$

$$H_2(z) = (1 - 2b\cos\omega_0 z^{-1} + b^2 z^{-2})^{-1} = \frac{U(z)}{X(z)}$$

Let $\quad H_1(z) = 1 - b\cos\omega_0 z^{-1} = \dfrac{Y(z)}{U(z)}$

$$X(z) = U(z)\{1 - 2\ b\cos\omega_0 z^{-1} + b^2 z^{-2}\}$$

Hence, $\quad x(n) = u(n) - 2b\cos\omega_0 u(n-1) + b^2 u(n-2)$

$$Y(z) = U(z)\{1 - b\cos\omega_0 z^{-1}\}$$

Hence, $\quad y(n) = u(n) - b\cos\omega_0 u(n-1)$

The Direct Form II realisation structure of the above function is shown in Fig. 9.3 (b).

9.3.2 Cascade Realisation of IIR Systems

In cascade realisation, the transfer function $H(z)$ is broken into a product of transfer functions $H_1(z)$, $H_2(z)$,..., $H_k(z)$. Factoring the numerator and denominator polynomials of the transfer function $H(z)$, we obtain

$$H(z) = G \cdot \frac{\prod_{k=1}^{M_1}\left(1 - g_k z^{-1}\right) \prod_{k=1}^{M_2}\left(1 - h_k z^{-1}\right)\left(1 - h_k^* z^{-1}\right)}{\prod_{k=1}^{N_1}\left(1 - c_k z^{-1}\right) \prod_{k=1}^{N_1}\left(1 - d_k z^{-1}\right)\left(1 - d_k^* z^{-1}\right)} \tag{9.3}$$

where $M = M_1 + 2M_2$, $N = N_1 + 2N_2$ and the constant G is a gain term. In the above equation, the first-order factors represent real zeros at g_k and real poles at c_k and the second-order factors represent complex conjugate zeros at h_k and h_k^*, and complex conjugate poles at d_k and d_k^*.

Generally, the numerator and denominator polynomials of the transfer function $H(z)$ are factored into a product of first-order and second-order polynomials. Here $H(z)$ can be expressed as

$$H(z) = G \prod_{k=1}^{[(N+1)/2]} \frac{1+\beta_{1k} z^{-1} + \beta_{2k} z^{-2}}{1-\alpha_{1k} z^{-1} - \alpha_{2k} z^{-2}} \tag{9.4}$$

where $[(N + 1)/2]$ means the largest integer contained in $(N + 1)/2$. For a first-order factor, $\omega_{2k} = \beta_{2k} = 0$. Here, it is assumed that $M \le N$ and the real poles and the real zeros are combined in pairs. For the filter coefficients to be real, two complex conjugate zeros of the system function $H(z)$ are combined for each numerator term and two complex conjugate poles are combined for each denominator term. Two real-valued singularities can also be combined to form a second-order term as well. If the number of poles or zeros is odd, the remaining single singularity is implemented by making $\beta_{2k} = 0$ or $\alpha_{2k} = 0$ for one of the sections.

There are several cascade realisations for the factored form, based on pole-zero pairings and ordering. A possible realisation of a third-order transfer function is shown in Fig. 9.8.

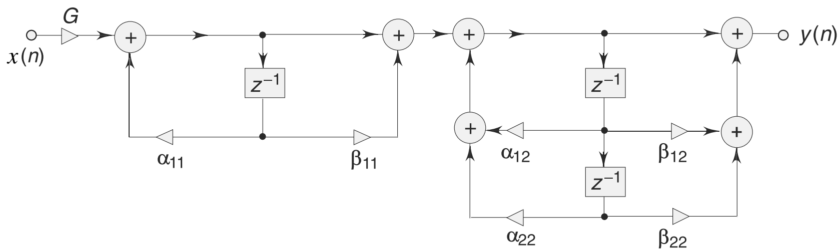


Fig. 9.8 *Cascade Structure with Direct Form II Realisation of a Third-Order IIR Transfer Function*

The advantage of the cascade form is that the overall transfer function of the filter can be determined, and also it has the same zeros and poles as the individual components, since the transfer function has the product of the components.

Example 9.4 *Obtain a cascade realisation of the system characterised by the transfer function*

$$H(z) = \frac{2(z+2)}{z(z-0.1)(z+0.5)(z+0.4)}$$

Solution Multiplying the transfer function numerator and denominator by z^{-4}, we obtain the standard form of the transfer function given by

$$H(z) = \frac{2z^{-3}(1+2z^{-1})}{(1-0.1z^{-1})(1+0.5z^{-1})(1+0.4z^{-1})}$$

$$= 2.z^{-3} \cdot \frac{(1+2z^{-1})}{(1-0.1z^{-1})} \cdot \frac{1}{(1+0.5z^{-1})} \cdot \frac{1}{(1+0.4z^{-1})}$$

$$= 2 \cdot H_1(z) \cdot H_2(z) \cdot H_3(z) \cdot H_4(z)$$

The cascade realisation of the system transfer function is shown in Fig. E9.4.

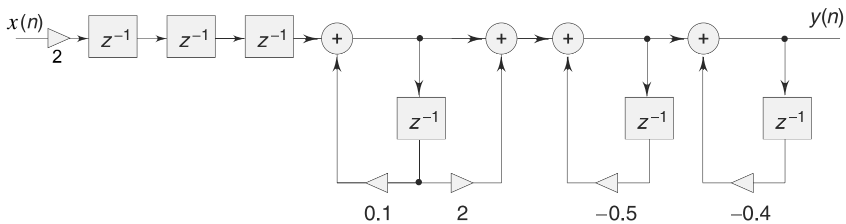


Fig. E9.4

9.3.3 Parallel Realisation of IIR Systems

By using the partial fraction expansion, the transfer function of an IIR system can be realised in a parallel form. A partial fraction expansion of the transfer function in the form given below will lead to the parallel form.

$$H(z)=\sum_{k=1}^{N_1}\frac{A_k}{1-c_k z^{-1}}+\sum_{k=1}^{N_2}\frac{B_k(1-e_k z^{-1})}{(1-d_k z^{-1})(1-d_k^{*} z^{-1})}+\sum_{k=0}^{M-N}C_k z^{-k} \tag{9.5}$$

The quantities A_k, B_k, C_k, c_k and e_k are real when the coefficients a_k and b_k of Eq. (9.2) are real. If $M < N$, then the last term $\sum_{k=0}^{M-N} C_k z^{-k}$ will become zero.

Thus, assuming simple poles, $H(z)$ is expressed in the form

$$H(z)=\gamma_0+\sum_{k=1}^{(N+1)/2}\frac{\gamma_{0k}+\gamma_{1k}z^{-1}}{1-\alpha_{1k}z^{-1}-\alpha_{2k}z^{-2}} \tag{9.6}$$

For a real pole, $\gamma_{1k} = \alpha_{2k} = 0$. As an example, the parallel form realisation with $M = N$ is illustrated in Fig. 9.9.

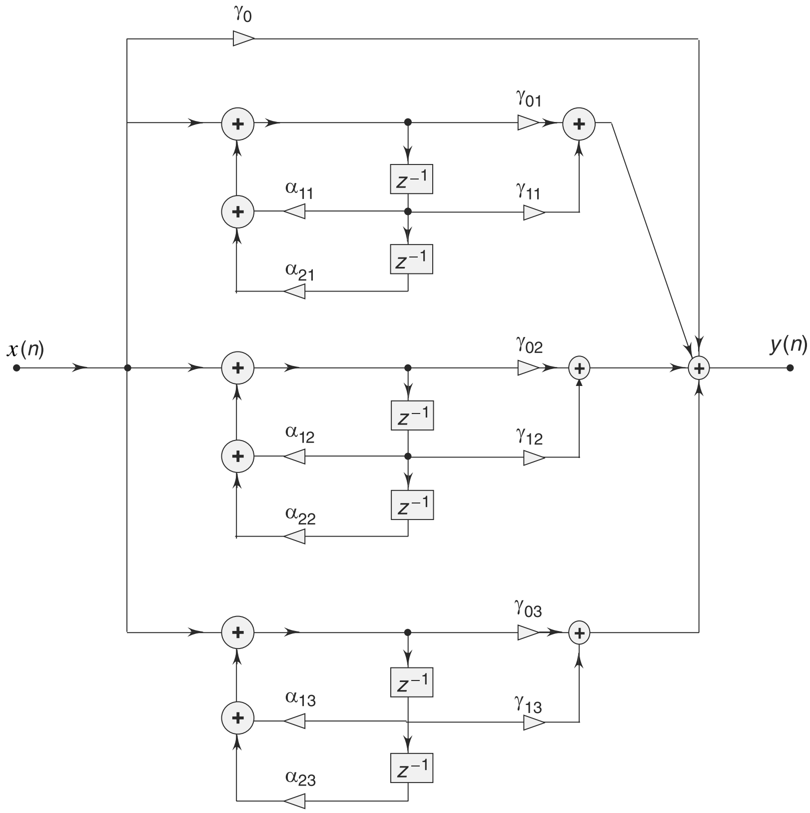


Fig. 9.9 *Parallel Form Realisation Structure with the Real and Complex Poles Grouped in Pairs*

The parallel realisation is useful for high speed filtering applications since the filter operation is performed in parallel, i.e. the processing is performed simultaneously.

Example 9.5 *Determine the parallel realisation of the IIR digital filter transfer functions*

(a) $H(z)=\dfrac{3(2z^2+5z+4)}{(2z+1)(z+2)}$ (b) $H(z)=\dfrac{3z(5z-2)}{\left(z+\frac{1}{2}\right)(3z-1)}$

Solution (a) In order to find the parallel realisation, the partial fraction expansion of $H(z)/z$ is first determined, just as we did for inverse z-transforms. This gives

$$F(z)=\frac{H(z)}{z}=\frac{\frac{3}{2}(2z^2+5z+4)}{z\left(z+\frac{1}{2}\right)(z+2)}=\frac{A_1}{z}+\frac{A_2}{\left(z+\frac{1}{2}\right)}+\frac{A_3}{(z+2)}$$

where
$$A_1=zF(z)\big|_{z=0}=\left.\frac{\frac{3}{2}(2z^2+5z+4)}{\left(z+\frac{1}{2}\right)(z+2)}\right|_{z=0}=6$$

$$A_2=\left(z+\frac{1}{2}\right)F(z)\bigg|_{z=-\frac{1}{2}}=\left.\frac{\frac{3}{2}(2z^2+5z+4)}{z(z+2)}\right|_{z=-\frac{1}{2}}=-4$$

$$A_3=(z+2)F(z)\big|_{z=-2}=\left.\frac{\frac{3}{2}(2z^2+5z+4)}{z\left(z+\frac{1}{2}\right)}\right|_{z=-2}=1$$

Therefore,
$$\frac{H(z)}{z}=\frac{6}{z}-\frac{4}{z+\frac{1}{2}}+\frac{1}{(z+2)}$$

Hence,
$$H(z)=6-\frac{4z}{z+\frac{1}{2}}+\frac{z}{(z+2)}=6-\frac{4}{\left(1+\frac{1}{2}z^{-1}\right)}+\frac{1}{1+2z^{-1}}$$

The parallel realisation of this transfer function is shown in Fig. E9.5(a).

(b)
$$H(z)=\frac{3z(5z-2)}{(z+\frac{1}{2})(3z-1)}=\frac{z(5z-2)}{\left(z+\frac{1}{2}\right)\left(z-\frac{1}{3}\right)}$$

$$F(z)=\frac{H(z)}{z}=\frac{5z-2}{\left(z+\frac{1}{2}\right)\left(z-\frac{1}{3}\right)}=\frac{A_1}{z+\frac{1}{2}}+\frac{A_2}{\left(z-\frac{1}{3}\right)}$$

where
$$A_1=F(z)\left(z+\frac{1}{2}\right)\bigg|_{z=-\frac{1}{2}}=\left.\frac{5z-2}{\left(z-\frac{1}{3}\right)}\right|_{z=-\frac{1}{2}}=\frac{27}{5}$$

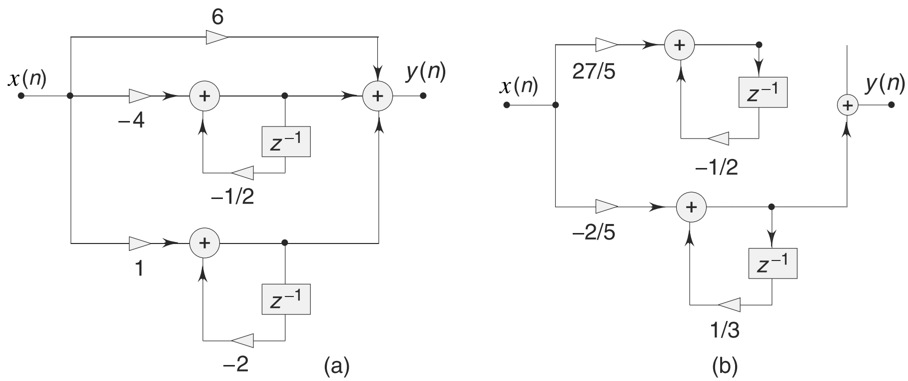


Fig. E9.5

$$A_2 = F(z)\left(z-\frac{1}{3}\right)\Bigg|_{z=\frac{1}{3}} = \frac{5z-2}{\left(z+\frac{1}{2}\right)}\Bigg|_{z=\frac{1}{3}} = -\frac{2}{5}$$

Therefore, $$H(z) = \frac{27}{5}\frac{z}{\left(z+\frac{1}{2}\right)} - \frac{2}{5}\frac{z}{\left(z-\frac{1}{3}\right)}$$

$$= \frac{27}{5}\frac{1}{\left(1+\frac{1}{2}z^{-1}\right)} - \frac{2}{5}\frac{1}{\left(1-\frac{1}{3}z^{-1}\right)}$$

The parallel realisation of this transfer function is shown in Fig. E9.5(b).

9.3.4 Transposed Forms

If two digital filter structures have the same transfer function, then they are called **equivalent structures**. A simple way to generate an equivalent structure from a given realisation structure is via the **transpose operation.** The transposed form is obtained by (i) reversing the paths, (ii) replacing pick-off nodes by adders, and vice-versa, and (iii) interchanging the input and output nodes. For a single input-output system, the transposed structure has the same transfer function as the original realisation structure.

While using infinite precision arithmetic, any realisation structure will behave identically to any other equivalent structure. But, due to the finite word length limitations, the behaviour of a particular realisation structure is totally different from its equivalent structures. Therefore, it is important to select a structure which has the minimum quantisation effects from the finite word length implementation.

Figure 9.10 shows the transposed direct form I realisation network of Fig. 9.4. Figure 9.11 also illustrates the direct form I network of Fig. 9.10 which is redrawn to have the input on the left and output on the right conventionally. Figure 9.12 shows the transposed direct form II structure.

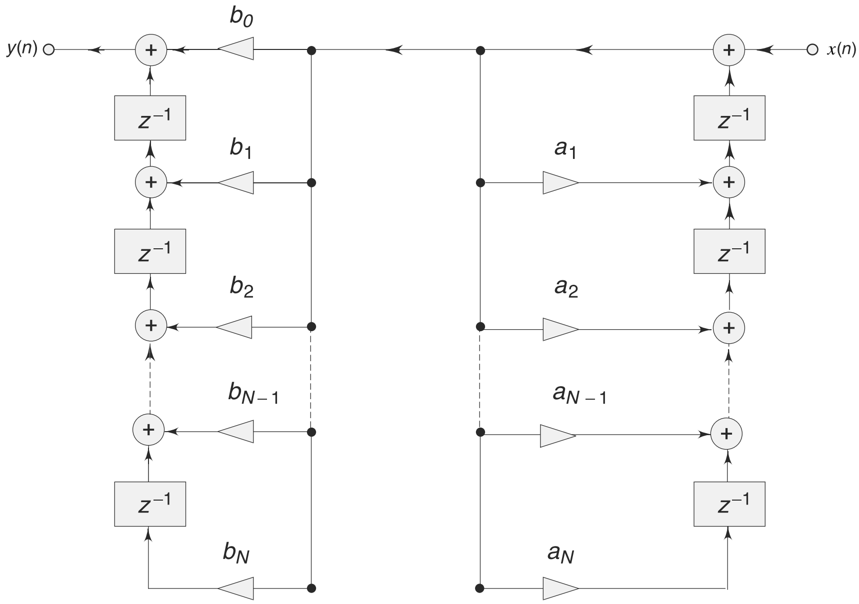


Fig. 9.10 *Transposed Direct Form I Realisation Structure of Fig. 9.4*

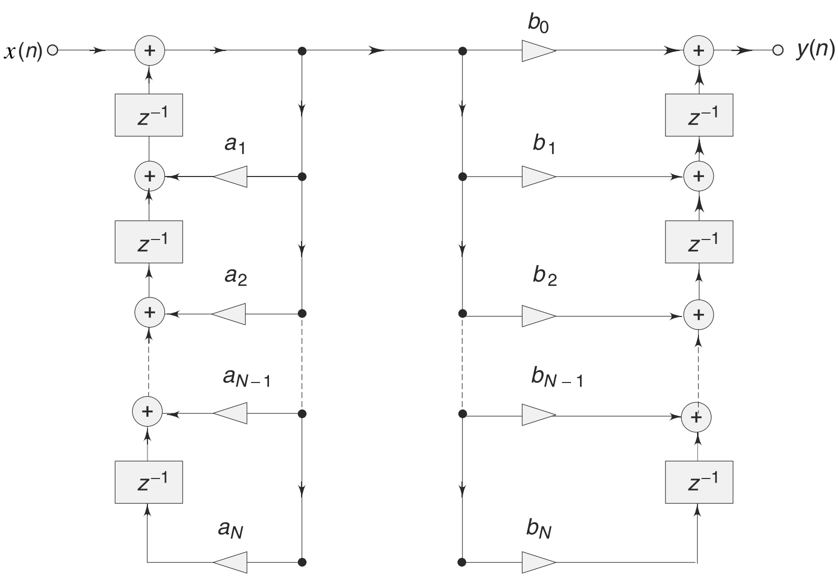


Fig. 9.11 *Transposed Direct Form I Realisation Structure*

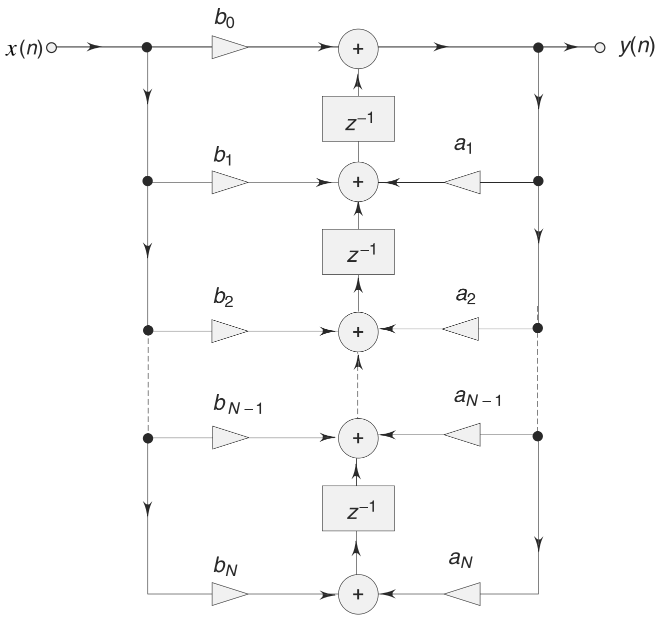


Fig. 9.12 *Transposed Direct Form II Realisation Structure of Fig. 9.7*

Example 9.6 *Draw the structures of cascade and parallel realisations of*

$$H(z) = \frac{(1-z^{-1})^3}{\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{1}{8}z^{-1}\right)}$$

Solution

Cascade Realisation The given transfer function $H(z)$ can be written in the form

$$H(z) = \frac{(1-z^{-1})}{\left(1-\frac{1}{2}z^{-1}\right)} \cdot \frac{(1-z^{-1})}{\left(1-\frac{1}{8}z^{-1}\right)} \cdot (1-z^{-1}) = H_1(z) \cdot H_2(z) \cdot H_3(z)$$

This function is realised in cascade structure as shown in Fig. E9.6(a).

Parallel Realisation The transfer function $H(z)$ can be written as

$$H(z) = \frac{(1-z^{-1})^3}{\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{1}{8}z^{-1}\right)} = \frac{(z-1)^3}{z\left(z-\frac{1}{2}\right)\left(z-\frac{1}{8}\right)}$$

For finding the parallel realisation, the partial fraction expansion of $H(z)/z$ is determined as

$$F(z) = \frac{H(z)}{z} = \frac{(z-1)^3}{z^2\left(z-\frac{1}{2}\right)\left(z-\frac{1}{8}\right)} = \frac{A_1}{z^2} + \frac{A_2}{z} + \frac{A_3}{z-\frac{1}{2}} + \frac{A_4}{z-\frac{1}{8}}$$

where

$$A_1 = \left.\frac{(z-1)^3}{\left(z-\frac{1}{2}\right)\left(z-\frac{1}{8}\right)}\right|_{z=0} = -16 \qquad A_3 = \left.\frac{(z-1)^3}{z^2\left(z-\frac{1}{8}\right)}\right|_{z=\frac{1}{4}} = -\frac{4}{3}$$

$$A_2 = \frac{d}{dz}\left[\frac{(z-1)^3}{\left(z-\frac{1}{2}\right)\left(z-\frac{1}{8}\right)}\right]_{z=0} = -112 \qquad A_4 = \left.\frac{(z-1)^3}{z^2\left(z-\frac{1}{2}\right)}\right|_{z=\frac{1}{8}} = \frac{343}{3}$$

Thus,

$$H(z) = -112 - 16z^{-1} - \frac{4}{3}\frac{1}{\left(1-\frac{1}{2}z^{-1}\right)} + \frac{343}{3}\frac{1}{\left(1-\frac{1}{8}z^{-1}\right)}$$

This function is realised in parallel structure as shown in Fig. E9.6(b).

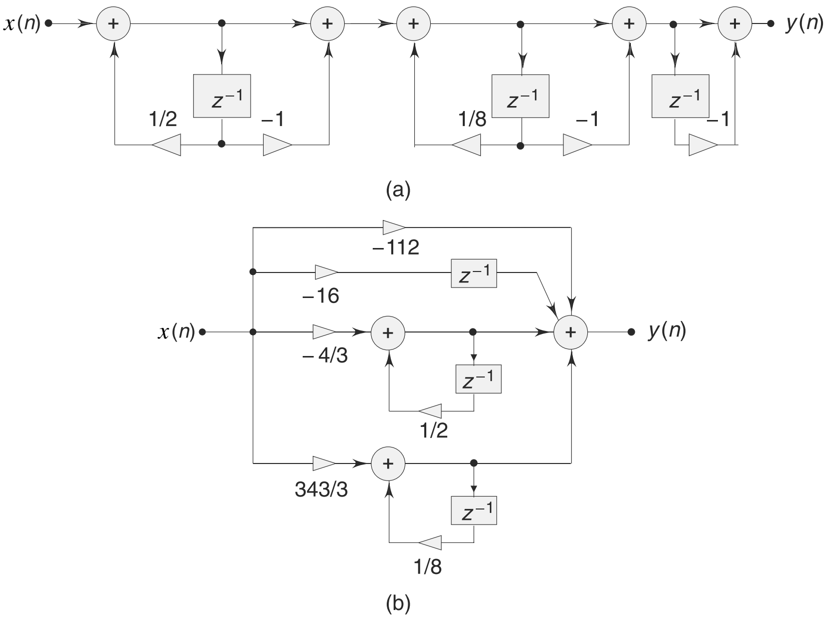


Fig. E9.6

Example 9.7 *Draw the cascade and parallel realisation structures for the system described by the system function*

$$H(z) = \frac{5\left(1-\frac{1}{4}z^{-1}\right)\left(1-\frac{2}{3}z^{-1}\right)(1+2z^{-1})}{\left(1-\frac{3}{4}z^{-1}\right)\left(1-\frac{1}{8}z^{-1}\right)\left[1-\left(\frac{1}{2}+j\frac{1}{2}\right)z^{-1}\right]\left[1-\left(\frac{1}{2}-j\frac{1}{2}\right)z^{-1}\right]}$$

Solution

Cascade Realisation This realisation can be easily obtained by grouping poles and zeros of the system function in several possible ways; one possible pairing of poles and zeros is

$$H_1(z)=\frac{1-\frac{2}{3}z^{-1}}{1-\frac{7}{8}z^{-1}+\frac{3}{32}z^{-2}},\ H_2(z)=\frac{1+\frac{7}{4}z^{-1}-\frac{1}{2}z^{-2}}{1-z^{-1}+\frac{1}{2}z^{-2}}$$

Hence, $H(z) = 5H_1(z)\,H_2(z)$

The difference equations for $H_1(z)$ and $H_2(z)$ are

$$y_1(n)=\frac{7}{8}y_1(n-1)-\frac{3}{32}y_1(n-2)+x_1(n)-\frac{2}{3}x_1(n-1)$$

$$y_2(n)=y_2(n-1)-\frac{1}{2}y_2(n-2)+x_2(n)+\frac{7}{4}x_2(n-1)-\frac{1}{2}x_2(n-2)$$

These functions are realised in cascade form as shown in Fig. E9.7(a).

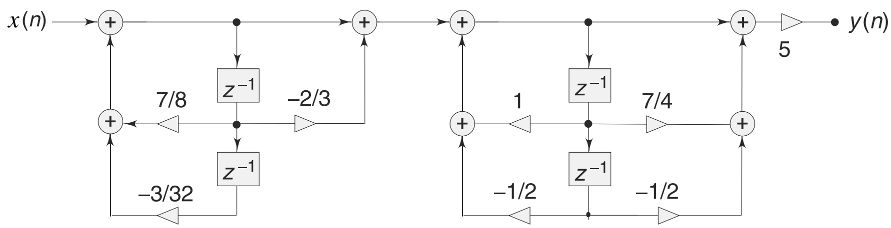


Fig. E9.7(a)

Parallel Realisation To obtain the parallel form realisation, $H(z)$ must be expanded in partial fractions. Thus we have

$$H(z)=\frac{A_1}{1-\frac{3}{4}z^{-1}}+\frac{A_2}{1-\frac{1}{8}z^{-1}}+\frac{A_3}{1-\left(\frac{1}{2}+j\frac{1}{2}\right)z^{-1}}+\frac{A_3^*}{1-\left(\frac{1}{2}-j\frac{1}{2}\right)z^{-1}}$$

Upon solving, we find $A_1 = 2.933$, $A_2 = -2.947$, $A_3 = 2.507 - j10.45$ and $A_3^* = 2.507 + j10.45$

Therefore, $$H(z)=\frac{2.933}{\left(1-\frac{3}{4}z^{-1}\right)}-\frac{2.947}{\left(1-\frac{1}{8}z^{-1}\right)}+\frac{2.507-j10.45}{1-\frac{1}{2}(1+j)z^{-1}}+\frac{2.507+j10.45}{1-\frac{1}{2}(1-j)z^{-1}}$$

$$=\frac{-0.02+1.846z^{-1}}{1-\frac{7}{8}z^{-1}+\frac{3}{32}z^{-2}}+\frac{5.02+7.743z^{-1}}{1-z^{-1}+\frac{1}{2}z^{-2}}$$

The parallel form realisation is illustrated in Fig. E9.7(b).

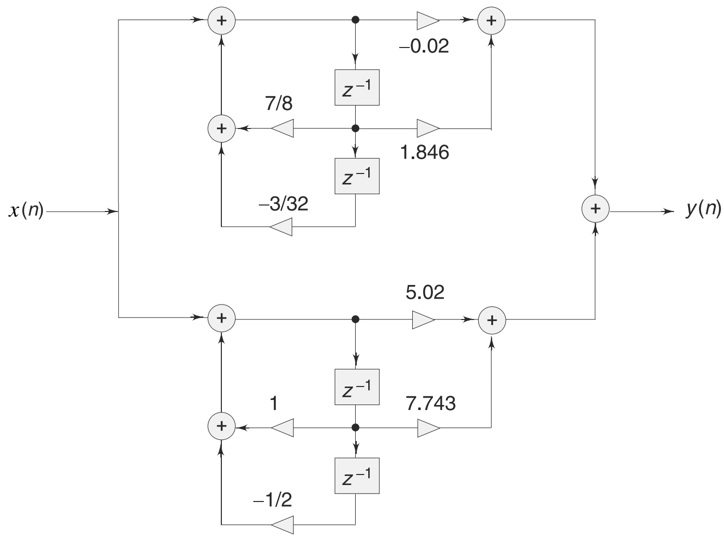


Fig. E9.7(b)

Example 9.8 *Obtain the cascade and parallel realisations for the system function given by*

$$H(z) = \frac{1 + \frac{1}{4}z^{-1}}{\left(1 + \frac{1}{2}z^{-1}\right)\left(1 + \frac{1}{2}z^{-1} + \frac{1}{4}z^{-2}\right)}$$

Solution

Cascade Realisation To obtain the cascade realisation, the transfer function is broken into a product of two functions as

$$H(z) = H_1(z)\, H_2(z)$$

where $$H_1(z) = \frac{1 + \frac{1}{4}z^{-1}}{1 + \frac{1}{2}z^{-1}} \text{ and } H_2(z) = \frac{1}{1 + \frac{1}{2}z^{-1} + \frac{1}{4}z^{-2}}$$

The cascade realisation structure for this system function is shown in Fig. E9.8(a).

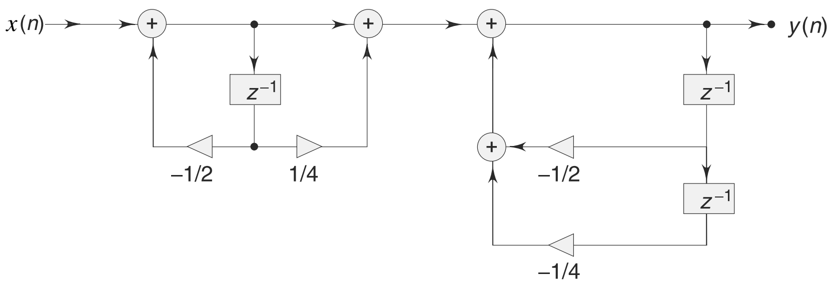


Fig. E9.8(a)

Parallel Realisation To obtain the parallel realisation, we make a partial fraction expansion of $H(z)$, resulting in

$$H(z)=\frac{1+\frac{1}{4}z^{-1}}{\left(1+\frac{1}{2}z^{-1}\right)\left(1+\frac{1}{2}z^{-1}+\frac{1}{4}z^{-2}\right)}=\frac{A_1}{\left(1+\frac{1}{2}z^{-1}\right)}+\frac{A_2 z^{-1}+A_3}{1+\frac{1}{2}z^{-1}+\frac{1}{4}z^{-2}}$$

Multiplying out and equating the coefficients of negative powers of z and then solving, we obtain, $A_1 = \frac{1}{2}, A_2 = -\frac{1}{4}$ and $A_3 = \frac{1}{2}$

Therefore,
$$H(z)=\frac{\frac{1}{2}}{1+\frac{1}{2}z^{-1}}+\frac{\frac{1}{2}-\frac{1}{4}z^{-1}}{1+\frac{1}{2}z^{-1}+\frac{1}{4}z^{-2}}$$

The parallel realisation structure is shown in Fig. E9.8(b).

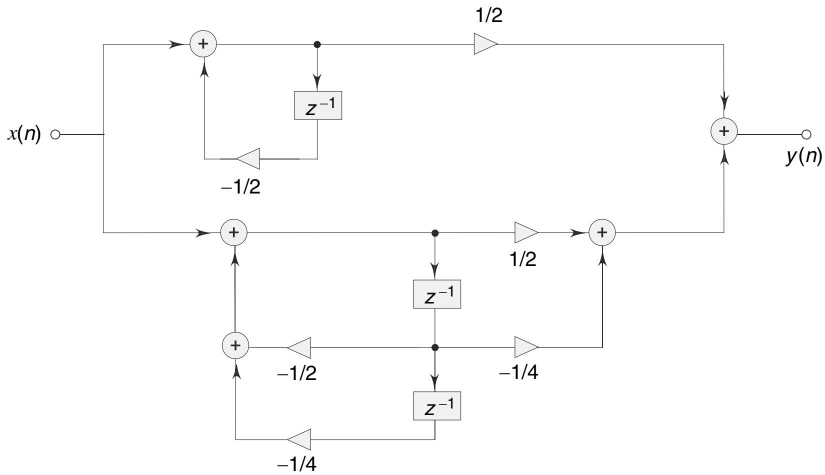


Fig. E9.8(b)

Example 9.9 *Develop cascade and parallel realisation structures for*

$$H(z)=\frac{\frac{z}{6}+\frac{5}{24}+\frac{5}{24}z^{-1}+\frac{1}{24}z^{-2}}{1-\frac{1}{2}z^{-1}+\frac{1}{4}z^{-2}}$$

Solution

Cascade Realisation The transfer function can be rearranged as

$$H(z)=\frac{\frac{z}{6}\left(1+\frac{5}{4}z^{-1}+\frac{5}{4}z^{-2}+\frac{1}{4}z^{-3}\right)}{1-\frac{1}{2}z^{-1}+\frac{1}{4}z^{-2}}=\frac{\frac{z}{6}\left(1+\frac{1}{4}z^{-1}\right)(1+z^{-1}+z^{-2})}{1-\frac{1}{2}z^{-1}+\frac{1}{4}z^{-2}}$$

$$= \frac{1}{6} z(1 + z^{-1} + z^{-2}) \frac{1 + \frac{1}{4} z^{-1}}{1 - \frac{1}{2} z^{-1} + \frac{1}{4} z^{-2}}$$

The above function is in the form of $H(z) = \frac{1}{6} H_1(z) \,.\, H_2(z).\, H_3(z)$.

There are two ways of implementing this system, depending upon the combination of numerator term with the denominator term.

The cascade realisation structure is shown in Fig. E9.9(a).

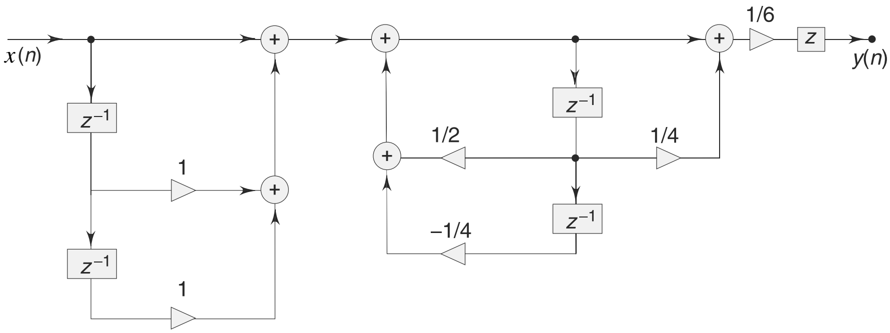


Fig. E9.9(a)

Parallel Realisation

$$H(z) = \frac{\frac{z}{6} + \frac{5}{24} + \frac{5}{24} z^{-1} + \frac{1}{24} z^{-2}}{1 - \frac{1}{2} z^{-1} + \frac{1}{4} z^{-2}} = \frac{\frac{z}{6}\left(1 + \frac{5}{4} z^{-1} + \frac{5}{4} z^{-2} + \frac{1}{4} z^{-3}\right)}{1 - \frac{1}{2} z^{-1} + \frac{1}{4} z^{-2}}$$

Dividing the numerator by the denominator, we obtain

$$\begin{array}{r|l} & z^{-1} + 7 \\ \frac{1}{4} z^{-2} - \frac{1}{2} z^{-1} + 1 & \frac{1}{4} z^{-3} + \frac{5}{4} z^{-2} + \frac{5}{4} z^{-1} + 1 \\ & \frac{1}{4} z^{-3} - \frac{1}{2} z^{-2} + z^{-1} \\ \hline & \frac{7}{4} z^{-2} + \frac{1}{4} z^{-1} + 1 \\ & \frac{7}{4} z^{-2} + \frac{7}{2} z^{-1} + 7 \\ \hline & \frac{15}{4} z^{-1} - 6 \\ \hline \end{array}$$

Therefore,

$$H(z) = \frac{z}{6}\left[7 + z^{-1} + \frac{-6 + \frac{15}{4} z^{-1}}{1 - \frac{1}{2} z^{-1} + \frac{1}{4} z^{-2}}\right] = z\left[\frac{7}{6} + \frac{1}{6} z^{-1} + \frac{-1 + \frac{5}{8} z^{-1}}{1 - \frac{1}{2} z^{-1} + \frac{1}{4} z^{-2}}\right]$$

$$H(z) = z\,[H_1(z) + H_2(z)]$$

where $$H_1(z) = \frac{7}{6} + \frac{1}{6}z^{-1} \text{ and } H_2(z) = \frac{-1+\frac{5}{8}z^{-1}}{1-\frac{1}{2}z^{-1}+\frac{1}{4}z^{-2}}$$

The parallel realisation structure is shown in Fig. E9.9(b).

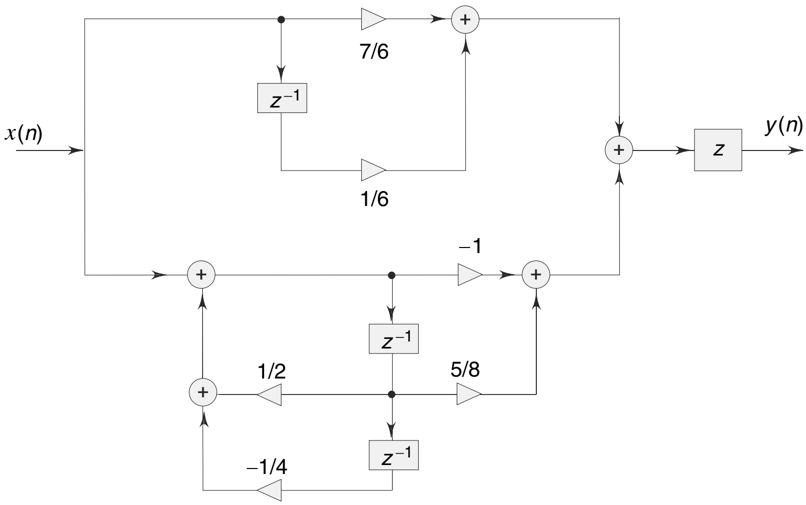


Fig. E9.9(b)

9.3.5 Ladder Structures

Filters realised using ladder structures have desirable coefficient sensitivity properties. Small changes in the filter parameters have little effect on its performance.

Consider a filter represented by the transfer function

$$H(z) = \frac{a_N z^{-N} + a_{N-1} z^{N+1} + \cdots + a_1 z^{-1} + a_0}{b_N z^{-N} + b_{N-1} z^{-N+1} + \cdots + b_1 z^{-1} + b_0}$$

$$= \alpha_0 + \cfrac{1}{\beta_1 z^{-1} + \cfrac{1}{\alpha_1 + \cfrac{1}{\beta_2 z^{-1} + \cdots + \cfrac{1}{\alpha_N}}}}$$

To calculate the values of $\alpha_i s$ and β_is, form the Routh array.

z^{-N}	a_N	a_{N-1}	$a_{N-2} \ldots a_1 a_0$
z^{-N}	b_N	b_{N-1}	$b_{N-2} \ldots b_1 b_0$
z^{-N+1}	c_{N-1}	c_{N-2}	$c_{N-3} \ldots c_0$
z^{-N+1}	d_{N-1}	d_{N-2}	$d_{N-3} \ldots d_0$
z^{-N+2}	e_{N-2}	e_{N-3}	$e_{N-4} \ldots$
z^{-N+2}	f_{N-2}	f_{N-3}	$f_{N-4} \ldots$
1	g_0		
1	h_0		

The ladder structure parameters are given by

$$\alpha_0 = \frac{a_N}{b_N}, \quad \beta_1 = \frac{b_N}{c_{N-1}}, \quad \alpha_1 = \frac{c_{N-1}}{d_{N-1}}$$

and so on.

The system is represented as,

$$Y(z) = H(z)\,X(z)$$

$$= \left[\alpha_0 + \cfrac{1}{\beta_1 z^{-1} + \cfrac{1}{R_1(z)}}\right] X(z) = \alpha_0 X(z) + H_1(z)\,X(z)$$

where, $$H_1(z) = \cfrac{1}{\beta_1 z^{-1} + \cfrac{1}{R_1(z)}}$$

and

$$R_1(z) = h\alpha_1 + \cfrac{1}{\beta_2 z^{-1} + \cfrac{1}{\alpha_2 + \vdots + \cfrac{1}{\alpha_N}}}$$

$$H_1(z) = \cfrac{1}{\beta_1 z^{-1} + \cfrac{1}{R_1(z)}} = \frac{Y_1(z)}{X(z)}$$

$$Y_1(z)\left\{\beta_1 z^{-1} + \frac{1}{R_1(z)}\right\} = X(z)$$

$$Y_1(z) = R_1(z)\{X(z) - \beta_1 z^{-1} Y_1(z)\} = R_1(z)\,X_1(z)$$

Combining Figs. 9.13(a)) and (b), We obtain the structure as shown in Fig. 9.13(c).

$$R_1(z) = \frac{Y_1(z)}{X_1(z)} = \alpha_1 + \cfrac{1}{\beta_2 z^{-1} + \cfrac{1}{R_1(z)}}$$

With continued fractional realisation of $H(z)$, the resultant structure is shown in Fig. 9.13(e).

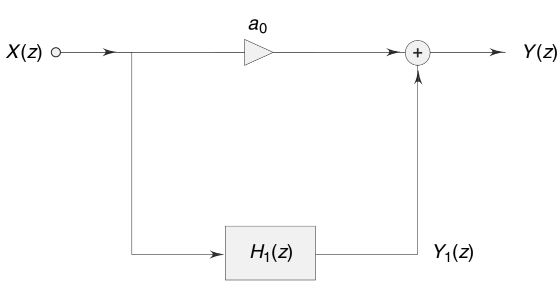


Fig. 9.13(a)

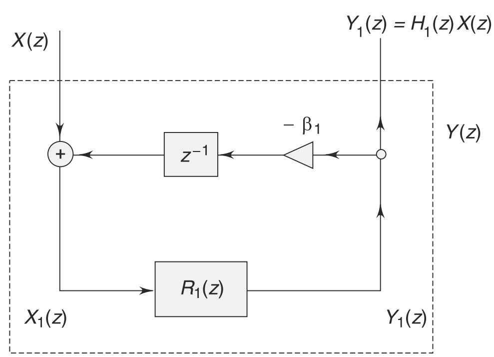


Fig. 9.13(b)

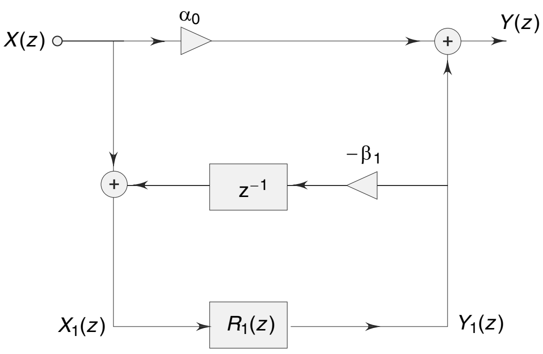


Fig. 9.13(c)

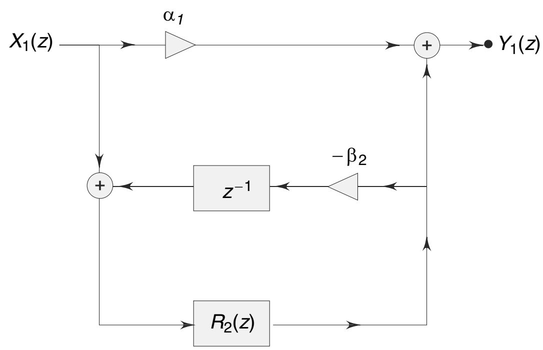


Fig. 9.13(d)

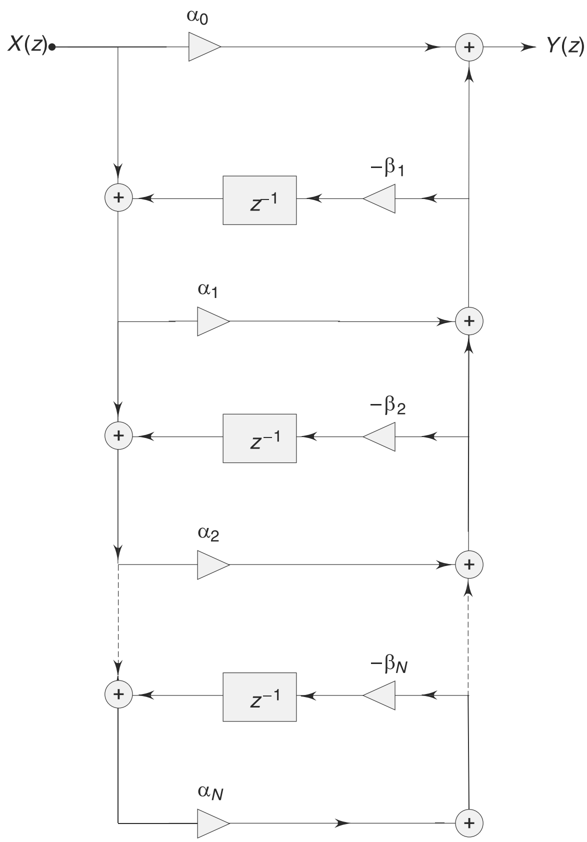


Fig. 9.13(e)

Example 9.10 *Given the system function*

$$H(z)=\frac{2+8z^{-1}+6z^{-2}}{1+8z^{-1}+12z^{-2}}$$

Realise using ladder structure.

Solution For the given system, obtain the Routh array

z^{-2}	6	8	2
z^{-2}	12	8	1
z^{-1}	4	3/2	
z^{-1}	7/2	1	
1	5/14	0	
1	1		

The ladder structure parameters are

$$\alpha_0 = \frac{1}{2}, \quad \beta_1 = 3, \quad \alpha_1 = \frac{8}{7}, \quad \beta_2 = \frac{49}{5}, \quad \alpha_2 = \frac{5}{14}$$

$$H(z) = \frac{1}{2} + \cfrac{1}{3z^{-1} \cfrac{1}{\frac{8}{7} + \cfrac{1}{(49/5)z^{-1} + \cfrac{1}{5/14}}}}$$

The ladder structure is shown in Fig. E9.10.

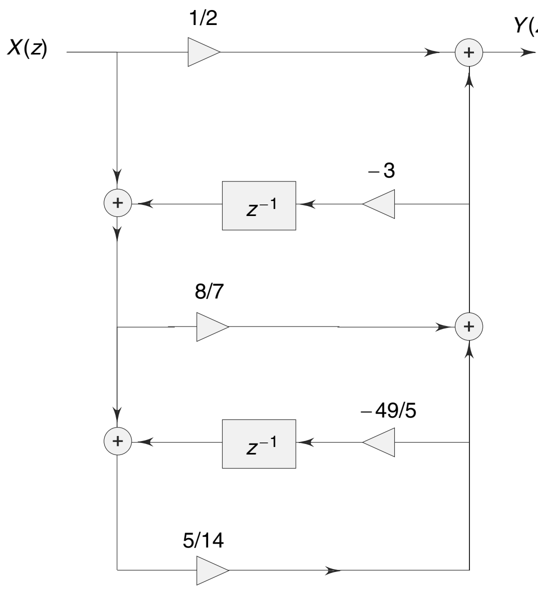


Fig. E9.10

9.3.6 State Space Structure

State space realisation requires a large number of computations as compared to the other forms. But under finite word length constrains, it provides improved performance.

Consider a second-order IIR filter described by its state space equations,

$$\dot{r} = Ar + Bx$$
$$y = Cr + Dx$$

where
x = input
y = output
A = state matrix
B = input matrix
C = output matrix
D = direct transmission matrix
r = state variables

$$\begin{bmatrix} r_1(n+1) \\ r_2(n+2) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} r_1(n) \\ r_2(n) \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} x(n)$$

$$y(n) = \begin{bmatrix} c_1 & c_2 \end{bmatrix} \begin{bmatrix} r_1(n) \\ r_2(n) \end{bmatrix} + d\,x(n)$$

The state space realisation of an Nth order IIR filter requires $(N + 1)^2$ multipliers and $N(N + 1)$ two input adders.

From the state space equation of the given system, express the intermediate state variables, input and output

$$r_1(n + 1) = a_{11}\, r_1(n) + a_{12}\, r_2(n) + b_1\, x(n)$$
$$r_2(n + 1) = a_{21}\, r_1(n) + a_{22}\, r_2(n) + b_2\, x(n)$$
$$y(n) = c_1\, r_1(n) + c_2\, r_2(n) + d\, x(n)$$

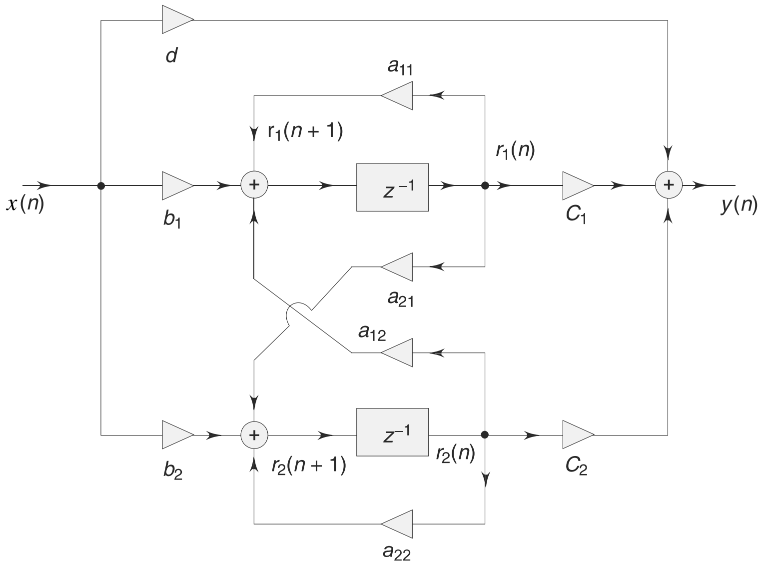


Fig. 9.14

Example 9.11 *Realise a system defined by the following state space equations*

$$\begin{bmatrix} r_1(n+1) \\ r_2(n+2) \end{bmatrix} = \begin{bmatrix} -5 & -1 \\ 3 & -1 \end{bmatrix} \begin{bmatrix} r_1(n) \\ r_2(n) \end{bmatrix} + \begin{bmatrix} 4 \\ 6 \end{bmatrix} x(n)$$

$$y(n) = \begin{bmatrix} 2 & 3 \end{bmatrix} \begin{bmatrix} r_1(n) \\ r_2(n) \end{bmatrix}$$

Solution Expand the state space equations and obtain the relation between the state variables, input and output.

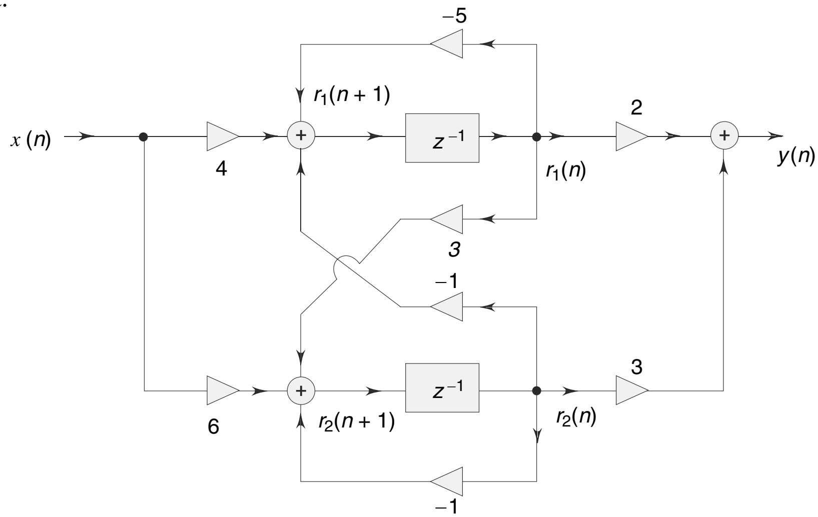


Fig. E9.11

$$r_1(n+1) = -5r_1(n) - r_2(n) + 4x(n)$$
$$r_2(n+1) = 3r_1(n) - r_2(n) + 6x(n)$$
$$y(n) = 2r_1(n) + 3r_2(n)$$

Example 9.12 *Develop the state space representation of the digital filter structure shown in Fig. E9.12 and draw the block diagram representation of this state space structure.*

Solution

$$r_2(n+1) = r_1(n)$$
$$r_1(n+1) = x(n) + \{(r_2(n) + x(n))\, b + r_1(n)\}\, a$$
$$= x(n)\,(1 + ab) + ab\, r_2(n) = a\, r_1(n)$$
$$y(n) = r_2(n) - a\, \{(r_2(n) + x(n))\, b + r_1(n)\}$$
$$= r_2(n)\,(1 - ab) - ab\, x(n) - a\, r_1(n)$$

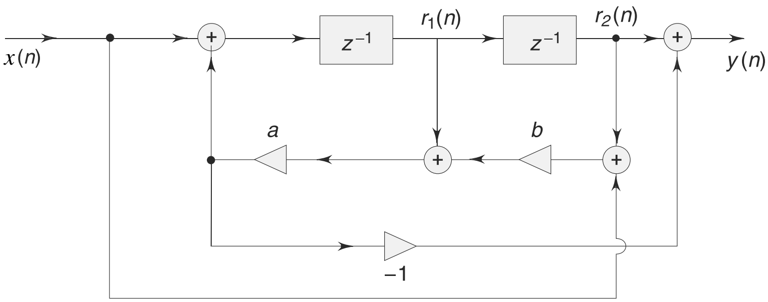


Fig. E9.12(a)

The state space equation is given by

$$\begin{bmatrix} r_1(n+1) \\ r_2(n+1) \end{bmatrix} = \begin{bmatrix} a & ab \\ 1 & 0 \end{bmatrix} \begin{bmatrix} r_1(n) \\ r_2(n) \end{bmatrix} + \begin{bmatrix} 1+ab \\ 0 \end{bmatrix} x(n)$$

$$y(n) = \begin{bmatrix} -a & 1-ab \end{bmatrix} \begin{bmatrix} r_1(n) \\ r_2(n) \end{bmatrix} - ab\, x(n)$$

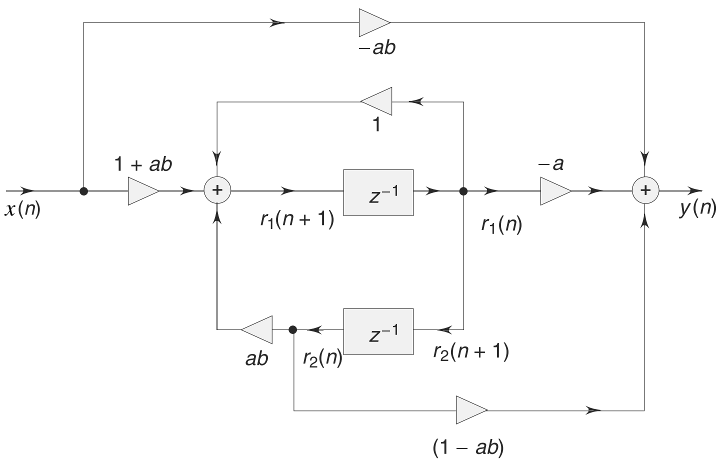


Fig. E9.12(b)

BASIC STRUCTURES FOR FIR SYSTEMS 9.4

The system function of the FIR systems whose realisation take the form of a non-recursive computational algorithm may be written as

$$H(z) = \sum_{n=0}^{M-1} h(n) z^{-n} \tag{9.7}$$

That is, if the impulse response is M samples in duration, the $H(z)$ is a polynomial in z^{-1} of degree $M - 1$. Thus $H(z)$ has $(M - 1)$ poles at $z = 0$ and $(M - 1)$ zeros that can be anywhere in the finite z-plane.

FIR filters are often preferred in many applications, since they provide an exact linear phase over the whole frequency range and they are always BIBO stable independent of the filter coefficients. Two simple realisation methods for FIR filters, viz. direct form and cascade form are outlined in this section.

9.4.1 Direct Form Realisation of FIR Systems

The convolution sum relationship gives the system response as

$$y(n) = \sum_{K=0}^{M-1} h(k) x(n-k) \tag{9.8}$$

where $y(n)$ and $x(n)$ are the output and input sequence, respectively. This equation gives the input-output relation of the FIR filter in the time-domain. This equation can be obtained from Eq. (9.1) by setting $b_k = h(k)$, $a_k = 0$, $k = 1, 2,..., M$.

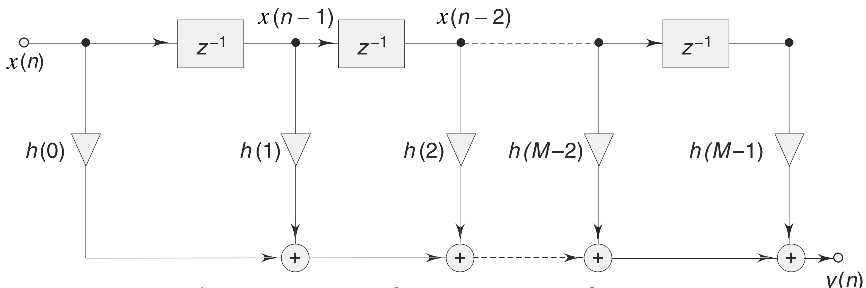


Fig. 9.15 *Direct Form Realisation Structure of an FIR System*

The direct form realisation for Eq. (9.8) is shown in Fig. 9.15. This is similar to that of Fig. 9.7 when all the coefficients $a_k = 0$. Hence, the direct form realisation structure for FIR system is the special case of the direct form realisation structure for IIR system.

The transposed structure shown in Fig. 9.16 is the second direct form structure. Both of these direct form structures are canonic with respect to delays.

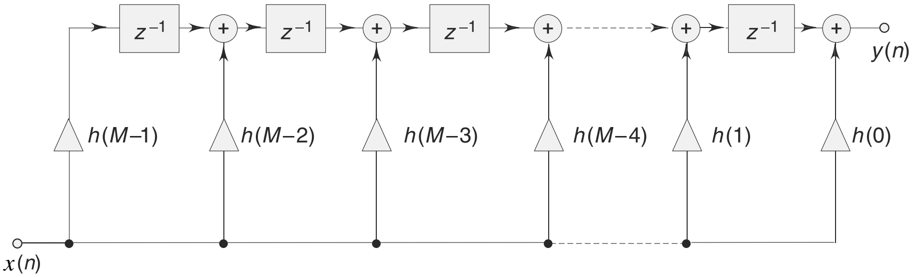


Fig. 9.16 *Transposed Direct Form Realisation Structure of an FIR System of Fig. 9.15*

9.4.2 Cascade Form Realisation of FIR Systems

As an alternative to the Direct-Form, we factor the FIR transfer function $H(z)$ as a product of second-order factors given by

$$H(z) = \prod_{k=1}^{M/2} (\beta_{0k} + \beta_{1k} z^{-1} + \beta_{2k} z^{-2}) \tag{9.9}$$

where if M is even, $\beta_2 k = 0$ and $H(z)$ has an odd number of real roots. A realisation of Eq. (9.9) is shown in Fig. 9.17 for a cascade of second-order sections. This cascade form is canonic with respect to delay, and also employs $(M - 1)$ two input adders and M multipliers for an $(M - 1)$th order FIR transfer function.

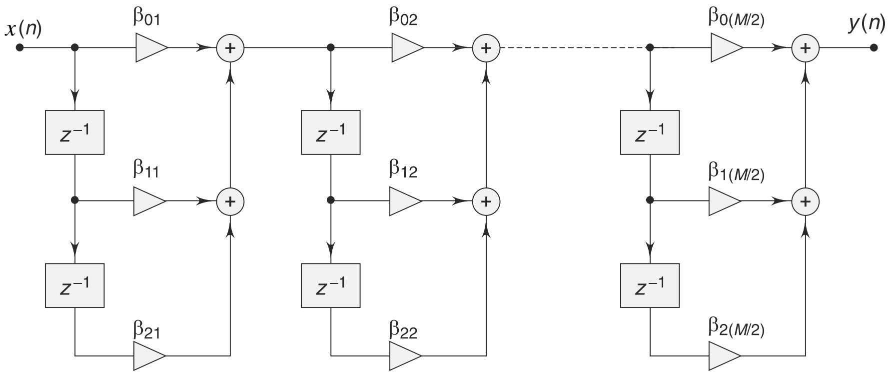


Fig. 9.17 *Cascade Form Realisation Structure of an FIR System*

Example 9.13 *Obtain direct form and cascade form realisation for the transfer function of an FIR system given by*

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

Solution

Direct Form Realisation Expanding the transfer function $H(z)$, we have

$$H(z) = 1 - \frac{3}{8} z^{-1} - \frac{3}{32} z^{-2} + \frac{5}{64} z^{-3} - \frac{3}{16} z^{-4}$$

This function is realised in FIR direct form as shown in Fig. E9.13(a).

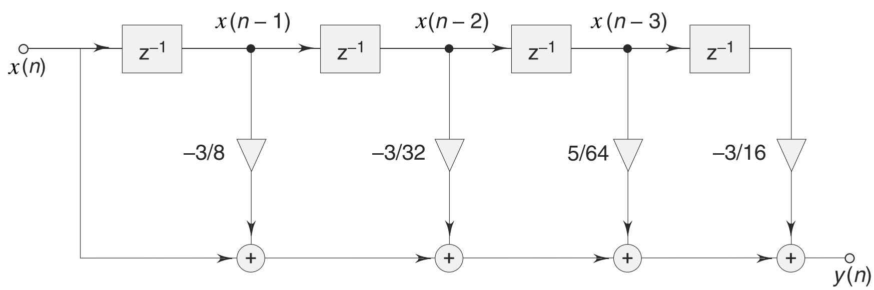


Fig. E9.13(a)

Cascade Realisation

$$H(z)=\left(1-\frac{1}{4}z^{-1}+\frac{3}{8}z^{-2}\right)\left(1-\frac{1}{8}z^{-1}-\frac{1}{2}z^{-2}\right) = H_1(z)\cdot H_2(z)$$

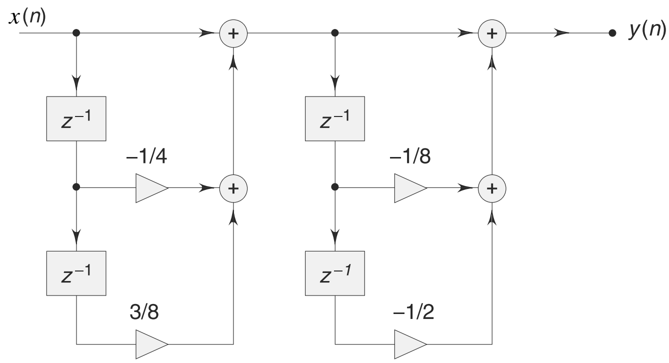


Fig. E9.13(b)

This function is realised in FIR cascade form as shown in Fig. E9.13(b).

9.4.3 Realisation of Linear Phase FIR Systems

If the impulse response is symmetric about its origin, linear phase results. If all zero filter has a linear phase response, a special non-recursive structure that reduces the number of multiplications by approximately one-half can be implemented. The impulse response for a causal filter begins at zero and ends at $M-1$.

As discussed in Section 7.2, a linear phase FIR filter of length M is characterised by

$$h(n) = h(M-1-n) \tag{9.10}$$

The symmetry property of a linear phase FIR filter is used to reduce the multipliers required in these realisations. Using this condition, the z-transform of the impulse response can be expressed as

$$H(z)=Z\left[h(n)\right]=\sum_{n=0}^{M-1}h(n)\,z^{-n} \tag{9.11}$$

For *M* even

$$H(z)=\sum_{n=0}^{M/2-1}h(n)\left[z^{-n}+z^{-(M-1-n)}\right] \tag{9.12}$$

For *M* odd

$$H(z)=h\left(\frac{M-1}{2}\right)z^{-(M-1)/2}+\sum_{n=0}^{(M-3)/2}h(n)\left[z^{-n}+z^{-(M-1-n)}\right] \tag{9.13}$$

We know that the output transform $Y(z) = H(z)\,X(z)$. When M is even,

$$\begin{aligned} Y(z) &= \sum_{n=0}^{\frac{M}{2}-1}h(n)\left[z^{-n}+z^{-(M-1-n)}\right]X(z) \\ &= h(0)\left[1+z^{-(M-1)}\right]X(z)+h(1)\left[z^{-1}+z^{-(M-2)}\right]X(z) \\ &\quad +\cdots+h\left(\frac{M}{2}-1\right)\left[z^{-(M/2-1)}+z^{-M/2}\right]X(z) \end{aligned} \tag{9.14}$$

The inverse z-transform or $Y(z)$ is

$$y(n) = h(0)[x(n) + x\{n - (M - 1)\}] + h(1)\,[x(n - 1) + x\{n - (M - 2)\}]$$
$$+\cdots+h\left(\frac{M}{2}-1\right)\left[x\left\{n-\left(\frac{M}{2}-1\right)\right\}\right]+x\left(n-\frac{M}{2}\right) \qquad (9.15)$$

The realisation structure for even M, corresponding to Eq. (9.15), is shown in Fig. 9.18. For this case, $M/2$ multipliers are required.

The same procedure can be applied to Eq. (9.13) for M odd, resulting in the realisation shown in Fig. 9.19. Here, $\frac{M-3}{2}+1+1$ or $\frac{(M+1)}{2}$ multipliers are required.

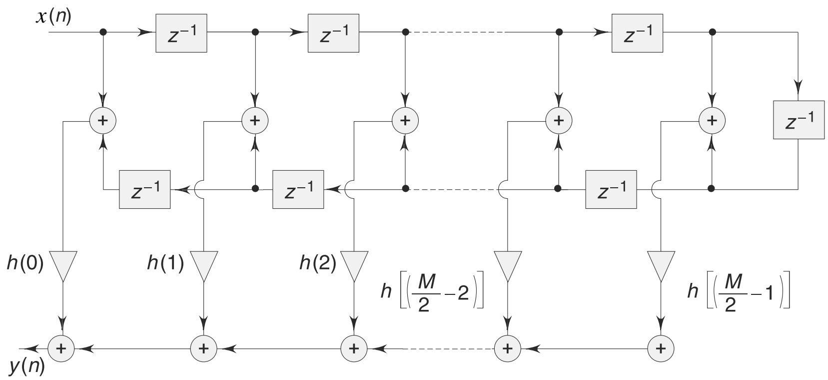


Fig. 9.18 *Direct Form Realisation Structure of a Linear Phase FIR System when M is Even*

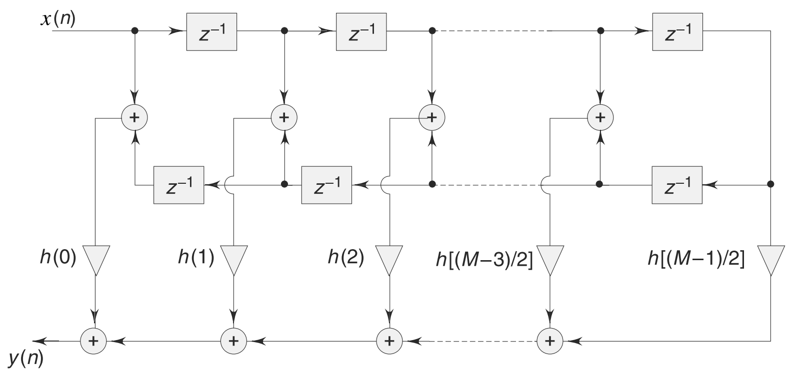


Fig. 9.19 *Direct Form Realisation Structure of a Linear Phase FIR System when M is Odd*

The linear phase FIR system realisations require approximately only half the number of multipliers than direct form and cascade form realisations.

Example 9.14 *Obtain FIR linear-phase and cascade realisations of the system function*

$$H(z)=\left(1+\frac{1}{2}z^{-1}+z^{-2}\right)\left(1+\frac{1}{4}z^{-1}+z^{-2}\right)$$

Solution

Linear Phase Realisation Expanding $H(z)$, we have

$$H(z) = 1 + \frac{3}{4}z^{-1} + \frac{17}{8}z^{-2} + \frac{3}{4}z^{-3} + z^{-4}$$

Here $M = 5$ and the realisation is shown in Fig. E9.14(a).

Cascade Realisation

$$H(z) = \left(1 + \frac{1}{2}z^{-1} + z^{-2}\right)\left(1 + \frac{1}{4}z^{-1} + z^{-2}\right)$$

$H(z)$ has a product of two sections which have the linear phase symmetry property. The corresponding cascade realisation is shown in Fig. E9.14(b).

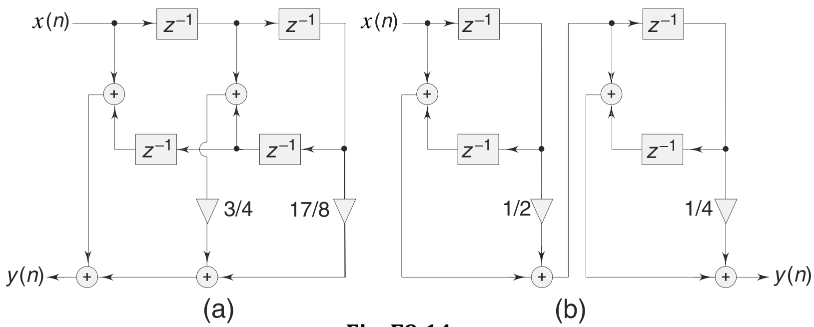


Fig. E9.14

Example 9.15 *Realise the following non-causal linear phase FIR system function*

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

Solution This function can be implemented in the direct form structure as shown in Fig. E9.15(a), which uses two multipliers.

Using the symmetry property of a linear phase FIR system, the equal-valued coefficients can be combined to give

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

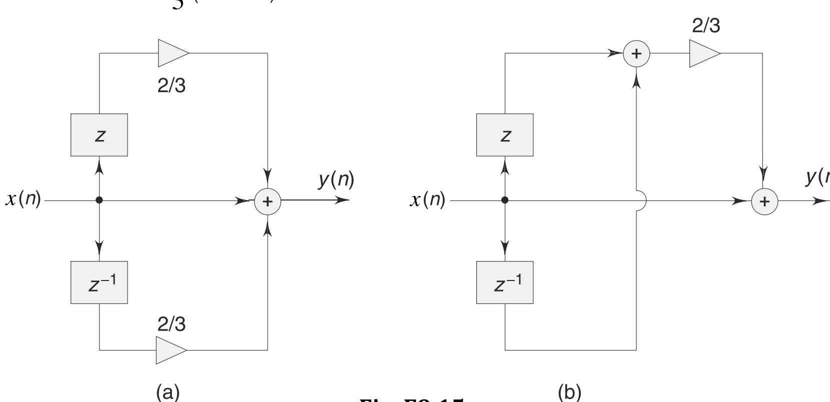


Fig. E9.15

This filter structure has only one multiplier as shown in Fig. E9.15(b). Both the structures shown in Fig. E9.15(a) and (b) are equivalent to a non-causal linear phase FIR system.

Example 9.16 *Realise the following causal linear phase FIR system function*

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

Solution The equal-valued coefficients are combined and the system function is expressed as

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

This function is realised in two equivalent structures as shown in Fig. E9.16(a) and (b).

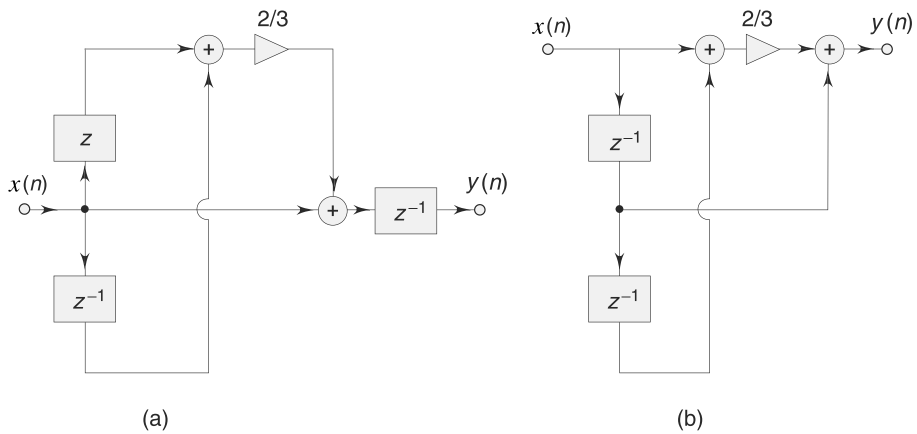


Fig. E9.16

REVIEW QUESTIONS

9.1 How will you obtain the transfer function

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 - \sum_{k=1}^{N} a_k z^{-k}}$$

from the linear difference equation

$$y(n) = \sum_{k=1}^{N} a_k\, y(n-k) + \sum_{k=0}^{M} b_k\, x(n-k)?$$

9.2 Distinguish between the methods of realisation namely, block diagram representation and signal flow graph for implementing the digital filter transfer function.

9.3 What are the basic building blocks of realisation structures?

9.4 What are the advantages of representing digital systems in block diagram form?

9.5 Define canonic and non-canonic structures.

9.6 Compare direct form I and direct form II realisations of IIR systems.

9.7 What are the drawbacks of direct form realisation of IIR systems?

9.8 Explain any two IIR filter realisation methods.

9.9 How will you develop a cascade structure with direct form II realisation of a sixth-order IIR transfer function?

9.10 How will you develop a parallel structure with direct form II realisation of a sixth-order IIR transfer function?

9.11 Draw the transposed direct forms I and II structures of a third-order IIR filter.

9.12 Discuss in detail direct form and cascade form realisations of FIR systems.

9.13 What is meant by a linear phase filter?

9.14 Why do FIR filters have inherent linear phase characteristics?

9.15 What is the necessary and sufficient condition for the linear phase characteristics in an FIR filter?

9.16 How will you develop direct form realisations of third and fourth order functions of linear phase FIR systems.

9.17 Draw a block diagram representation for the following systems

(a) $y(n) = ay(n-1) + ax(n) - x(n-1)$

(b) $y(n) = x(n) + \frac{1}{2}x(n-1) + x(n-2)$

(c) $y(n) = \frac{1}{2}y(n-1) + x(n) - \frac{1}{2}x(n-1)$

9.18 Draw the block diagram representation of the direct form I and II realisations of the systems with the following transfer functions.

(a) $H(z) = \dfrac{6z(z^2-4)}{5z^3 - 4.5z^2 + 1.4z - 0.8}$

(b) $H(z) = \dfrac{z^{-1} - 3z^{-2}}{(10 - z^{-1})(1 + 0.5z^{-1} + 0.5z^{-2})}$

(c) $H(z) = \dfrac{2 + 3z^{-2} - 1.5z^{-4}}{1 - 0.2z^{-1} + 0.35z^{-2}}$

9.19 Develop a canonic direct form realisation of the transfer function

$$H(z) = \frac{3 + 5z^{-1} - 8z^{-2} + 4z^{-5}}{2 + 3z^{-1} + 6z^{-3}}$$

and then determine its transpose configuration.

9.20 Obtain the cascade and parallel realisation structures for the following signals.

(a) $y(n) = \frac{3}{4}y(n-1) - \frac{1}{8}y(n-2) + x(n) + \frac{1}{3}x(n-1)$

(b) $y(n) = -0.1\, y(n-1) + 0.72\, y(n-2) + 0.7\, x(n) - 0.252\, x(n-2)$

(c) $H(z) = \dfrac{2(1-z^{-1})(1+\sqrt{2}z^{-1}+z^{-2})}{((1+0.5z^{-1}(1-0.9z^{-1}+0.81z^{-2}))}$

(d) $y(n) = \dfrac{1}{2}y(n-1) + \dfrac{1}{4}y(n-2) + x(n) + x(n-1)$

(e) $y(n) = y(n-1) - \dfrac{1}{2}y(n-2) + x(n) - x(n-1) + x(n-2)$

9.21 Develop the cascade and parallel forms of the following casual IIR transfer functions.

(a) $H_1(z) = \dfrac{\left(3+5z^{-1}\right)\left(0.6+3z^{-1}\right)}{\left(1-2z^{-1}+2z^{-2}\right)\left(1-z^{-1}\right)}$

(b) $H_2(z) = \dfrac{z\left(z+\frac{1}{2}\right)\left(z^2+\frac{1}{3}\right)}{\left(z-\frac{1}{2}\right)\left(z+\frac{1}{2}\right)\left(z^2-z+\frac{1}{2}\right)}$

(c) $H(z) = \dfrac{8z^3-4z^2+11z-2}{\left(z-\frac{1}{4}\right)\left(z^2-z+\frac{1}{2}\right)}$

(d) $H(z) = \dfrac{2+5z^{-1}+12z^{-2}}{\left(1+\frac{1}{2}z^{-1}-\frac{1}{4}z^{-2}\right)\left(1+\frac{1}{4}z^{-1}+\frac{1}{8}z^{-2}\right)}$

9.22 Draw the structures of cascade and parallel realisations for the transfer function

$$H(z) = \frac{\left(2+z^{-1}\right)}{\left(1-z^{-1}\right)} + \frac{\left(-3+z^{-1}\right)^2}{\left(5+z^{-1}\right)^2}$$

9.23 Realise the following IIR system functions in the direct forms I and II, cascade and parallel structures.

(a) $H(z) = \dfrac{1}{(1+az^{-1})(1-bz^{-1})}$

(b) $H(z) = \dfrac{1}{(1+rz^{-1})^3}$

(c) $H(z) = \dfrac{1}{(1-az^{-1})^2} + \dfrac{1}{(1-bz^{-1})^2}$

(d) $H(z) = \dfrac{G(1+z^{-1})(1+z^{-2})(1+z^{-1}+z^{-2})}{(1-0.7z^{-1})(1-1.2z^{-1}+0.8z^{-2})(1-1.2z^{-1}+0.6z^{-2})}$

(e) $H(z) = G\left(\dfrac{1+z^{-1}}{1-0.8z^{-1}}\right)\left(\dfrac{1+2z^{-1}+z^{-2}}{1-1.6z^{-1}+0.8z^{-2}}\right)$

(f) $H(z) = \dfrac{1+z^{-1}}{1-0.5z^{-1}+0.06z^{-2}}$

9.24 The transfer function of a causal IIR filter is given by

$$H(z) = \frac{5z(3z-2)}{(z+0.5)(2z-1)}$$

Determine the values of the multiplier coefficients of the realisation structure shown in Fig. Q9.24.

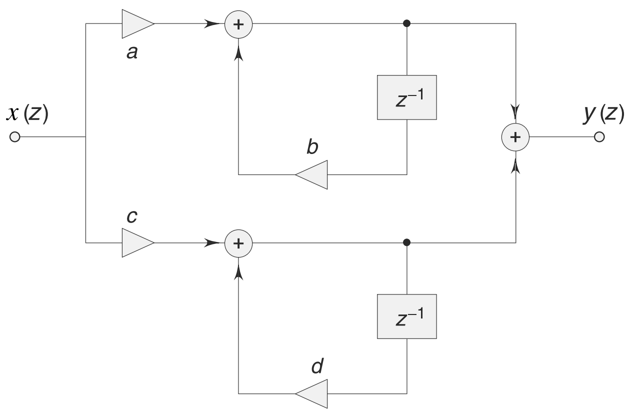


Fig. Q9.24

Ans: $a = 35/4$, $b = -1/2$, $c = -5/4$, $d = 1/2$

9.25 Realise the FIR transfer function $H(z) = (1 + 0.6z^{-1})^5$ in (i) two different direct forms, (ii) cascade of first-order sections, (iii) cascade of one first-order and two second-order sections, and (iv) cascade of one second-order and one third-order sections.

9.26 Realise the following system functions using a minimum number of multipliers.

(a) $H(z) = \left(1+\frac{1}{2}z^{-1} - z^{-2}\right)\left(1-\frac{1}{4}z^{-1} + z^{-2}\right)$ (b) $H(z) = 1+\frac{1}{2}z^{-1} + \frac{1}{4}z^{-2} + z^{-3}$

9.27 Obtain the system function $H(z)$ for each of the systems shown in Fig. Q9.27.

Ans: (a) $H(z) = 8\frac{\left(1+\frac{1}{2}z^{-1}\right)}{\left(1-\frac{2}{3}z^{-1}\right)}\cdot\frac{1+1.2z^{-1}+\frac{7}{8}z^{-2}}{1-0.83z^{-1}-0.76z^{-2}}$

(b) $H(z) = 10z^{-2}\cdot\frac{\left(1+5z^{-1}\right)}{1-0.2z^{-1}}\cdot\frac{1}{\left(1+1.5z^{-1}\right)}\cdot\frac{1}{\left(1+0.8z^{-1}\right)}$

(c) $H(z) = 31-4z^{-1}+\frac{2}{3}\cdot\frac{1}{\left(1+2z^{-1}\right)}$

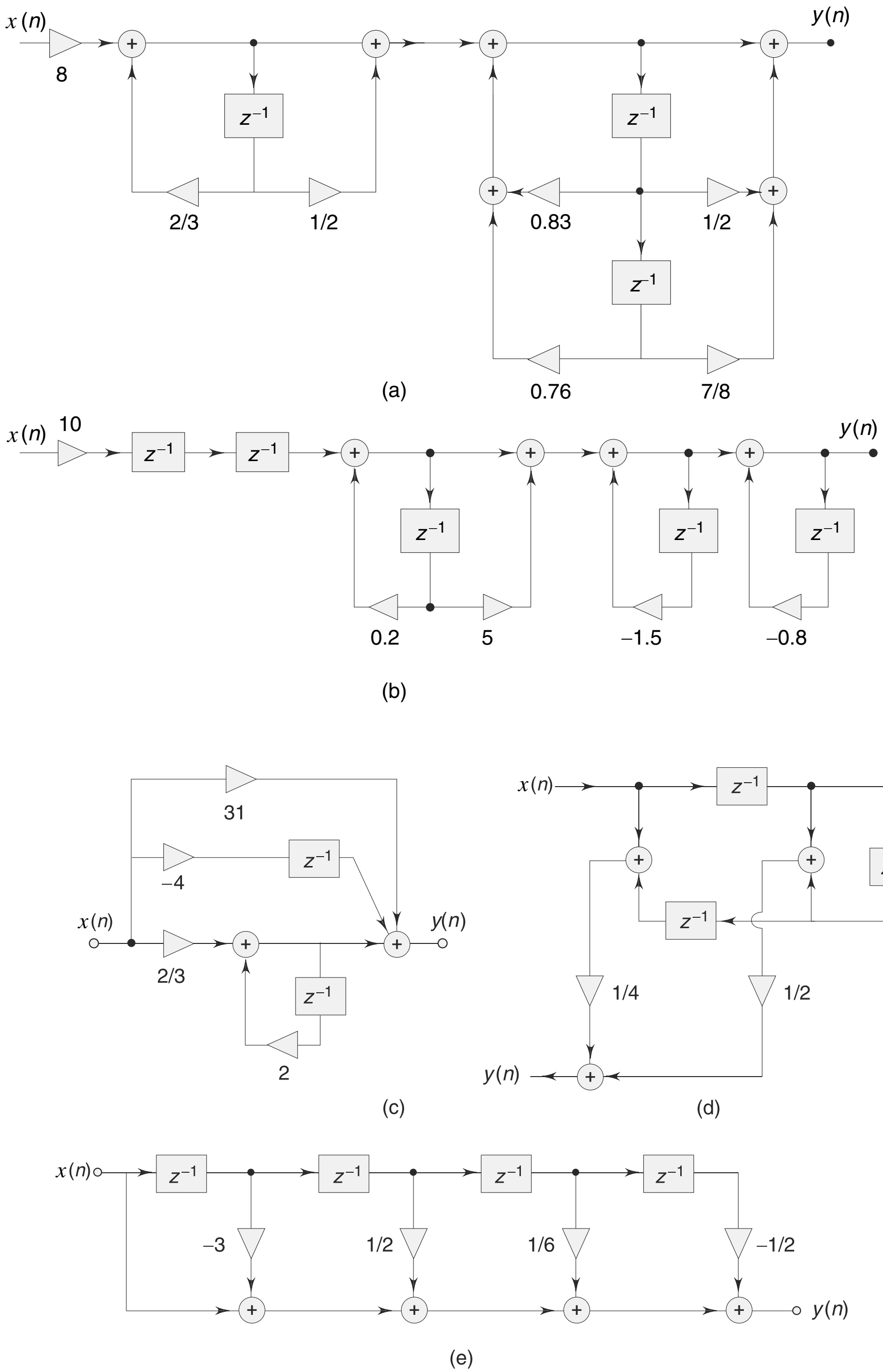
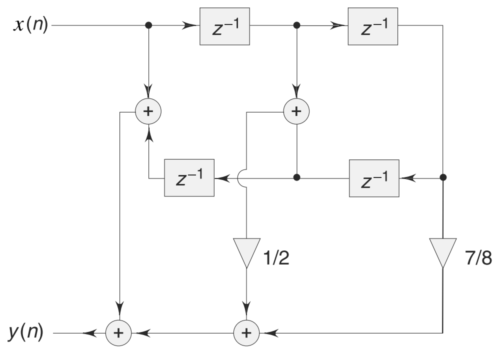
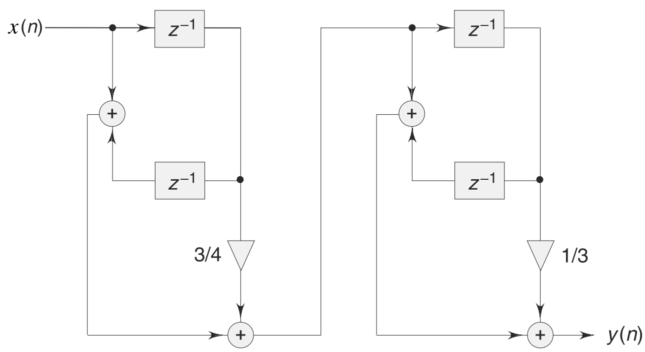


Fig. Q9.27 *(Contd.)*



(f)



(g)

Fig. Q9.27

9.28 Realise an FIR filter with impulse response $h(n)$ given by

$$h(n)=\left(\frac{1}{2}\right)^n\left[u(n)-u(n-5)\right]$$

9.29 Realise the linear phase filter with the impulse response

$$h(n)=\delta(n)+\frac{1}{2}\delta(n-1)-\frac{1}{4}\delta(n-2)+\frac{1}{2}\delta(n-3)+\delta(n-4)$$

9.30 A system is represented by its transfer function $H(z)$ given by

$$H(z)=4+\frac{3z}{\left(z-\frac{1}{2}\right)}-\frac{1}{\left(z-\frac{1}{4}\right)}$$

(a) Does this $H(z)$ represent an FIR or an IIR filter?

(b) Realise the system function in direct forms I and II and give the governing equations for implementation.

9.31 Sketch the ladder structure for the system $H(z)=\dfrac{1-0.6z^{-1}+1.2z^{-2}}{1+0.15z^{-1}-0.64z^{-2}}$ Also check whether the system is stable.

9.32 Determine the state space model for the system defined by

$$y(n) = 0.8\, y(n-1) + 0.12\, y(n-2) + x(n)$$

and sketch the state space structure.

9.33 Obtain the ladder structure for $H(z)=\dfrac{1}{z^{-3}+2z^{-2}+2z^{-1}+1}$.

Chapter 10

Effects of Finite Word Length in Digital Filters

INTRODUCTION 10.1

When digital systems are implemented either in hardware or in software, the filter coefficients are stored in binary registers. These registers can accommodate only a finite number of bits and hence, the filter coefficients have to be truncated or rounded-off in order to fit into these registers. Truncation or rounding of the data results in degradation of system performance. Also, in digital processing systems, a continuous-time input signal is sampled and quantised in order to get the digital signal. The process of quantisation introduces an error in the signal. The various effects of quantisation that arise in digital signal processing are discussed in this chapter. Following are some of the issues connected with finite word length effects.

(i) Quantisation effects in analog-to-digital conversion
(ii) Product quantisation and coefficient quantisation errors in digital filters
(iii) Limit cycles in IIR filters, and
(iv) Finite word length effects in Fast Fourier Transforms.

ROUNDING AND TRUNCATION ERRORS 10.2

Rounding or truncation introduces an error whose magnitude depends on the number of bits truncated or rounded-off. Also, the characteristics of the error depends on the form of binary number representation. The sign magnitude and the two's complement representation of fixed point binary numbers are considered here. Table 10.1 gives the sign magnitude and two's complement representation of fixed point numbers.

Consider a number x, whose original length is 'L' bits. Let this number be quantised (truncated or rounded) to 'B' bits, as shown below. This quantised number is represented by $Q(x)$. Both x and $Q(x)$ are shown below. Note that $B < L$.

	Sign bit	0	1	2		L-2	L-1
$x =$	X	X	X	X	$\cdots$	X	X

and

	Sign bit	0	1	2		B-2	B-1
$Q(x) =$	X	X	X	X	$\cdots$	X	X

Table 10.1 *Fixed point binary representation*

Number	*Sign magnitude*	*Two's complement*
7	0111	0111
6	0110	0110
5	0101	0101
4	0100	0100
3	0011	0011
2	0010	0010
1	0001	0001
0	0000	0000
− 0	1000	0000
− 1	1001	1111
− 2	1010	1110
− 3	1011	1101
− 4	1100	1100
− 5	1101	1011
− 6	1110	1010
− 7	1111	1001

A truncation error, ε_T, is introduced in the input signal and thus the quantised signal is

$$Q_T(x) = x + \varepsilon_T \qquad (10.1)$$

The range of values of the error due to truncation of the signal is analysed here for both sign magnitude and two's complement representations.

(*i*) ***Truncation error for sign magnitude representation*** When the input number x is positive, truncation results in reducing the magnitude of the number. Thus, the truncation error is negative and the range is given by

$$-(2^{-B} - 2^{-L}) \le \varepsilon_T \le 0 \qquad (10.2)$$

The largest error occurs when all the discarded bits are one's. When the number x is negative, truncation results in reduction of the magnitude only. However, because of the negative sign, the resulting number will be greater than the original number. For example, let the number be $x = -0.375$. That is, in sign magnitude form it is represented as $x = 1011$ and after truncation of one bit, $Q(x) = 101$. This is equivalent to -0.25 in decimal. But -0.25 is greater than -0.375. Therefore, the truncation error is positive and its range is

$$0 \le \varepsilon_T \le (2^{-B} - 2^{-L}) \qquad (10.3)$$

The overall range of the truncation error for the sign magnitude representation is

$$-(2^{-B} - 2^{-L}) \le \varepsilon_T \le (2^{-B} - 2^{-L}) \qquad (10.4)$$

(*ii*) ***Truncation error for two's complement representation*** When the input number is positive, truncation results in a smaller number, as in the case of sign magnitude numbers. Hence, the truncation error is negative and its range is same as that given in Eq. (10.2). If the number is negative, truncation of the number in two's complement form results in a smaller number and the error is negative. Thus the complete range of the truncation error for the two's complement representation is

$$-(2^{-B} - 2^{-L}) \le \varepsilon_T \le 0 \qquad (10.5)$$

(*iii*) *Round-off error for sign magnitude and two's complement representation* The rounding of a binary number involves only the magnitude of the number and is independent of the type of fixed-point binary representation. The error due to rounding may be either positive or negative and the peak value is $\frac{(2^{-B}-2^{-L})}{2}$. The round-off error is symmetric about zero and its range is

$$-\frac{(2^{-B}-2^{-L})}{2} \le \varepsilon_R \le \frac{(2^{-B}-2^{-L})}{2} \tag{10.6}$$

where ε_R is the round-off error.

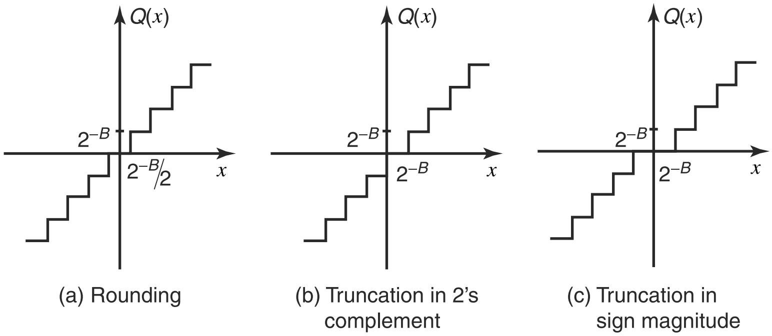


Fig. 10.1 *Quantisation Error in Rounding and Truncation*

In most cases infinite precision is assumed, i.e. the length of the unquantised number is assumed to be infinity and as a result, Eqs. (10.4), (10.5) and (10.6) can be modified and the range of values of error for different cases are as follows.

(i) Truncation error for sign magnitude representation

$$-2^{-B} \le \varepsilon_T \le 2^{-B} \tag{10.7}$$

(ii) Truncation error for two's complement representation

$$-2^{-B} \le \varepsilon_T \le 0 \tag{10.8}$$

(iii) Round-off error for sign magnitude and two's complement representation

$$-\frac{2^{-B}}{2} \le \varepsilon_R \le \frac{2^{-B}}{2} \tag{10.9}$$

Figure 10.1 shows the quantisation errors in truncation and rounding of numbers. In computations involving quantisation, a statistical approach is used in the characterisation of these errors.

The quantising unit is modelled as one that introduces an additive noise to the unquantised value of the number x.

$$Q(x) = x + \varepsilon \tag{10.10}$$

where $\varepsilon = \varepsilon_R$ for rounding and $\varepsilon = \varepsilon_T$ for truncation. The number x may fall within any of the levels of the quantiser and the quantisation error can be modelled as a random variable. This error is assumed to

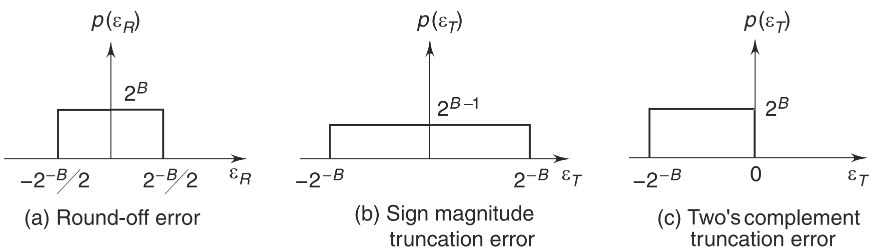


Fig. 10.2 *Probabilistic Characteristics of Quantisation Errors*

be uniformly distributed over the range specified by Eqs. (10.7), (10.8) and (10.10). The probability density functions of these errors are shown in Fig. 10.2. Usually, rounding is used to perform quantisation because of its desirable properties. The error signal due to rounding is independent of the type of binary representation and its mean is zero. Rounding yields lower variance than truncation or any other method.

10.3 QUANTISATION EFFECTS IN ANALOG-TO-DIGITAL CONVERSION OF SIGNALS

The process of analog-to-digital conversion involves (i) sampling the continuous time signal at a rate much greater than the Nyquist rate, and (ii) quantising the amplitude of the sampled signal into a set of discrete amplitude levels. The input-output characteristics of a uniform quantiser is shown in Fig. 10.3. This quantiser rounds the sampled signal to the nearest quantised output level. The difference between the quantised signal amplitude $x_Q(n)$ and the actual signal amplitude $x(n)$ is called the quantisation error $e(n)$. That is

$$e(n) = x_Q(n) - x(n) \tag{10.11a}$$

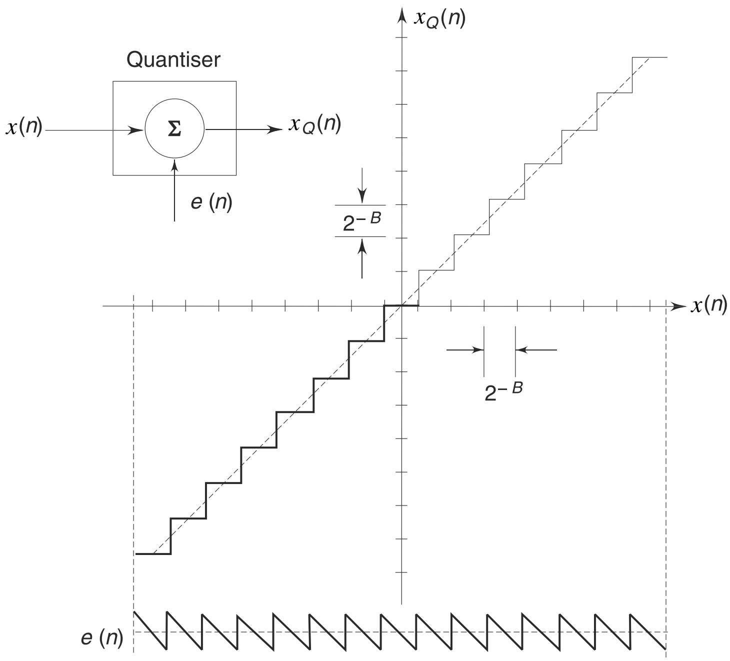


Fig. 10.3 *Two's Complement Number Quantisation*

Since rounding is involved in the process of quantisation, the range of values for this quantisation error is

$$-\frac{2^{-B}}{2} \le e(n) \le \frac{2^{-B}}{2} \tag{10.11b}$$

The quantisation error is assumed to be uniformly distributed over the range specified by Eq. (10.11b). It is also assumed that this quantisation noise $\{e(n)\}$ is a stationary white noise sequence and it is uncorrelated with the signal sequence $x(n)$. The above assumptions are valid if the quantisation step size is small and the signal sequence $x(n)$ traverses several quantisation levels between two successive samples. Based on these assumptions, the quantisation error sequence $\{e(n)\}$ can be modelled as an additive noise affecting the desired signal $x(n)$. The signal-to-noise power ratio (*SNR*) can be evaluated to quantify the effect of the additive noise $e(n)$. The *SNR* is given by

$$SNR = 10\log\frac{P_{x(n)}}{P_{e(n)}} \tag{10.12}$$

where $P_{x(n)}$ is the signal power and $P_{e(n)}$ is the power of the quantised noise.

In the process of quantisation, the sampled value is rounded-off to the nearest quantisation level. The probability density function for the quantisation round-off error in A/D conversion is shown in Fig. 10.4.

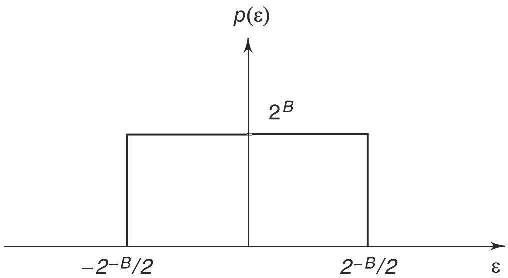


Fig. 10.4 *Probability Density Function for the Quantisation Round-off Error in A/D Conversion*

It can be noted from Fig. 10.4 that the quantisation error is uniformly distributed and the mean value of the error is zero. The power of the quantisation noise, which is nothing but the variance (σ_e^2) is given by

$$P_{e(n)} = \sigma_e^2 = \int_{\frac{2^{-B}}{2}}^{\frac{2^{-B}}{2}} e^2 p(\varepsilon)\,\mathrm{d}e$$

$$P_{e(n)} = \frac{1}{2^{-B}} \int_{-2^{-B-1}}^{2^{-B-1}} e^2\,\mathrm{d}e$$

$$= \frac{2^{-2B}}{12} \tag{10.13}$$

Therefore,

$$SNR = 10\log\frac{P_{x(n)}}{P_{e(n)}}$$

$$= 10\log P_{x(n)} + 10\log(12) + 10\log(2^{2B})$$

$$= 10\log P_{x(n)} + 10.8 + 6B \tag{10.14}$$

Equation 10.14 indicates that each bit used in *A/D* conversion or in the quantiser, increases the signal-to-noise power ratio by 6 dB. Alternatively, from Eq. (10.13),

$$10\log P_{e(n)} = 10\log\frac{2^{-2B}}{12} = 10\log(2^{-2B}) - 10\log 12$$

$$= -6B - 10.8 \tag{10.15}$$

From Eq. (10.15), it can be noted that each additional bit accommodated in the quantisation process reduces the quantisation noise power by 6 dB. Equation (10.15) can be used to determine the dynamic range of the quantiser for a sinusoidal signal. The dynamic range (*DR*) is expressed as

$$DR = -10 \log P_{e(n)} = 6B + 10.8 \tag{10.16}$$

From Eq. (10.16), the number of binary bits that must be used to represent the data when the dynamic range is specified can be determined.

OUTPUT NOISE POWER FROM A DIGITAL SYSTEM 10.4

After converting the continuous time signal into a digital signal, let us assume that this quantised signal is applied as an input to a digital system with system function $H(z)$. This quantised input to the digital system consists of two components, namely, (i) the unquantised input signal $x(n)$ and (ii) the quantisation error signal $e(n)$. The output of the digital system, therefore, consists of two components, namely, (i) the output $y_Q(n)$, due to the quantised input signal, and (ii) the error output $e_0(n)$ due to the quantisation error signal at the input of the digital system. From Fig. 10.5, it can be seen that the output $Y(n)$ of the digital system is given by

$$Y(n) = y_Q(n) + e_o(n) \tag{10.17}$$

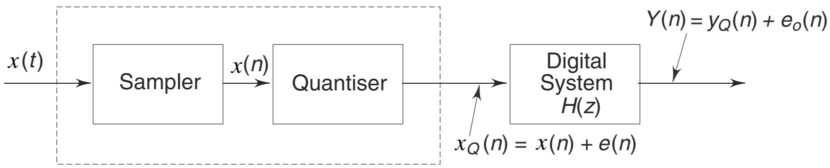


Fig. 10.5 *Output Quantisation Noise of a Digital System*

The error output $e_o(n)$ is a random process and it is the response of the digital system to the input error signal $e(n)$. The digital system is assumed to be causal. The error output $e_o(n)$ is obtained by convolving the system impulse response $h(n)$ with input error signal $e(n)$. Thus,

$$e_o(n) = \sum_{k=0}^{\infty} h(k)e(n-k) \tag{10.18}$$

Let us relate the statistical characteristics of the output error signal to the statistical characteristics of the input error signal and the characteristics of the system. The autocorrelation sequence for the output error signal $e_o(n)$ is

$$\gamma_{eo\,eo}(m) = E\left[e_o^*(n) \cdot e_o(n+m)\right] \tag{10.19}$$

where E represents the statistical expectation. Using Eq. (10.18) in Eq. (10.19), we get

$$\gamma_{eo\,eo}(m) = E\left[\sum_{k=0}^{\infty} h(k)e^*(n-k) \cdot \sum_{k=0}^{\infty} h(k)e(n+m-k)\right]$$

$$= \sum_{k=0}^{\infty} h^2(k) E\left[e^*(n-k) \cdot e(n+m-k)\right]$$

$$\gamma_{eo\,eo}(m) = \sum_{k=0}^{\infty} h^2(k)\gamma_{ee}(m) \tag{10.20}$$

It has been assumed that the noise resulting from the quantisation process is a white noise. For this case, we have

$$\gamma_{eoeo}(m) = \sigma_{eo}^2 \quad \text{and} \quad \gamma_{ee}(m) = \sigma_e^2 \tag{10.21}$$

where σ_{eo}^2 is the output noise power (or power of the output error) and σ_e^2 is the input noise power. Using Eq. (10.21) in Eq. (10.20) and replacing the variable k with n,

$$\sigma_{eo}^2 = \sigma_e^2 \sum_{n=0}^{\infty} h^2(n) \tag{10.22}$$

Using Parseval's relation (see Example 10.1),

$$\sum_{n=0}^{\infty} h^2(n) = \frac{1}{2\pi j} \oint_C H(z) H\left(z^{-1}\right) z^{-1} \mathrm{d}z$$

In Eq. (10.22), we get

$$\sigma_{eo}^2 = \frac{\sigma_e^2}{2\pi j} \oint_C H(z)H(z^{-1})z^{-1}\,\mathrm{d}z \tag{10.23}$$

where the closed contour of integration is around the unit circle $|z| = 1$. This integration is evaluated using the method of residues, taking only the poles that lie inside the unit circle.

Example 10.1 *Prove that* $\sum_{n=0}^{\infty} x^2(n) = \frac{1}{2\pi j} \oint_C X(z)X(z^{-1})z^{-1}\,\mathrm{d}z$

Solution The z-transform of $x(n)$ is

$$X(z) = \sum_{n=0}^{\infty} x(n) z^{-n} \tag{1}$$

Taking the z-transform of $x^2(n)$,

$$Z\left[x^2(n)\right] = \sum_{n=0}^{\infty} x(n)x(n)z^{-n} = \sum_{n=0}^{\infty} x^2(n)z^{-n} \tag{2}$$

The integral formula for the inverse z-transform is given by

$$Z^{-1}\left[X(z)\right] = x(n) = \frac{1}{2\pi j}\oint_C X(z)z^{n-1}\mathrm{d}z$$

Using the above formula for one of the $x(n)$ in Eq. (2),

$$Z\left[x^2(n)\right] = \sum_{n=0}^{\infty} x^2(n)z^{-n} = \sum_{n=0}^{\infty}\left[\frac{1}{2\pi j}\oint_C X(z)z^{n-1}\mathrm{d}z\right]x(n)z^{-n}$$

Interchanging the order of summation and integration,

$$\sum_{n=0}^{\infty} x^2(n)z^{-n} = \frac{1}{2\pi j}\oint_C X(z)\left[\sum_{n=0}^{\infty} x(n)z^{-1}\right]\mathrm{d}z$$

That is,

$$\sum_{n=0}^{\infty} x^2(n) = \frac{1}{2\pi j}\oint_C X(z)\left[\sum_{n=0}^{\infty} x(n)\left(z^{-1}\right)^{-n}\right] z^{-1}\mathrm{d}z$$

Using the definition of z-transform, the above expression can be written as,

$$\sum_{n=0}^{\infty} x^2(n) = \frac{1}{2\pi j}\oint_C X(z)X(z)^{-1}z^{-1}\mathrm{d}z$$

Thus, the expression is obtained. The above expression is a form of the Parseval's relation.

Example 10.2 *The output of an A/D converter is applied to a digital filter with the system function.*

$$H(z)=\frac{0.5z}{z-0.5}$$

Find the output noise power from the digital filter, when the input signal is quantised to have eight bits.

Solution The quantisation noise power is

$$\sigma_e^2=\frac{2^{-2B}}{12}=\frac{2^{-16}}{12}=1.27\times10^{-6}$$

The output noise power is given by

$$\sigma_{eo}^2=\frac{\sigma_e^2}{2\pi j}\oint_C H(z)H(z^{-1})z^{-1}\mathrm{d}z$$

That is,

$$\sigma_{eo}^2=\frac{\sigma_e^2}{2\pi j}\oint_C\left(\frac{0.5z}{z-0.5}\right)\left(\frac{0.5z^{-1}}{z^{-1}-0.5}\right)z^{-1}\mathrm{d}z=\frac{\sigma_e^2}{2\pi j}\oint_C\frac{0.25}{(z-0.5)(1-0.5z)}\mathrm{d}z$$

$$\sigma_{eo}^2=\sigma_e^2\cdot I$$

where $$I=\frac{1}{2\pi j}\oint_C\frac{0.25}{(z-0.5)(1-0.5z)}\mathrm{d}z$$

The above integral can be evaluated by the method of residues.

I = sum of the residues at the poles within the unit circle, i.e. within $|z|<1$. The poles are at $z=0.5$ and $z=2$. Therefore,

I = residue at $z=0.5$

$$I=(z-0.5)\left.\frac{0.25}{(z-0.5)(1-0.5z)}\right|_{z=0.5}=\frac{1}{3}$$

Therefore,

$$\sigma_{eo}^2=\sigma_e^2\cdot I=\frac{1}{3}\sigma_e^2=\frac{1.27\times10^{-6}}{3}=0.423\times10^{-6}$$

COEFFICIENT QUANTISATION EFFECTS IN DIRECT FORM REALISATION OF IIR FILTERS 10.5

The realisation of digital filters in hardware or software has some limitations due to the finite word length of the registers that are available to store these filter coefficients. Since the coefficients stored in these registers are either truncated or rounded-off, the system that is realised using these coefficients is not accurate. The location of poles and zeros of the resulting system will be different from the original locations and consequently, the system may have a different frequency response than the one desired.

In general, there are two approaches for the analysis and synthesis of digital filters with quantised coefficients. In the first approach, the quantisation error is treated as a statistical quantity. In this case, the digital filter is assumed to be an ideal one and represented by a transfer function in parallel with this ideal filter. In the second approach, each filter is studied separately and the quantised coefficients can be optimised to minimise the maximum weighted difference between the ideal and actual frequency responses. This method yields the best finite precision representation of the desired frequency response.

Let the transfer function of the coefficient-quantised digital filter being realised be

$$H(z)=\frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} \tag{10.24}$$

If $\bar{a}_k$ and $\bar{b}_k$ are the filter coefficients of unquantised filter (ideal) and α_k and β_k are the error quantities (assumed to be statistically independent, uniformly distributed random variables), the filter coefficients of the quantised filter can be written as

$$a_k = \bar{a}_k + \alpha_k \text{ and } b_k = \bar{b}_k + \beta_k \tag{10.25}$$

The error due to quantisation of the filter coefficients is

$$e(n) = y'(n) - y(n) \tag{10.26}$$

where $y'(n)$ is the output of the actual filter (finite precision) and $y(n)$ is the output of the ideal filter (infinite precision). If $x(n)$ is the input to the filter, then Eq. (10.26) can be written as

$$e(n)=\left[\sum_{k=0}^{M} b_k x(n-k)-\sum_{k=1}^{N} a_k y'(n-k)\right]-\left[\sum_{k=0}^{M} \bar{b}_k x(n-k)-\sum_{k=1}^{N} \bar{a}_k y(n-k)\right]$$

$$=\sum_{k=0}^{M} b_k x(n-k)-\sum_{k=0}^{M} \bar{b}_k x(n-k)\;-\sum_{k=1}^{N} a_k y'(n-k)+\sum_{k=1}^{N} \bar{a}_k y(n-k) \tag{10.27}$$

Using Eq. (10.25) in Eq. (10.27),

$$e(n)=\sum_{k=0}^{M} \beta_k x(n-k)-\sum_{k=1}^{N} (\bar{a}_k+\alpha_k) y'(n-k)+\sum_{k=1}^{N} \bar{a}_k y(n-k)$$

$$=\sum_{k=0}^{M} \beta_k x(n-k)-\sum_{k=1}^{N} \bar{a}_k \left[y'(n-k)-y(n-k)\right]-\sum_{k=1}^{N} \alpha_k y'(n-k)$$

Using Eq. (10.26),

$$e(n)=\sum_{k=0}^{M} \beta_k x(n-k)-\sum_{k=1}^{N} \bar{a}_k e(n-k)-\sum_{k=1}^{N} \alpha_k \left[e(n-k)+y(n-k)\right]$$

$$e(n)=\sum_{k=0}^{M} \beta_k x(n-k)-\sum_{k=1}^{N} \bar{a}_k e(n-k)\;-\sum_{k=1}^{N} \alpha_k e(n-k)-\sum_{k=1}^{N} \alpha_k y(n-k) \tag{10.28}$$

Neglecting the second order quantities, i.e. $\sum_{k=1}^{N} \alpha_k e(n-k)$, Eq. (10.28) becomes

$$e(n)=\sum_{k=0}^{M} \beta_k x(n-k)-\sum_{k=1}^{N} \bar{a}_k e(n-k)-\sum_{k=1}^{N} \alpha_k y(n-k) \tag{10.29}$$

Taking z-transform of Eq. (10.29),

$$E(z)=X(z)\sum_{k=0}^{M} \beta_k z^{-k}-E(z)\sum_{k=1}^{N} \bar{a}_k z^{-k}-Y(z)\sum_{k=1}^{N} \alpha_k z^{-k}$$

That is,

$$E(z)\left(1+\sum_{k=1}^{N}\bar{a}_k z^{-k}\right)=X(z)\sum_{k=0}^{M}\beta_k z^{-k}-Y(z)\sum_{k=1}^{N}\alpha_k z^{-k} \tag{10.30}$$

The output $Y(z)$ of the ideal filter is given by

$$Y(z) = H_{ideal}(z)\cdot X(z)$$

Using Eq. (10.30),

$$E(z)\left(1+\sum_{k=1}^{N}\bar{a}_k z^{-k}\right)=X(z)\left(\sum_{k=0}^{M}\beta_k z^{-k}-H_{ideal}(z)\sum_{k=1}^{N}\alpha_k z^{-k}\right)$$

Therefore,

$$E(z)=X(z)\left(\frac{\sum_{k=0}^{M}\beta_k z^{-k}-H_{ideal}(z)\sum_{k=1}^{N}\alpha_k z^{-k}}{1+\sum_{k=1}^{N}\bar{a}_k z^{-k}}\right) \tag{10.31}$$

From Eqs. (10.27), (10.30) and (10.31), the output of the actual filter is given by

$$Y'(z)=X(z)\left(H_{ideal}(z)+\frac{\sum_{k=0}^{M}\beta_k z^{-k}-H_{ideal}(z)\sum_{k=1}^{N}\alpha_k z^{-k}}{1+\sum_{k=1}^{N}\bar{a}_k z^{-k}}\right)$$

That is,

$$Y'(z) = [H_{ideal}(z)\, X(z) + E(z)] \tag{10.32}$$

From Eq. (10.32), it can be noted that the actual filter may be considered as an ideal filter (unquantised) in parallel with a stray transfer function due to the quantisation error as shown in Fig. 10.6. Similar formulae can be obtained for parallel and cascade realisation of IIR filters.

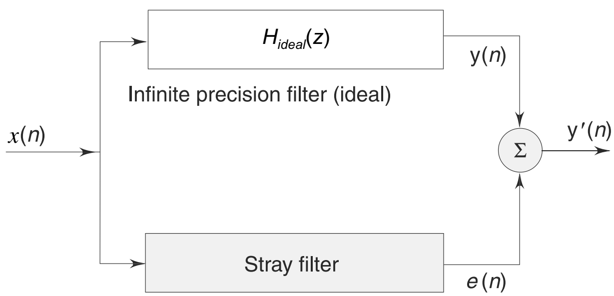


Fig. 10.6 *Coefficient Quantisation Model for Direct Form Realisation of IIR Filter*

10.6 COEFFICIENT QUANTISATION IN DIRECT FORM REALISATION OF FIR FILTERS

The statistical bounds on the error in the frequency response due to coefficient quantisation (rounding) is given here. The frequency response of a linear phase FIR filter is given by

$$H\left(e^{j\omega}\right)=e^{-j\omega(M-1)/2}\left\{h\left(\frac{M-1}{2}\right)+2\sum_{n=0}^{(M-3)/2}h(n)\cos\left[\left(\frac{M-1}{2}-n\right)\omega\right]\right\}$$

$$= e^{j\Phi(\omega)}M(\omega) \tag{10.33}$$

For a linear phase FIR filter, $h(n) = h(M-1-n)$. The term $e^{j\Phi(\omega)} = e^{-j\omega(M-1)/2}$ in the above expression represents the delay and is unaffected by quantisation. Hence, the quantisation effect is solely on the pseudomagnitude term $M(\omega)$. Let $\{h_q(n)\}$ be the sequence resulting from rounding $\{h(n)\}$ to a quantisation step size of 2^{-B}. Therefore,

$$h_q(n) = h(n) + e(n) \tag{10.34}$$

and $h_q(n) = h_q(M-1-n)$ for $0 \le n \le (M-1)/2$. Let $e(n)$ be a random sequence and uniformly distributed over the range $-\frac{2^{-B}}{2}$ and $\frac{2^{-B}}{2}$. Let $H_q(z)$ be the z-transform of $\{h_q(n)\}$ and $M_q(\omega)$ be the pseudomagnitude of the quantised linear phase FIR filter. The error function is defined to be

$$E(e^{j\omega}) = M_q(\omega) - M(\omega) \tag{10.35}$$

$$\text{or } E\left(e^{j\omega}\right) = e\left(\frac{M-1}{2}\right) + 2\sum_{n=0}^{(M-3)/2} e(n)\cos\left[\left(\frac{M-1}{2}-n\right)\omega\right]$$

where $E(e^{j\omega})$ is the frequency response of a linear phase FIR filter that has $\{e(n)\}$ as the impulse response for the first half and the second-half can be obtained using $e(n) = e(M-1-n)$. Thus, the filter with its coefficients rounded-off can be considered as a parallel connection of the ideal filter (infinite precision) with a filter whose frequency response is $E(e^{j\omega})e^{-j\omega(M-1)/2}$. Since the error $e(n)$ due to rounding of the filter coefficients is always lesser than or equal to $\frac{2^{-B}}{2}$, a bound on $|E(e^{j}\omega)|$ can be obtained as shown below.

$$\left|E(e^{j\omega})\right| \le \left|e\left(\frac{M-1}{2}\right)\right| + 2\sum_{n=0}^{(M-3)/2} |e(n)|\left|\cos\left[\left(\frac{M-1}{2}-n\right)\omega\right]\right| \tag{10.36}$$

Letting $\frac{M-1}{2} - n = k$ in the second term of the above expression

$$\left|E(e^{j\omega})\right| \le \left|e\left(\frac{M-1}{2}\right)\right| + 2\sum_{k=1}^{(M-1)/2} \left|e\left(\frac{M-1}{2}-k\right)\right| |\cos k\omega|$$

That is,

$$\left|E\left(e^{j\omega}\right)\right| \le \frac{2^{-B}}{2}\left[1 + 2\sum_{k=1}^{\frac{M-1}{2}} \left|\cos(k\omega)\right|\right] \tag{10.37}$$

or

$$\left|E\left(e^{j\omega}\right)\right| \le M\frac{2^{-B}}{2} \tag{10.38}$$

A more useful statistical bound than that given by Eq. (10.38) can be obtained if it is assumed that the errors due to quantisation of different coefficients are statistically independent. This statistical bound provides the filter designer a means of predicting how much accuracy is required for the coefficients to obtain a desired frequency response without a priori knowledge of the value of the filter coefficients. From Eq. (10.36), an expression for the power of the error signal, $\overline{E(e^{j\omega})^2}$, is obtained.

$$\sigma^2_{E(\omega)} = \overline{E\left(e^{j\omega}\right)^2} = \overline{e\left(\frac{M-1}{2}\right)^2} + \sum_{n=0}^{(M-3)/2} \overline{4e(n)^2}\cos^2\left[\left(\frac{M-1}{2}-n\right)\omega\right] \tag{10.39}$$

For uniform distribution, $\overline{e(n)^2} = \sigma^2_{e(n)} = \dfrac{2^{-2B}}{12}$. Therefore,

$$\sigma^2_{E(\omega)} = \overline{E\left(e^{j\omega}\right)^2} = \frac{2^{-2B}}{12}\left[1+4\sum_{n=1}^{\frac{M-1}{2}}\cos^2(\omega n)\right] \tag{10.40}$$

Defining a weighting function $W(\omega)$ as

$$W(\omega) = \left\{\frac{1}{2M-1}\left[1+4\sum_{n=1}^{\frac{M-1}{2}}\cos^2(\omega n)\right]\right\}^{\frac{1}{2}} \tag{10.41}$$

The standard deviation of the error is given by

$$\sigma_{E(\omega)} = \sqrt{\frac{2M-1}{3}}\,\frac{2^{-B}}{2}W(\omega) \tag{10.42}$$

The value of the weighting function $W(\omega)$ lies between 0 and 1, and $W(0) = W(\pi) = 1$ for all M. Thus,

$$\sigma_{E(\omega)} \le \frac{2^{-B}}{2}\sqrt{\frac{2M-1}{3}} \tag{10.43}$$

The value of $E(e^{j\omega})$ is obtained as a summation of independent random variables, for any ω. $E(e^{j\omega})$ is a Gaussian function for large values of M. Thus, the mean and variance of $E(e^{j\omega})$ gives an excellent description of $E(e^{j\omega})$.

LIMIT CYCLE OSCILLATIONS 10.7

In Section 10.3, it was assumed that the input sequence traverses several quantisation levels between two successive samples and so the samples of the round-off noise sequence were uncorrelated with each other and with the input signal. These assumptions are invalid in cases such as a constant or zero input to a digital filter. In such cases, the input signal remains constant during successive samples and does not traverse several quantisation levels. There are two types of limit cycles, namely, zero input limit cycle and overflow limit cycle. Zero input limit cycles are usually of lower amplitudes in comparison with overflow limit cycles. Let us consider a system with the difference equation,

$$y(n) = 0.8\,y(n-1) + x(n) \tag{10.44}$$

with zero input, i.e. $x(n) = 0$ and initial condition $y(-1) = 10$. A comparison between the exact values of $y(n)$ as given by Eq. (10.44) using unquantised arithmetic and the rounded values of $y(n)$ as obtained from quantised arithmetic are given in Table 10.2.

Table 10.2 *A Comparison of exact y(n) and rounded y(n)*

n	*y(n)-unquantised*	*y(n)-quantised*
−1	10.0	10
0	8.0	8
1	6.4	6
2	5.12	5
3	4.096	4
4	3.2768	3
5	2.62144	2
6	2.0972	2
7	1.6772	2

From Table 10.2, it can be observed that for zero input, the unquantised output $y(n)$ decays exponentially to zero with increasing n. However, the rounded-off (quantised) output $y(n)$ gets stuck at a value of two and never decays further. Thus, the output is finite even when no input is applied. This is referred to as zero input limit cycle effect. It can also be seen that for any value of the input condition $|y(-1)| \leq 2$, the output $y(n) = y(-1)$, $n \geq 0$, when the input is zero. Thus, the deadband in this case is the interval $[-2, 2]$.

The effects of limit cycles in first-order and second-order systems were studied by Jackson using an "effective value" model. It was realised that limit cycles can occur only if the result of rounding leads to poles on the unit circle.

Consider the first-order difference equation

$$y(n) = x(n) - [ay(n-1)]^* \tag{10.45}$$

where $[.]^*$ denotes rounding to the nearest integer with $x(n) = 0$, $n \geq 0$. The deadband in which limit cycles can exist is the range $[-l, l]$, where l is the largest integer satisfying,

$$l \leq \frac{0.5}{1-|\alpha|} \tag{10.46}$$

If a is negative, the limit cycle will have constant magnitude and sign. If a is positive, the limit cycle will have constant magnitude by alternating sign.

Consider the second-order system with the difference equation

$$Y(n) = x(n) - [b_1 y(n-1)]^* - [b_2 y(n-2)]^* \tag{10.47}$$

The deadband for this system is the region $[-l, l]$ where l is the largest integer satisfying

$$l \leq \frac{0.5}{1-|b_2|}\left(0 < b_2 < 1\right) \tag{10.48}$$

Overflow Limit Cycles

Limit cycles can also occur due to overflow in digital filters implemented with finite precision arithmetic. The amplitudes of such overflow oscillations are much more serious in nature than zero input limit cycle oscillations. Consider a causal, all-pole second-order IIR digital filter implemented using two's complement arithmetic with a rounding of the sum of the products by a single quantiser. The difference equation describing the system is given by

$$\tilde{y}(n) = Q_R[-a_1\tilde{y}(n-1) - a_2\tilde{y}(n-2) + x(n)]$$

where $Q_R[.]$ represents the rounding operation and $\tilde{y}(n)$ is the actual output of the filter. The filter coefficients are represented by signed 4-bit fractions. Let $a_1 = 1_\Delta 001 = -0.8751_d$ and $a_2 = 0_\Delta 111 = 0.875_d$ and the initial conditions be $\tilde{y}(-1) = 0_\Delta 110 = 0.75_d$ and $\tilde{y}(-2) = 1_\Delta 010 = -0.75_d$. For zero input, i.e. $x(n) = 0$ and for $n \geq 0$, we get the values for $\tilde{y}(n)$ as shown below.

For $n = 1$, the sum of two products has resulted in a carry bit to the left of the sign bit that is automatically lost, resulting in a positive number. The same thing happens for $n = 3$ and also for other values of n. It can be noted from the above table that the output swings between positive and negative values and the swing of oscillations is also large. Such limit cycles are referred to as overflow limit cycle oscillations.

The study of limit cycles is important for two reasons. In a communication environment, when no signal is transmitted, limit cycles can occur which are extremely undesirable. For example, in a telephone no one would like to hear unwanted noise when no signal is put in from the other end. Consequently, when digital filters are used in telephone exchanges, care must be taken regarding this problem. The second reason for studying limit cycles is that this effect can be effectively used in digital waveform generators. By producing desirable limit cycles in a reliable manner, these limit cycles can be used as a source in digital signal processing.

Table 10.3

n	$\tilde{y}(n-1)$	$\tilde{y}(n-2)$	$-a_1\,\tilde{y}(n-1) - a_2\,\tilde{y}(n-2)$	$\tilde{y}(n)=Q_R[.]$	$\tilde{y}(n)$ *in decimal*
0	$\tilde{y}(-1) = 0_\Delta 110$	$\tilde{y}(-2) = 1_\Delta 010$	$1_\Delta 010100$	$1_\Delta 011$	−0.625
1	$\tilde{y}(0) = 1_\Delta 011$	$\tilde{y}(-1) = 0_\Delta 110$	$10_\Delta 110011$	$0_\Delta 110$	+0.75
2	$\tilde{y}(1) = 0_\Delta 110$	$\tilde{y}(0) = 1_\Delta 011$	$1_\Delta 001101$	$1_\Delta 010$	−0.75
3	$\tilde{y}(2) = 1_\Delta 010$	$\tilde{y}(1) = 0_\Delta 110$	$10_\Delta 101100$	$0_\Delta 110$	+0.75
4	$\tilde{y}(3) = 0_\Delta 110$	$\tilde{y}(2) = 1_\Delta 010$	$1_\Delta 010100$	$1_\Delta 011$	−0.625

Example 10.3 *A digital system is characterised by the difference equation*

$$y(n) = 0.9y\,(n-1) + x(n)$$

with $x(n) = 0$ and initial condition $y(-1) = 12$. Determine the deadband of the system.

Solution A comparison between exact values of $y(n)$ and rounded values of $y(n)$ is given below

n	$y(n)$-*exact*	$y(n)$-*rounded*
−1	12	12
0	10.8	11
1	9.72	10
2	8.748	9
3	7.8732	8
4	7.08588	7
5	6.377292	6
6	5.73956	5
7	5.16561	5
8	4.64905	5

From the above table, it is seen that for any value of $|y(-1)| \le 5$, $y(n) = y(-1)$, $n \ge 0$ for zero input. Thus, the deadband is the interval $[-5, 5]$.

Example 10.4 *The deadband of the system described in the previous example is determined here using Eq.* 10.46.

Solution The difference equation describing the system is $y(n) = 0.9y\,(n-1) + x(n)$ and the value of $a = 0.9$.

$$l \le \frac{0.5}{1-|a|} \le \frac{0.5}{1-|0.9|} = 5$$

Thus, the deadband is the interval $[-5, 5]$.

PRODUCT QUANTISATION 10.8

When two B-bit numbers are multiplied the product is $2B$-bits long. This product must be rounded to B-bits in all digital processing applications. The output of a finite word length multiplier can be expressed as

$$Q[\alpha_i x(n)] = \alpha_i x(n) + e(n) \tag{10.49}$$

where $\alpha_i x\,(n)$ is the product which is $2B$-bits long and $e(n)$ is the error resulting from rounding the product to B-bits. The fixed-point, finite word length multiplier can be modelled as shown in Fig. 10.7.

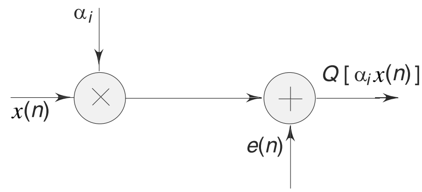


Fig. 10.7 *Product Quantisation Noise Model*

The analysis of product round-off noise is similar to A/D conversion noise. The product quantisation noise can be considered as a random process with uniform probability density function and zero mean. The average power (variance) is given by

$$\sigma_e^2 = \frac{2^{-2B}}{12} \tag{10.50}$$

The effects of rounding due to multiplication in cascaded IIR sections are discussed here. Let $h(n)$ be the system response and $e_o(n)$ be the response of the system to the input error $e(n)$. Then, as discussed in Section 10.4, the output noise power is given by

$$\sigma_{eo}^2 = \sigma_e^2 \sum_{n=0}^{\infty} h^2(n) \tag{10.51}$$

Using Parseval's relation (see Example 9.1), we get

$$\sigma_{eo}^2 = \frac{\sigma_e^2}{2\pi j} \oint_C H(z)H(z^{-1})z^{-1}\mathrm{d}z \tag{10.52}$$

Equation (10.51) or Eq. (10.52) can be used in determining the output noise power depending on whether $h(n)$ or $H(z)$ is given. Consider a cascaded second-order IIR structure shown in Fig. 10.8.

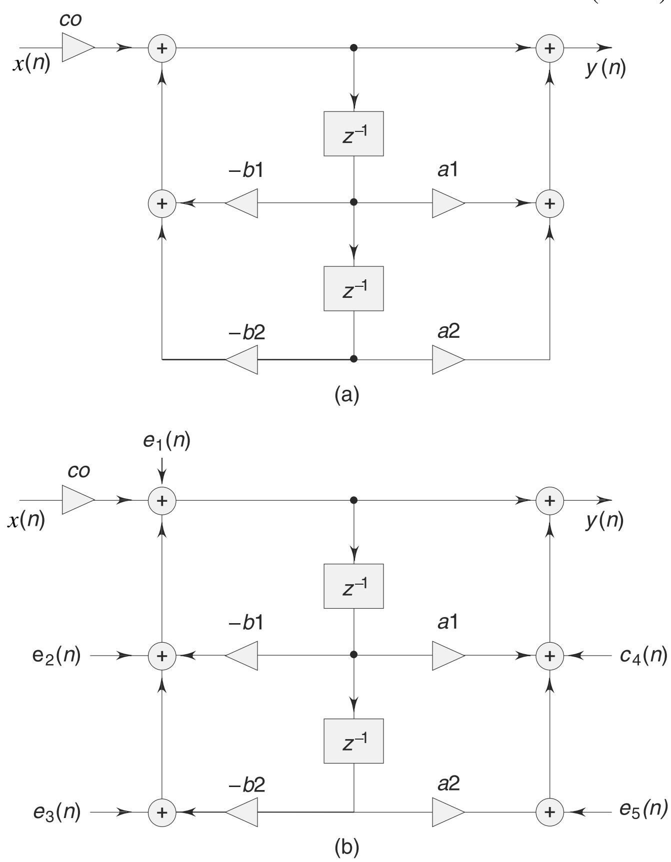


Fig. 10.8 *(a) Cascaded II-order IIR system (b) Product Quantisation Noise Model*

As shown in Fig. 10.8(b), and error signal $e_i(n)$ is added at the sum node for each multiplier. For such a case, the output noise power is given by

$$\sigma_{ek}^2 = \sigma_e^2 \sum_{n=0}^{\infty} h_i^2(n) \tag{10.53}$$

where σ^2_{ek} is the output noise power due to the k^{th} product in the i^{th} section of the cascaded structure. Thus, the overall output noise power, σ^2_{err} is given by

$$\sigma^2_{err} = \sum_{i=1}^{M} \sigma^2_{oi} \qquad (10.54)$$

where M is the total number of cascaded sections available in the system.

Example 10.5 *A cascaded realisation of the two first-order digital filters is shown below. The system functions of the individual sections are*

$$H_1(z) = \frac{1}{1-0.9z^{-1}} \text{ and } H_2(z) = \frac{1}{1-0.8z^{-1}}$$

Draw the product quantisation noise model of the system and determine the overall output noise power.

Solution The product quantisation noise model is shown in Fig. E10.2(a) and (b).

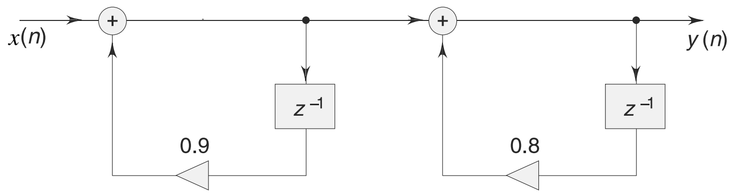


Fig. E10.5(a) *Figure for Example 10.5*

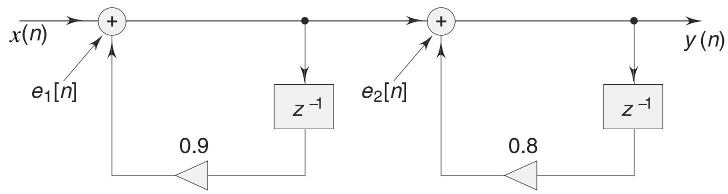


Fig. E10.5(b) *Figure for Example 10.5*

The transfer function of the system is

$$H(z) = H_1(z)\, H_2(z)$$

The noise transfer function seen by $e_1(n)$ is $H_1(z)$ and by $e_2(n)$ is $H_2(z)$. The overall output noise power is given by

$$\sigma^2_{err} = \sigma^2_{o1} + \sigma^2_{o2}$$

where, from Eq.(10.52),

$$\sigma^2_{o1} = \frac{\sigma^2_e}{2\pi j} \oint_C H(z)H(z^{-1})z^{-1}\mathrm{d}z$$

$$= \frac{\sigma^2_e}{2\pi j} \oint_C \frac{1}{\left(1-0.9z^{-1}\right)\left(1-0.8z^{-1}\right)} \times \frac{1}{\left(1-0.9z\right)\left(1-0.8z\right)} \times z^{-1}\mathrm{d}z$$

$$= \frac{\sigma^2_e}{2\pi j} \oint_C \frac{z}{(z-0.9)(z-0.8)(1-0.9z)(1-0.8z)}\mathrm{d}z$$

$$\sigma^2_{o1} = \sigma^2{}_e \times I_1$$

where

$$I_1 = \frac{1}{2\pi j}\oint_C \frac{z}{(z-0.9)(z-0.8)(1-0.9z)(1-0.8z)}\mathrm{d}z$$

The integral I_1 can be solved by the method of residues.

I_1 = sum of the residues at the poles within the unit circle.

= (residue at z = 0.9) + (residue at z = 0.8)

$$= (z-0.9)\frac{z}{(z-0.9)(z-0.8)(1-0.9z)(1-0.8)_z}\Bigg|_{z=0.9}$$

$$+(z-0.8)\frac{z}{(z-0.9)(z-0.8)(1-0.9z)(1-0.8z)}\Bigg|_{z=0.8}$$

= 89.8079

Therefore,

$$\sigma_{o1}^2 = \sigma_e^2 \times I_1 = 89.8079\,\sigma_e^2$$

Similarly,

$$\sigma_{o2}^2 = \frac{\sigma_e^2}{2\pi j}\oint_C H(z)H(z^{-1})z^{-1}\mathrm{d}z$$

$$= \frac{\sigma_e^2}{2\pi j}\oint_C \frac{1}{(1-0.8z^{-1})} \times \frac{1}{(1-0.8z)} \times z^{-1} = \frac{\sigma_e^2}{2\pi j}\oint_C \frac{\mathrm{d}z}{(z-0.8)(1-0.8z)}$$

$$\sigma_{o2}^2 = \sigma_e^2 \times I_2$$

where

$$I_2 = \frac{1}{2\pi j}\oint_C \frac{\mathrm{d}z}{(z-0.8)(1-0.8z)}$$

The integral I_2 can be solved by the method of residues.

I_2 = sum of the residues at poles within the unit circle.

= residue at z = 0.8

$$= (z-0.8)\frac{1}{(z-0.8)(1-0.8z)}\Bigg|_{z=0.8} = 2.778$$

Therefore,

$$\sigma_{o2}^2 = \sigma_e^2 \times I_2 = 2.778\ \sigma_e^2$$

The overall output noise power is,

$$\sigma_{err}^2 = \sigma_{o1}^2 + \sigma_{o2}^2$$

$$= \sigma_e^2(89.8079 + 2.778) = 92.586\ \sigma_e^2$$

Substituting $\sigma_e^2 = \dfrac{2^{-2B}}{12}$, we get

$$\sigma_{err}^2 = 92.586 \times \frac{2^{-2B}}{12} = 7.715 \times 2^{-2B}$$

Example 10.5 can be solved using Eq. (10.51) as shown in Example 10.6.

Example 10.6 *Repeat Example* 10.5 *using Eq.* (10.51).

Solution Given

$$H_1(z) = \frac{1}{1-0.9z^{-1}} \text{ and } H_2(z) = \frac{1}{1-0.8z^{-1}}$$

From Eq. (10.51),

$$\sigma_{err}^2 = \sigma_{o1}^2 + \sigma_{o2}^2$$

where

$$\sigma_{o1}^2 = \sigma_e^2 \sum_{n=0}^{\infty} h_1^2(n) \text{ and } \sigma_{o2}^2 = \sigma_e^2 \sum_{n=0}^{\infty} h_2^2(n)$$

Let us first determine $h_1(n)$ and $h_2(n)$.

$$H(z) = \frac{1}{\left(1-0.9z^{-1}\right)\left(1-0.8z^{-1}\right)} = \frac{z^2}{(z-0.9)(z-0.8)}$$

Therefore,

$$\frac{H(z)}{z} = \frac{z}{(1-0.9)(1-0.8)} = \frac{9}{(z-0.9)} - \frac{8}{(z-0.8)}$$

and

$$H(z) == \frac{9}{\left(1-0.9z^{-1}\right)} - \frac{8}{\left(1-0.8z^{-1}\right)}$$

Taking inverse z-transform, we get

$$h_1(n) = [9(0.9)^n - 8(0.8)^n]u(n)$$

and

$$h_1^2(n) = [9(0.9)^n - 8(0.8)^n]^2 u(n)$$

Similarly,

$$h_2(n) = Z^{-1}[H_2(z)] = Z^{-1}\left[\frac{1}{1-0.8z^{-1}}\right] = (0.8)^n u(n)$$

and

$$h_2^2(n) = (0.8)^{2n} u(n)$$

Therefore,

$$\sigma_{o1}^2 = \sigma_e^2 \sum_{n=0}^{\infty} h_1^2(n) = \sigma_e^2 \sum_{n=0}^{\infty} \left[9(0.9)^n - 8(0.8)^n\right]^2$$

$$= \sigma_e^2 \sum_{n=0}^{\infty} (81(0.81)^n + 64(0.64)^n - 144(0.72)^n)$$

$$= \sigma_e^2 \left[\frac{81}{1-0.81} + \frac{64}{1-0.64} - \frac{144}{1-0.72}\right] = 89.80\sigma_e^2$$

and

$$\sigma_{o2}^2 = \sigma_e^2 \sum_{n=0}^{\infty} h_2^2(n) = \sigma_e^2 \sum_{n=0}^{\infty} (0.8)^{2n}$$

$$= \sigma_e^2 \sum_{n=0}^{\infty} (0.64)^n = \sigma_e^2 \left(\frac{1}{1-0.64}\right) = 2.778\sigma_e^2$$

The overall output noise power is,

$$\sigma_{err}^2 = \sigma_{o1}^2 + \sigma_{o2}^2$$

$$= \sigma_e^2(89.80 + 2.778) = 92.578\sigma_e^2$$

Substituting $\sigma_e^2 = \dfrac{2^{-2B}}{12}$, we get

$$\sigma_{err}^2 = 92.578 \times \frac{2^{-2B}}{12} = 7.715 \times 2^{-2B}$$

It can be noted that the result is same as that in Example 10.5.

SCALING 10.9

It has been discussed earlier that whenever the number of bits in the digital signal exceeds the capacity of the registers, overflow occurs. This can happen when two binary numbers are added or multiplied. Overflow can also occur in the *A*/*D* conversion process when the input signal's amplitude exceeds the maximum range of the *ADC*. The overflow of signals results in a number of errors, namely, overflow limit cycles, product quantisation noise or saturation noise. Once these errors are detected they can be easily handled, however, the most efficient approach is to prevent these errors from occurring. This can be done by ***scaling*** the input at certain points in the digital filter. Scaling must be done in such a way that no overflow occurs at the summing point (accumulator) of the digital filter.

Let us consider an *n*th order system, where scaling is done to the input and the unit impulse response between the input and the *i*th summing node. Let $y_i(n)$ be the output of the system at the *i*th summing node. $x(n)$ be the input sequence and $h_i(n)$ be the unit impulse response between the *i*th node and the input. The output $y_i(n)$ is given by

$$y_i(n) = \sum_{k=-\infty}^{\infty} h_i(k)x(n-k) \tag{10.55}$$

The magnitude of the output is

$$|y_i(n)| = \left|\sum_{k=-\infty}^{\infty} h_i(k)x(n-k)\right| \tag{10.56}$$

Using Schwarz's inequality,

$$|y_i(n)| \le \sum_{k=-\infty}^{\infty} |h_i(k)||x(n-k)| \tag{10.57}$$

If the maximum value of *x* is *X*, then

$$|y_i(n)| \le X \sum_{k=-\infty}^{\infty} |h_i(k)| \quad \text{for all } n$$

If the maximum range of data that can be handled by the digital processing system is [−1, 1], the output $y_i(n)$ must be lesser than 1.

Therefore,

$$X \sum_{k=-\infty}^{\infty} |h_i(k)| < 1$$

or

$$X < \frac{1}{\sum_{k=0}^{\infty} |h_i(k)|} \tag{10.58}$$

Equation (10.58) gives the necessary and sufficient condition for preventing overflow in a digital system. For an FIR filter, the above condition is slightly modified as shown below.

$$X < \frac{1}{\sum_{k=0}^{M-1} |h_i(k)|} \tag{10.59}$$

For an FIR filter, the unit impulse response is summed over only M non-zero terms.

ERRORS IN THE COMPUTATION OF DFT QUANTISATION 10.10

A major role is played by the discrete Fourier transforms in digital signal processing and hence the study of the effect of quantisation errors in the computation of DFT becomes important. The DFT of a finite duration sequence $\{x(n)\}$, $0 \le n \le M-1$ is given by

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

$$= \sum_{n=0}^{M-1} x(n) W_M^{kn} \qquad k = 0, 1, \ldots, M-1 \tag{10.60}$$

If $\{x(n)\}$ is assumed to be a complex-valued sequence, then the product $x(n)W_M^{kn}$ involves four real multiplications and each multiplication is rounded from $2B$ bits to B bits. Therefore, four quantisation errors are introduced for each complex-valued multiplication. In the direct computation of M-point DFT, M^2 complex-valued multiplications are involved. In other words. M complex-valued multiplications and $4M$ real-valued multiplications are performed for each point in the DFT. As a result, $4M$ quantisation errors are introduced for each point in the DFT. These $4M$ quantisation errors are assumed to be uncorrelated neither mutually nor with the sequence $\{x(n)\}$ and are uniformly distributed in the range $\left[-\frac{2^{-B}}{2}, \frac{2^{-B}}{2}\right]$.

The quantisation noise power (variance) due to rounding is given by

$$\sigma_e^2 = \frac{2^{-2B}}{12}$$

and the quantisation noise power from the $4M$ multiplication is

$$\sigma_{eq}^2 = 4M\sigma_e^2 = \frac{2^{-2B}}{3} M \tag{10.61}$$

It is seen from Eq. (10.61) that the quantisation noise power is directly proportional to the size of the DFT. If the size of DFT is a power of 2, i.e. $M = 2^u$, the quantisation noise power can be written as

$$\sigma_{eq}^2 = \frac{2^{-2B}}{3} M = \frac{2^{-2B}}{3} \cdot 2^u$$

$$= \frac{2^{-2(B-u/2)}}{3} \tag{10.62}$$

Equation (10.62) implies that every fourfold increase in the size of the discrete Fourier transform requires an addition a bit in computational precision to offset the additional quantisation errors.

Quantisation Errors in FFT Algorithms

FFT algorithms are better than DFTs because lesser number of multiplications are required in FFT algorithms. In spite of fewer multiplications involved in the computation of DFT through FFT algorithms, the quantisation error is not minimised in FFT. In the computation of FFTs, each butterfly computation involves one complex-valued multiplication or, equivalently, four real multiplications. The quantisation errors introduced in each butterfly propagate to the output.

The quantisation errors introduced in the first stage propagate through $(u - 1)$ stages, those introduced in the second stage propagate through $(u - 2)$ stages, and so on. As these quantisation errors propagate through a number of subsequent stages, they are phase shifted by the phase factor W_M^{kn}. These phase rotations do not change the statistical properties of the qunatisation errors and in particular, the variance of each quantisation error remains invariant. It is assumed that the quantisation errors in each butterfly are uncorrelated with the errors in other butterflies, then there are $4(M - 1)$ errors that affect the output of each point of the FFT. Consequently, the variance of the total quantisation error at the output is

$$\sigma_{eq}^2 = 4(M-1)\sigma_e^2 \approx \frac{2^{-2B}}{3}M \tag{10.63}$$

This is exactly the same result as the one obtained for direct computation of the DFT. This is because, the FFT algorithm does not really reduce the number of multiplications required to compute a single point of the DFT. It only exploits the periodicities in W_M^{kn} and thus reduces the number of multiplications in the computation of the entire block of M points in the DFT.

REVIEW QUESTIONS

10.1 What are the effects of finite word length in digital filters?

10.2 Analyse truncation and round-off processes in binary number representations.

10.3 List the errors which arise due to the quantisation process.

10.4 Give the quantisation error for positive and negative number representation.

10.5 What is sampling? Explain with an example.

10.6 What is the need for quantisation?

10.7 Obtain an expression for the variance of the round-off quantisation noise.

10.8 Obtain an expression for the variance of the output noise of a digital system which is fed with a quantised input signal.

10.9 The output of an A/D converter is fed through a digital filter whose system function is given by

$$H(z) = \frac{(1-\beta)z}{z-\beta}, \quad 0 < \beta < 1$$

Find the output noise power from the digital system.

Ans: $\sigma_{e0}^2 = \sigma_e^2\left[\frac{(1-\beta)^2}{1-\beta^2}\right]$

10.10 Find the output noise power of the digital filter whose system function is

$$H(z) = \frac{1}{1-0.999z^{-1}}$$

Ans: $\sigma_{eo}^2 = 500.25\ \sigma_e^2$

10.11 Explain coefficient quantisation.

10.12 Discuss coefficient quantisation in IIR filters.

10.13 Discuss coefficient quantisation in FIR filters.

10.14 What is meant by limit cycles in recursive structures?

10.15 What is dead band of a filter?

10.16 A digital system is characterised by the difference equation

$y(n) = 0.95\, y(n-1) + x(n)$

Determine the deadband of the system when $x(n) = 0$ and $y(-1) = 13$.

Ans: $[-10, 10]$

10.17 Explain product quantisation.

10.18 Two first order low-pass filters whose system functions are given below are connected in cascade. Determine the overall output noise power.

$$H_1(z) = \frac{1}{1 - b_1 z^{-1}} \quad \text{and} \quad H_2(z) = \frac{1}{1 - b_2 z^{-1}}$$

Ans: $\sigma_{err}^2 = \sigma_{o1}^2 + \sigma_{o2}^2;\ \sigma_{o1}^2 = \sigma_e^2 \frac{1}{1 - b_1^2};$

$$\sigma_{o2}^2 = \sigma_e^2 \left[\frac{1 + b_1 b_2}{(1 - b_1 b_2)(1 - b_1^2)(1 - b_2^2)} \right]$$

10.19 Determine the variance of the round-off noise at the output of the two cascade realisation of the filter with system function

$$H(z) = H_1(z) \cdot H_2(z)$$

where

$$H_1(z) = \frac{1}{1 - 0.5 z^{-1}} \quad \text{and} \quad H_2(z) = \frac{1}{1 - 0.25 z^{-1}}$$

Ans: $\sigma_{err}^2 = \sigma_{o1}^2 + \sigma_{o2}^2;\ \sigma_{o1}^2 = 2.90\, \sigma_e^2;\ \sigma_{o2}^2 = 3.16\, \sigma_e^2$

Chapter 11 Multirate Digital Signal Processing

INTRODUCTION 11.1

The systems that use single sampling frequency for analog-to-digital conversion are called single-rate systems. But, in many practical applications such as software assigned radios, teletype, facsimile, digital video and audio, different sampling rates are used to process the various signals corresponding to their bandwidths. Hence, multirate digital signal processing is required in digital systems where more than one sampling rate is needed. In digital audio, the different sampling rates used are 32kHz for broadcasting, 44.1 kHz for compact disc and 48 kHz for audio tape. In digital video, the sampling rates for composite video signals are 14.3181818 MHz and 17.134475 MHz for NTSC and PAL respectively. But the sampling rates for digital component of video signals are 13.5 MHz and 6.75 MHz for luminance and colour difference signal.

Different sampling rates can be achieved using an upsampler and downsampler. The basic operations in multirate processing to achieve this are decimation and interpolation. *Decimation* is for reducing the sampling rate and *interpolation* is for increasing the sampling rate. The sampling frequency is increased or decreased without any undesirable effects of errors due to quantisation and aliasing. While designing multirate systems, the effects of aliasing for decimation and pseudo-images for interpolation should be avoided.

In a digital transmission system like teletype, facsimile, low bit-rate speech, where data has to be handled in different rates, multirate signal processing is used. Multirate signal processing finds its application in (i) sub-band coding of speech or image (ii) voice privacy using analog phone lines, (iii) signal compression by sub-sampling, and (iv) A/D and D/A converters. Also, it is used in the areas such as communication systems, data acquisition and storage systems, speech and audio processing systems, antenna systems and radar systems.

The advantages of multirate signal processing are that it reduces computational requirement, storage for filter coefficients, finite arithmetic effects, filter order required in multirate application and sensitivity to filter coefficient length.

SAMPLING 11.2

Let $x(t)$ be a continuous time-varying signal. The signal $x(t)$ is sampled at regular intervals of time with a sampling period T as shown in Fig. 11.1. The sampled signal $x(nT)$ is given by

$$x(nT) = x(t)\big|_{t = nT}, \qquad -\infty < n < \infty \tag{11.1}$$

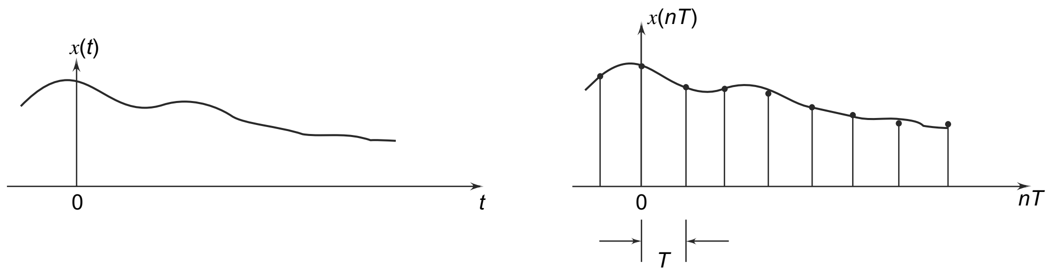


Fig. 11.1 *Sampling of a continuous time signal*

A sampling process can also be interpreted as a modulation or multiplication process, as shown in Fig. 11.2.

The continuous time signal $x(t)$ is multiplied by the sampling function $s(t)$ which is a series of impulses (periodic impulse train); the resultant signal is a discrete time signal $x(nT)$.

$$x(nT) = x(t) \sum_{n=-\infty} \delta(t - nT) \tag{11.2}$$

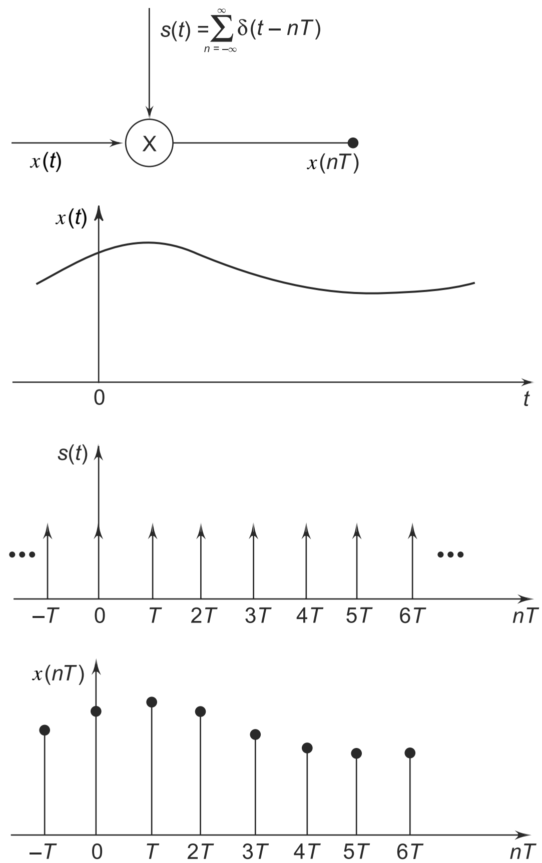


Fig. 11.2 *Periodic sampling of x(t)*

Sampling Theorem

The sampling theorem states that a band-limited signal $x(t)$ having finite energy which has no frequency components higher than f_h Hz can be completely reconstructed from its samples taken at the rate of $2f_h$ samples per second, i.e., $f_s \geq 2f_h$, where f_s is the sampling frequency and f_h is the highest signal frequency.

The sampling rate of $2f_h$ samples per second is the Nyquist rate and its reciprocal $1/(2f_h)$ is the Nyquist period. For simplicity, $x(nT)$ is denoted as $x(n)$.

SAMPLING RATE CONVERSION 11.3

Sampling rate conversion is the process of converting the sequence $x(n)$ which is obtain from sampling the continuous time signal $x(t)$ with a period T to another sequence $y(k)$ obtained from sampling $x(t)$ with a period T'.

The new sequence $y(k)$ can be obtained by first reconstructing the original signal $x(t)$ from the sequence $x(n)$ and then sampling the reconstructed signal with a period T'. Figure 11.3 shows the reconstruction of the original signal with a *D/A* converter, low-pass filter and resampler with a sampling period T'.

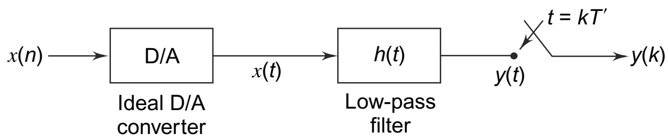


Fig. 11.3 *Conversion of a Sequence x(n) to Another Sequence y(k)*

11.3.1 Decimation by a Factor *M*

The process of reducing the sampling rate of a signal without resulting in aliasing is called *decimation* or *sampling rate reduction*. If M is the integer sampling rate reduction factor for the signal $x(n)$, then

$$\frac{T'}{T} = M$$

The new sampling rate F' becomes

$$F' = \frac{1}{T'} = \frac{1}{MT} = \frac{F}{M} \tag{11.3}$$

Let the signal $x(n)$ be a full band signal, with non-zero values in the frequency range $-F/2 \leq f \leq F/2$, where $\omega = 2\pi fT$.

$$|X(e^{j\omega})| \neq 0, \quad |\omega| \leq |2\pi fT| \leq 2\pi fT/2 = \pi \tag{11.4}$$

To avoid aliasing caused by downsampling, the high frequency components in signal $x(n)$ must be removed by using a low-pass filter which has the following spectral response:

$$H(e^{j\omega}) = \begin{cases} 1, & |\omega| \leq 2\pi F'T/2 = \pi / M \\ 0, & \text{otherwise} \end{cases} \tag{11.5}$$

The sequence $y(k)$ is obtained by selecting only the M^{th} sample of the filtered output which results in sampling rate reduction. If the impulse response of the filter is $h(n)$, then the filtered output $w(n)$ is given by

$$w(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) \tag{11.6}$$

The decimated signal $y(m)$ is

$$y(m) = w(M\, m) \tag{11.7}$$

The sampling rate reduction is represented by a down arrow along with the decimation factor M and the entire process is represented in Fig. 11.4.

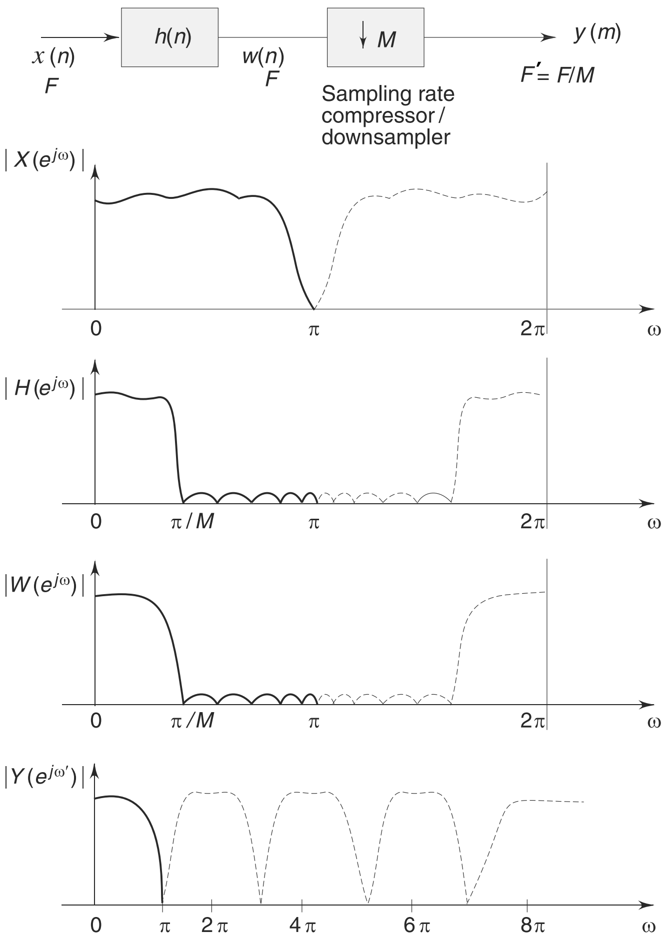


Fig. 11.4 *Decimation of x(n) by a factor M*

Combining Eqs. (11.6) and (11.7), $y(m)$ becomes

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

or
$$y(m) = \sum_{n=-\infty}^{\infty} h(Mm-n)x(n) \tag{11.8}$$

The decimator is also known as subsampler, downsampler or undersampler. In the practical case, where a non-ideal low-pass filter is used, the output signal $y(m)$ will not be a perfect one. Consider the signal $w'(n)$ defined by

$$w'(n) = \begin{cases} w(n), & n = 0, \pm M, \pm 2M, \ldots \\ 0, & \text{otherwise} \end{cases} \tag{11.9}$$

At the sampling instants of $y(m)$, $w'(n) = w(n)$ and in other cases, it is zero. The general representation can be

$$w'(n) = w(n)\left\{\frac{1}{M}\sum_{l=0}^{M-1} e^{j2\pi ln/M}\right\}, \quad -\infty < n < \infty \tag{11.10}$$

The term multiplied to $w(n)$ is the complex Fourier series representation of a periodic impulse train, shown in Fig.11.5, given by

$$s_M(l) = \frac{1}{M}\sum_{n=-\infty}^{\infty} e^{-j\frac{2\pi nl}{M}}, -\infty < l < \infty$$

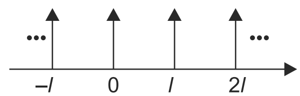


Fig. 11.5 *Output Signal*

Now, the output signal becomes

$$y(m) = w'(Mm) = w(Mm)$$

Spectrum of Downsampled Signal

The z-transform of the output signal is given by

$$Y(z) = \sum_{m=-\infty}^{\infty} y(m)z^{-m} = \sum_{m=-\infty}^{\infty} w'(Mm)z^{-m}$$

Since $w'(n)$ takes values only for $m = 0, \pm M, \pm 2M, \ldots$, $Y(z)$ becomes

$$Y(z) = \sum_{m=-\infty}^{\infty} w'(m)z^{-m/M}$$

Substituting the value for $w'(m)$, $Y(z)$ becomes

$$Y(z) = \sum_{m=-\infty}^{\infty} w(m)\left\{\left(\frac{1}{M}\right)\sum_{l=0}^{M-1} e^{j2\pi lm/M}\right\}z^{-m/M}$$

But $W(z) = X(z)\,H(z)$. Therefore,

$$Y(z) = \left(\frac{1}{M}\right)\sum_{l=0}^{M-1} H\left(e^{-j2\pi l/M}z^{1/M}\right)X\left(e^{-j2\pi l/M}z^{1/M}\right) \tag{11.11}$$

When evaluated on a unit circle, $z = e^{j\omega}$

$$Y\left(e^{j\omega}\right) = \left(\frac{1}{M}\right)\sum_{l=0}^{M-1} H\left(e^{-j(\omega-2\pi l)/M}\right)X\left(e^{-j(\omega-2\pi l/M)}\right) \tag{11.12}$$

The low-pass filter $H(z)$ serves as an anti-aliasing filter. It has to filter the spectral components of $x(n)$ above $\omega = \pi/M$, i.e. all terms corresponding to $l \neq 0$ are removed and the filter closely resembles the ideal case, then,

$$Y\left(e^{j\omega}\right) = \left(\frac{1}{M}\right)X\left(e^{j\omega/M}\right), \text{ for } |\omega| \leq \pi \tag{11.13}$$

Consider down sampling of the input signal $x(n)$ by a factor of two. Let $X(e^{j\omega})$ be a real function with an asymmetric frequency response.

$$Y\left(e^{j\omega}\right) = \frac{1}{2}\left(X\left(e^{j\omega/2}\right) + X\left(-e^{j\omega/2}\right)\right)$$

The spectrum for $X(e^{j\omega})$ and $Y(e^{j\omega})$ for a downsampler scaled by a factor $\frac{1}{M}$ is shown in Fig. 11.6. In the interval 0 to 2π, there will be $(M-1)$ equally spaced replica of $X(e^{j\omega})$ with period $\frac{2\pi}{M}$.

$$X\left(-e^{j\omega/2}\right) = X\left[-e^{j\frac{(\omega-2\pi)}{2}}\right]$$

The original shape of $X(e^{j\omega})$ is lost when $x(n)$ is downsampled. This results in aliasing which is because of undersampling or downsampling. There will be no aliasing if

$$X\left(e^{j\omega}\right) = 0, \text{ for } |\omega| \geq \frac{\pi}{2}$$

For a downsampler and decimator with factor M, $Y(e^{j\omega})$ now becomes

$$Y\left(e^{j\omega}\right) = \frac{1}{M}\sum_{k=0}^{M-1} X\left[e^{\frac{j(\omega-2\pi k)}{M}}\right]$$

$Y(e^{j\omega})$ can be expressed as a sum of M uniformly shifted and stretched versions of $X(e^{j\omega})$ and then scaled by a factor $\frac{1}{M}$. There will be no aliasing if the signal $x(n)$ is band limited to $\pm\frac{\pi}{M}$.

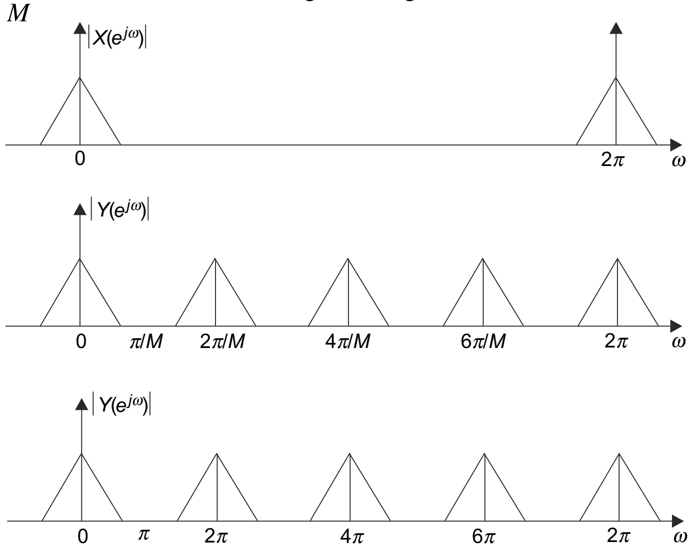


Fig. 11.6 *Magnitude of $X(e^{j\omega})$ and $Y(e^{j\omega})$*

Now consider the signal $x(n)$ whose frequency is greater than $\pm\frac{\pi}{M}$. The frequency spectra of $\left|X\left(e^{j\omega}\right)\right|$ and $\left|Y\left(e^{j\omega}\right)\right|$ are plotted as shown in Fig. 11.7(a) respectively. In Fig. 11.7(b), the aliasing

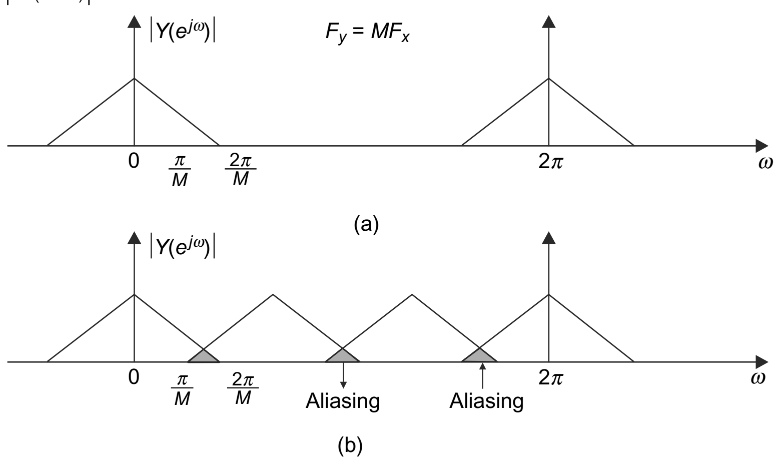


Fig. 11.7 *Spectrum with Aliasing Effect in Decimator*

effect is shown. Because of aliasing signal, distortion in $y(m)$ will take place and hence a low-pass filter is to be connected before the signal $x(n)$ is passed through the downsampler. This will limit the input signal to the downsampler to $\pm\frac{\pi}{M}$ which avoids aliasing and hence, signal distortion.

Example 11.1 *Obtain the decimated signal $y(m)$ by a factor three from the input signal $x(n)$ shown in Fig. E11.1(a).*

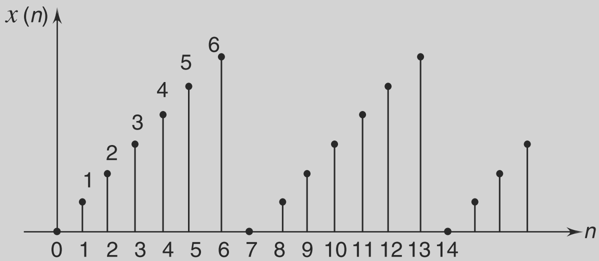


Fig. E11.1(a)

Solution The decimated signal is given by

$$y(m) = x(Mn), \text{ where } M = 3$$

The input and output signals are shown in Fig. E11.1(a) and (b) respectively.

Downsampling is a process that drops samples from a given sequence. Downsampling the given sequence by a factor 3 literally means drop two samples 1 and 2, 4 and 5, 7 and 8, 10 and 11. The output signal is shown in Fig. E11.1(b).

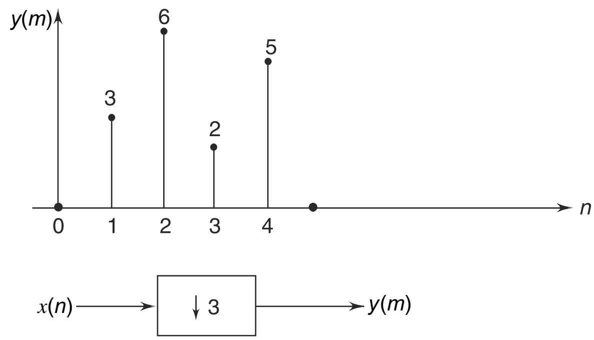


Fig. E11.1(b)

Example 11.2 *A one stage decimator shown in Fig. E11.2 is characterized by the following:*

Decimation factor = 3

Anti-aliasing filter coefficients

$$h(0) = -0.08 = h(4)$$
$$h(1) = 0.3 = h(3)$$
$$h(2) = 0.8$$

Given the data, $x(n)$ with successive values [5, −2, −3, 10, 6, 4, −2], *calculate and list the filtered output, $w(n)$, and the output of the decimator, $y(m)$.*

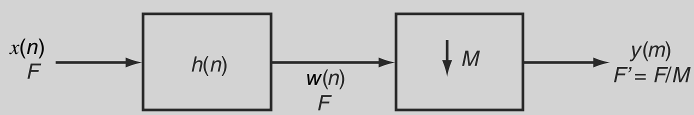


Fig. E11.2 *Decimation Process for the Factor M = 3*

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

Decimator output, $y(m)=\sum_{n=-\infty}^{\infty}h(Mn-n)x(n)$

Given $M = 3$ and $x(n) = \{5, -2, -3, 10, 6, 4, -2\}$

Filtered Output

$$w(0) = h(0)x(0) = (-0.08)(5) = -0.4$$
$$w(1) = h(0)x(1) + h(1)x(0) = (-0.08)(-2) + (0.3)(5) = 1.66$$
$$w(2) = h(0)x(2) + h(1)x(1) + h(2)x(0)$$
$$= (-0.08)(-3) + (0.3)(-2) + (0.8)(5)= 3.64$$
$$w(3) = h(0)x(3) + h(1)x(2) + h(2)x(1) + h(3)x(0)$$
$$= (-0.08)(10) + (0.3)(-3) + (0.08)(-2) + (0.3)(5) = -1.8$$
$$w(4) = h(0)x(4) + h(1)x(3) + h(2)x(2) + h(3)x(1) + h(4)x(0)$$
$$= (-0.08)(6) + (0.3)(10) + (0.8)(-3) + (0.3)(-2) + (-0.08)(5) = -0.88$$
$$w(5) = h(0)x(5) + h(1)x(4) + h(2)x(3) + h(3)x(2) + h(4)x(1)$$
$$= (-0.08)(4) + (0.3)(6) + (0.8)(10) + (0.3)(-3) + (-0.08)(-2) = 8.74$$
$$w(6) = h(0)x(6) + h(1)x(5) + h(2)x(4) + h(3)x(3) + h(4)x(2)$$
$$= -(0.08)(-2) + (0.3)(4) + (0.8)(6) + (0.3)(10) + (-0.08)(-3) = 9.4$$
$$w(7) = h(1)x(6) + h(2)x(5) + h(3)x(4) + h(4)x(3)$$
$$= (0.3)(-2) + (0.8)(4) + (0.3)(6) + (-0.08)(10) = 3.6$$
$$w(8) = h(2)x(6) + h(3)x(5) + h(4)x(4)$$
$$= (0.8)(-2) + (0.3)(4) + (-0.08)(6)= -0.88$$
$$w(9) = h(3)x(6) + h(4)x(5) = (0.3)(-2) + (-0.08)(4) = -0.92$$
$$w(10) = h(4)x(6) = (-0.08)(-2) = 0.16$$

Filtered output, $w(n) = \{-0.4, 1.66, 3.64, -1.8, -0.88, 8.74, 9.4, 3.6, -0.88, -0.92, 0.16\}$

Decimator Output

$$y(0) = h(0)x(0) = (-0.08)(5) = -0.4$$
$$y(1) = h(3)x(0) + h(2)x(1) + h(1)x(2) + h(0)x(3)$$
$$= (0.3)(5) + (0.8)(-2) + (0.3)(-3) + (-0.08)(10) = -1.8$$
$$y(2) = h(4)x(2) + h(3)x(3) + h(2)x(4) + h(1)x(5) + h(0)x(6)$$
$$= (-0.08)(-3) + (0.3)(10) + (0.8)(6) + (0.3)(4) + (-0.08)(-2) = 9.4$$
$$y(3) = h(4)x(5) + h(3)x(6) = (-0.08)(4) + (0.3)(-2) = -0.92$$

Decimator output, $y(m) = \{-0.4, -1.8, 9.4, -0.92\}$

Alternate Solution (using z-transform)

Given $x(n) = \{5, -2, -3, 10, 6, 4, -2\}$

Therefore, $X(z) = 5 - 2z^{-1} - 3z^{-2} + 10z^{-3} + 6z^{-4} + 4z^{-5} - 2z^{-6}$

Given, $h(n) = \{-0.08, 0.3, 0.8, 0.3, -0.08\}$

Therefore, $H(z) = -0.08 + 0.3z^{-1} + 0.8z^{-2} + 0.3z^{-3} - 0.08z^{-4}$

We know that, $W(z) = X(z)\, H(z)$

Taking inverse z-transform, we get

$$w(n) = \{-0.4, 1.66, 3.64, -1.8, -0.88, 8.74, 9.4, 3.6, -0.88, -0.92, 0.16\}$$

Decimating by factor 3, we get

$$y(m) = \{-0.4, -1.8, 9.4, -0.92\}$$

Example 11.3 *Determine the computational complexity of a single-stage decimator designed to reduce the sampling rate from 60 kHz to 3 kHz. The decimation filter is to be designed as an equiripple FIR filter with a passband edge at 1.25 kHz, a passband ripple of 0.02, and a stopband ripple of 0.01. Use the total multiplications per second as a measure of the computational complexity.*

Solution

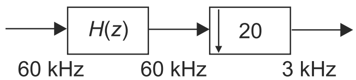


Fig. E11.3

Specifications for $H(z)$ are $F_p = 1250$ Hz, $F_s = 1500$ Hz, $\delta_p = 0.02$ and $\delta_s = 0.01$.
The length N of an equiripple linear phase FIR filter is given by

$$N = \frac{-20\log_{10}\sqrt{\delta_p \delta_s} - 13}{14.6\Delta f}$$

where $\Delta f = \dfrac{F_s - F_p}{F_T}$ is the normalised transition bandwidth.

Therefore, $N = \dfrac{-20\log_{10}\sqrt{0.02\times 0.01} - 13}{14.6(250/60000)} = \dfrac{23.989\times 60000}{14.6\times 250} = 394.34.$ We choose $N = 395$.

Hence, the computational complexity $= 395\times\dfrac{60{,}000}{20} = 11{,}85{,}000$.

11.3.2 Interpolation

The process of increasing the sampling rate of a signal is interpolation (sampling rate expansion). Letting L be an integer interpolating factor, of the signal $x(n)$, then

$$T'/T = 1/L$$

The sampling rate is given by

$$\begin{aligned} F' &= 1/T' \\ &= L/T \\ &= LF \end{aligned} \tag{11.14}$$

Interpolation of a signal $x(n)$ by a factor L refers to the process of interpolating $(L-1)$ samples between each pair of samples of $x(n)$.

The signal $w(m)$ is got by interpolating $(L-1)$ samples between each pair of the samples of $x(n)$.

$$w(m) = \begin{cases} x(m/L), & m = 0, \pm L, \pm 2L, \ldots \\ 0, & \text{otherwise} \end{cases} \tag{11.15}$$

The z-transform of the signal $w(m)$ is given by

$$W(z) = \sum_{m=-\infty}^{\infty} w(m) z^{-m}$$

$$= \sum_{m=-\infty}^{\infty} x(m) z^{-mL}$$

$$= X(z^L) \tag{11.16}$$

When considered over the unit circle, $z = e^{j\omega}$,

$$W(e^{j\omega}) = X(e^{j\omega L}) \tag{11.17}$$

where $\omega = 2\pi fT'$. The spectrum of the signal $w(m)$ contains the images of baseband placed at the harmonics of the sampling frequency $\pm 2\pi/L$, $\pm 4\pi/L$. To remove the images, an anti-imaging (low-pass) filter is used. The ideal characteristic of the low-pass filter is given by

$$H(e^{j\omega}) = \begin{cases} G, & |\omega| \geq 2\pi f\, T'/2 = \pi / L \\ 0, & \text{otherwise} \end{cases} \tag{11.18}$$

where G is the gain of the filter. The frequency response of the output signal is given by

$$Y(e^{j\omega}) = H(e^{j\omega})\, X(e^{j\omega L})$$

$$= \begin{cases} GX(e^{j\omega L}), & |\omega| \leq \pi / L \\ 0, & \text{otherwise} \end{cases} \tag{11.19}$$

The output signal $y(m)$ is given by

$$y(m) = \sum_{k-\infty}^{\infty} h(m-k)\, w(k)$$

$$= \sum_{k-\infty}^{\infty} h(m-k)\, x(k/L), \quad k/L \text{ an integer} \tag{11.20}$$

Figure 11.8 shows the sampling rate expansion which is represented by an up arrow along with the entire process of interpolation by a factor L. The interpolation process is also called upsampling.

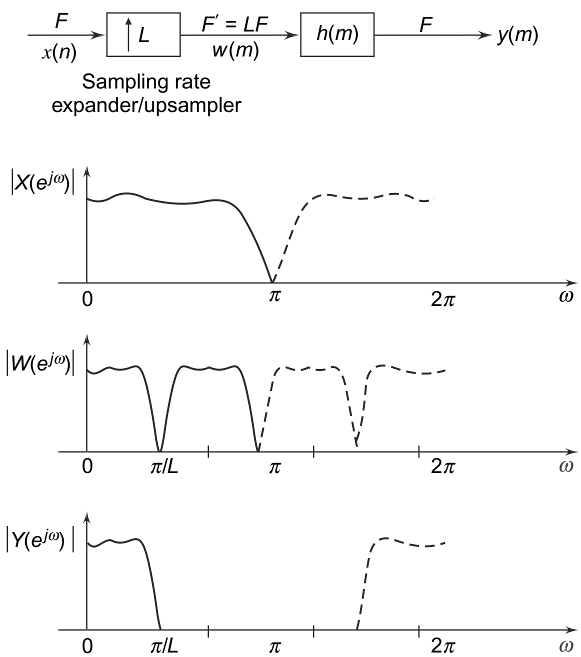


Fig. 11.8 *Interpolation of x(n) by a Factor L*

Spectrum of Upsampled Signal

The increase in sampling rate is achieved by an interpolator or an upsampler. The output of the interpolator is passed through the LPF to avoid removal of some of the desired frequency components of the sampled signal. Let $v(m)$ be the sequence obtained from the interpolator. $v(m)$ is obtained by adding $(L-1)$ zeros between successive values of $x(n)$. If F_s is the sampling rate of $x(n)$, then the sampling rate of $v(m)$ is LF_s which is same as the sampling rate of $y(m)$. The interpolator system is shown in Fig. 11.9(a). $v(m)$ is characterised by the following equation:

$$v(m)=\begin{cases} x\left(\dfrac{m}{L}\right), & m=0,\pm L,\pm 2L,\ldots \\ 0, & \text{otherwise} \end{cases} \tag{11.21}$$

The z-transform of the above equation is written as

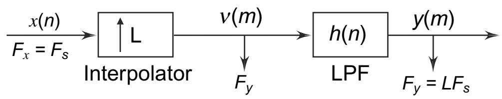


Fig. 11.9(a) *Interpolation with a Filter*

$$V(z)=\sum_{m=-\infty}^{\infty} v(m)z^{-m}$$

$$=\sum_{m=-\infty}^{\infty} x(m)z^{-mL}$$

$$=X(z^L) \tag{11.22}$$

The frequency spectrum of $v(m)$ is obtained by evaluating $V(z)$ on the unit circle $(z=e^{j\omega})$ in the z-plane. Thus,

$$V\left(e^{j\omega}\right)=X\left(e^{j\omega_y L}\right)$$

or $$V(\omega_y)=X(\omega_y L) \tag{11.23}$$

where $\omega_y = 2\pi F_y$ and $L\omega_y = \omega_x$.

The spectra of $|X(\omega_x)|$, $|V(\omega_y)|$, $|H(\omega_y)|$ and $|Y(\omega_y)|$ are shown in Fig. 11.9(b).

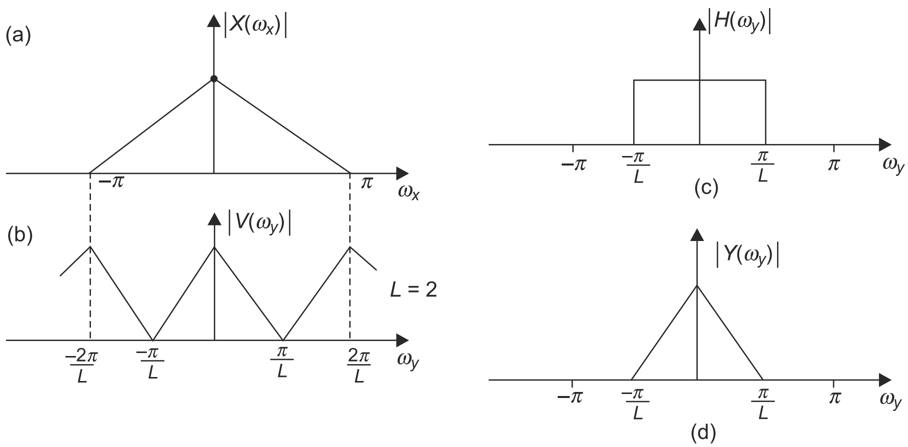


Fig. 11.9(b) *Frequency Spectra of x(n), v(n), h(n) and y(m)*

Example 11.4 *Obtain the two-fold expanded signal y(n) of the input signal x(n).*

$$x(n)=\begin{cases} n, & n>0 \\ 0, & otherwise \end{cases} \quad \textit{as shown in Fig. E11.4(a).}$$

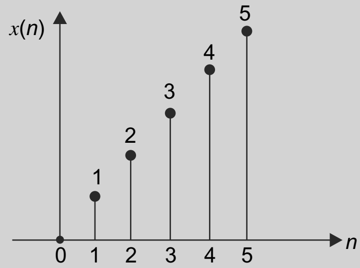


Fig. E11.4(a)

Solution The output signal $y(n)$ is given by

$$y(n)=\begin{cases} x(n/L), & n= \text{ multiples of } L \\ 0, & \text{otherwise} \end{cases}$$

where $L = 2$.

$$x(n) = 0, 1, 2, 3, 4, 5, \ldots$$
$$y(n) = 0, 0, 1, 0, 2, 0, 3, 0, 4, 0, 5, 0, \ldots$$

In general, to obtain the expanded signal $y(n)$ by a factor L, $(L-1)$ zeros are inserted between the samples of the original signal $x(n)$.

The z-transform of the expanded signal is

$$Y(z) = X(z^L), \quad L = 2.$$

The output signals is shown in Fig. E11.4(b).

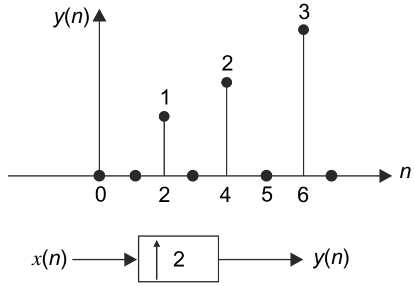


Fig. E11.4(b)

Example 11.5 *Determine the computational complexity of a single-stage interpolator to be designed to increase the sampling rate from 600 Hz to 9 kHz. The interpolator is to be designed as an equiripple FIR filter with a passband edge at 200 Hz, a passband ripple of 0.002, and a stopband ripple of 0.004. (a) Estimate the order of the FIR filter. (b) Develop a two-stage design of the above interpolator by the factors 3 and 5 and compare its computational complexity with that of the single-stage design.*

Solution

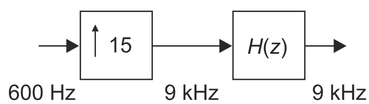


Fig. E11.5(a)

(a) Specifications for $H(z)$ are F_p = 200 Hz, F_s = 300 Hz, $\delta_p = 0.002$ and $\delta_s = 0.004$.

Here, $\Delta f = \dfrac{100}{9000}$.

We know that length of the equiripple linear-phase FIR filter is

$$N=\frac{-20\log_{10}\sqrt{\delta_p\delta_s}-13}{14.6\Delta f}$$

where δ_p is the passband ripple, δ_s is the stopband ripple and $\Delta f = \dfrac{F_s - F_p}{F_T}$ is the normalised transition bandwidth. Here, F_p is the passband edge and F_s is the stopband edge.

$$N=\frac{-20\log_{10}\sqrt{0.002\times 0.004}-13}{14.6(100/9000)}=\frac{37.969\times 9000}{14.6\times 100}=234.06. \text{ We choose } N = 235.$$

Hence, computational complexity of $H(z) = (235+1) \times \frac{9000}{15} = 141{,}600$ mps

(b) We realise $H(z) = C(z^5)\,A(z)$

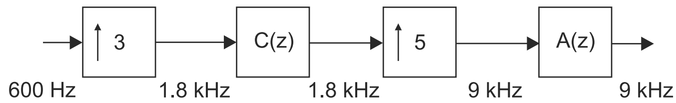


Fig. E11.5(b)

Specifications for $C(z)$are: F_p = 5 × 200 Hz = 1000 Hz, F_s = 5 × 300 Hz = 1500 Hz, δ_p = 0.002 and δ_s = 0.004. Here, $\Delta f = \frac{500}{9000}$.

Hence, the order N_C of $C(z)$ is given by

$$N_C = \frac{-20\log_{10}\sqrt{0.002 \times 0.004)} - 13}{14.6\Delta f} = 46.81. \text{ We choose } N_C = 47.$$

$$R_{M,C} = 47 \times \frac{1800}{3} = 28{,}200 \text{ mps}$$

Specifications for $A(z)$ are: F_p = 200 Hz = 1000 Hz, F_s = 1500 Hz, δ_p = 0.002 and δ_s = 0.004.

Here, $\Delta f = \frac{1300}{9000}$.

Hence, the order N_A of $A(z)$ is given by

$$N_A = \frac{-20\log_{10}\sqrt{0.002 \times 0.004)} - 13}{14.6(1300/9000)} = 18.004. \text{ We choose } N_A = 19.$$

$$R_{M,A} = 19 \times \frac{9000}{5} = 34{,}200 \text{ mps.}$$

Total computational complexity of the IFIR-based realisation is, therefore,

$R_{M,C} + R_{M,A} = 28{,}200 + 34{,}200 = 62{,}400$ mps.

11.3.3 Sampling Rate Conversion by a Rational Factor $\frac{M}{L}$

Consider the sampling rate conversion by a factor $\frac{M}{L}$, i.e., $\frac{T'}{T} = \frac{M}{L}$.

The sampling rate F' is $F' = \frac{L}{M}F$

The fractional conversion can be obtained by first increasing the sampling rate by L and then decreasing it by M. The interpolation process should be done before the decimation process to avoid any loss of information in the signal. Figures 11.10(a) and (b) show the process of fractional sampling rate conversion.

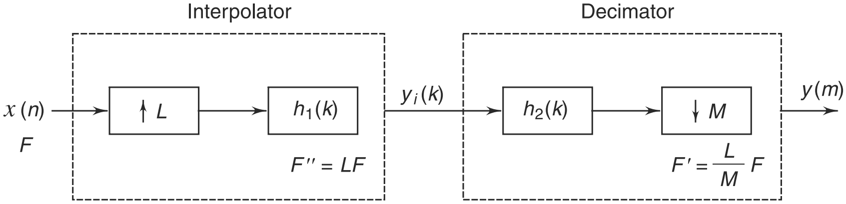


Fig. 11.10(a) *Cascade Connection of Interpolator and Decimator for Fractional Sampling Rate Conversion*

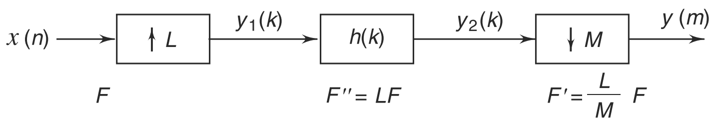


Fig. 11.10(b) *Efficient Way of Fractional Sampling Rate Conversion*

The filters $h_1(k)$ and $h_2(k)$ can be combined into one composite low-pass filter $h(k)$, operating at the sampling rate LF. The frequency respones of the filter $h(k)$ should be

$$H\left(e^{j\omega''}\right) = \begin{cases} L, & |\omega''| \le \min\left(\dfrac{\pi}{L}, \dfrac{\pi}{M}\right) \\ 0, & \text{otherwise} \end{cases}$$

where $$\omega'' = 2\pi f T'' = 2\pi f \frac{T}{L}$$

Since $$T'' = \frac{T}{L}$$

The sampling rate of the filter is $F'' = LF$.

The time-domain and frequency-domain input-output, output-input relationships are obtained as given below.

Time-domain Relationships

From Fig. 11.8(b),

$$y_2(k) = \sum_{r=-\infty}^{\infty} h(k - rL)\, x(r)$$

$$y(M) = y_2(Mm)$$

Expressing $y(m)$ in terms of $x(r)$,

$$y(m) = \sum_{r=-\infty}^{\infty} h(Mm - rL)\, x(r)$$

Frequency-domain Relationships

The output spectrum is given by

$$Y\left(e^{j\omega'}\right) = \frac{1}{M} \sum_{l=0}^{M-1} Y_2\left(e^{j(\omega' - 2\pi l)/m}\right)$$

where $$Y_2\left(e^{j\omega''}\right) = H\left(e^{j\omega''}\right) X\left(e^{j\omega'' L}\right)$$

$$Y_2\left(e^{j\omega''}\right) = \frac{1}{M} \sum_{l=0}^{M-1} H\left(e^{j(\omega' - 2\pi l/m}\right) X\left(e^{j(\omega' L - 2\pi l/m}\right)$$

$$Y\left(e^{j\omega'}\right) = \begin{cases} \dfrac{L}{M} X\left(e^{j\omega' L/M}\right), & \text{for } |\omega'| \le \min\left(\pi, \dfrac{\pi M}{L}\right) \\ 0, & \text{otherwise} \end{cases}$$

SIGNAL-FLOW GRAPHS 11.4

Any multirate digital system can be easily represented in terms of signal-flow graphs. A signal-flow graph comprises branches and nodes, where branches represent the signal operations and nodes represent the connection points.

Figure 11.11 shows the various branch operations in the signal-flow graphs. When signals enter a branch, they are associated with the input node of the branch. External signals enter the input branches and signals at the output branches are terminal signals. The sum of the signals entering the node is equal to the sum of the signals leaving the node. Based on the signal flow graph of Fig. 11.12, the network equations can be written as follows:

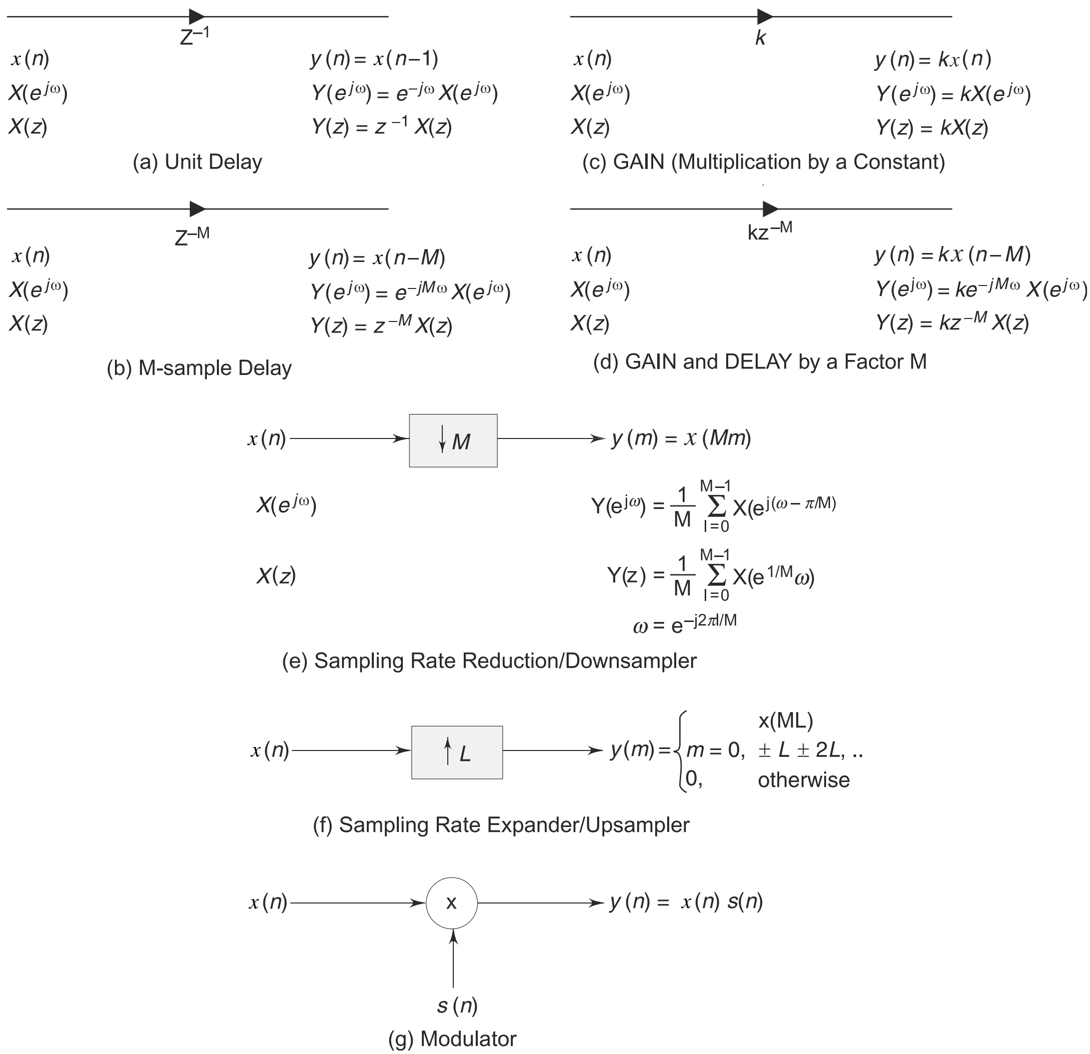


Fig. 11.11 *Branch Operations in Signal-Flow Graphs*

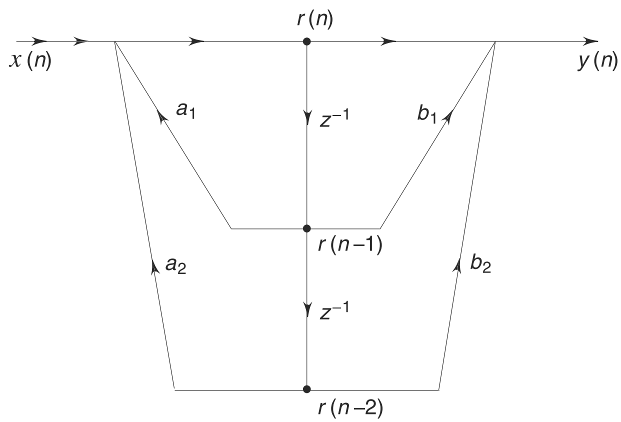


Fig. 11.12 *Signal-flow Graph for a Second–order System*

At the input node,

$$r(n) = x(n) + a_1 r(n-1) + a_2 r(n-2) \tag{11.24}$$

At the output node,

$$y(n) = r(n) + b_1 r(n-1) + b_2 r(n-2) \tag{11.25}$$

Combining both the equations,

$$y(n) = x(n) + b_1 x(n-1) + b_2 x(n-2) + a_1 y(n-1) + a_2 y(n-2) \tag{11.26}$$

11.4.1 Manipulation of Signal-Flow Graphs

Manipulation of signal-flow graphs which is shown in Fig. 11.13, corresponds to the ways how the set of network equations are represented. In multirate systems, it is easier to modify the signal-flow graphs than to modify the set of network equations.

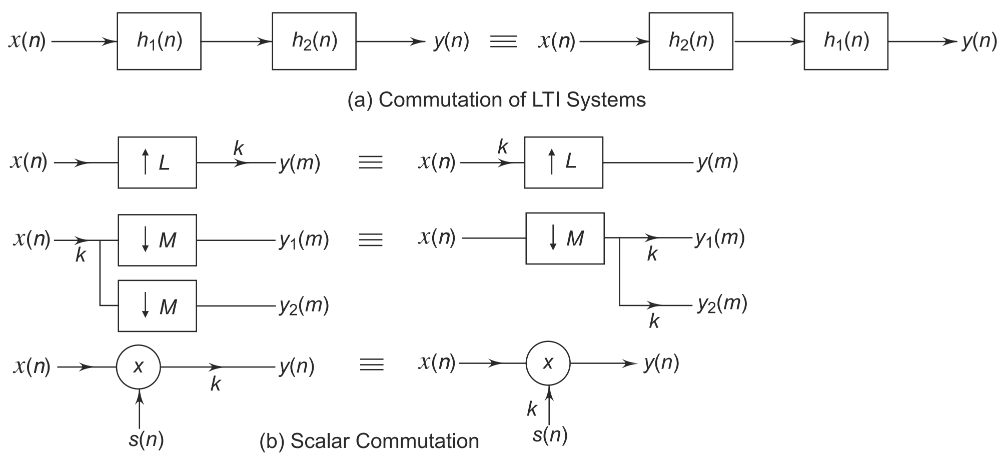
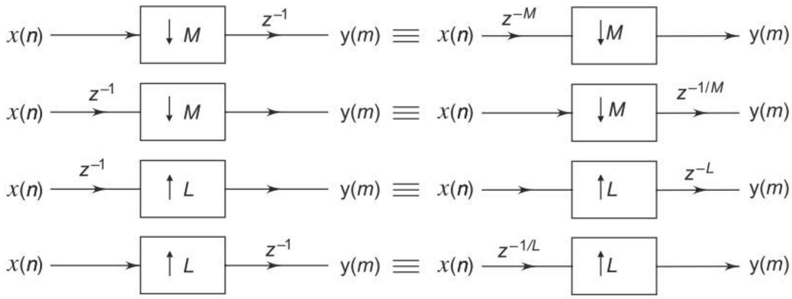
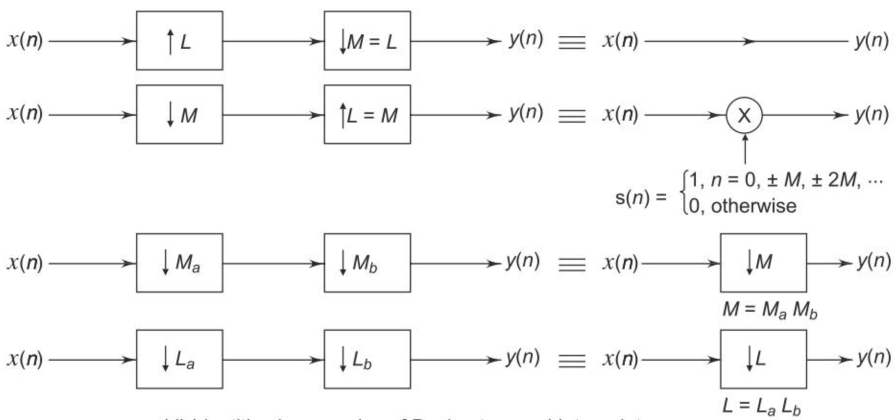


Fig. 11.13 *(Contd.)*



(c) Identities in Decimators and Interpolators



(d) Identities in cascades of Decimators and Interpolators

Fig. 11.13 *Manipulation of Signal-Flow Graphs*

The principle of commutation of branch operations is an important concept in the manipulation of signal-flow graphs. In a cascaded system, two branch operations commute in the order of the cascaded operations being changed.

Example 11.6 *Obtain the expression for the output y(n) in terms of x(n) for the multirate system shown in Fig. E11.6(a).*

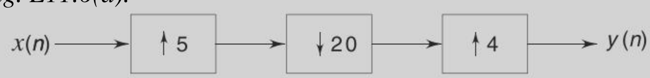


Fig. E11.6(a)

Solution The decimation with factor 20, can be represented as a cascade of two decimators with factors five and four. The resultant system is shown in Fig. E11.6(b).

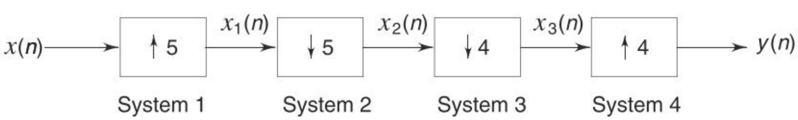


Fig. E11.6(b)

Systems 1 and 2 can be combined. The upsampler operation of System 1 is cancelled by the down-sampler operation of System 2 as shown in Fig. E11.6(c).

$x_2(n) = x(n)$

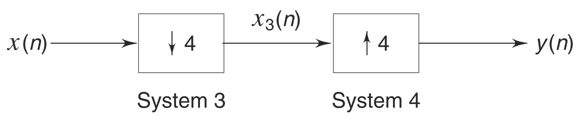


Fig. E11.6(c)

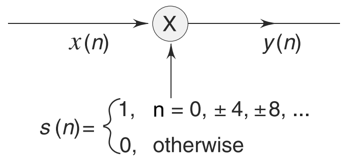


Fig. E11.6(d)

Combining systems 3 and 4, we get the system as shown in Fig. E11.6(d).

FILTER STRUCTURES 11.5

11.5.1 FIR Direct Form Structure

The general input-output relationship of an FIR filter is given by

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k)$$

The filter is represented by an N-point impulse response $h(k)$, i.e., $h(k)$ takes values only in the range $k = 0,\ldots, N-1$, outside this interval $h(k)$ is zero. Figure 11.14 shows the signal-flow graph representation of an FIR filter.

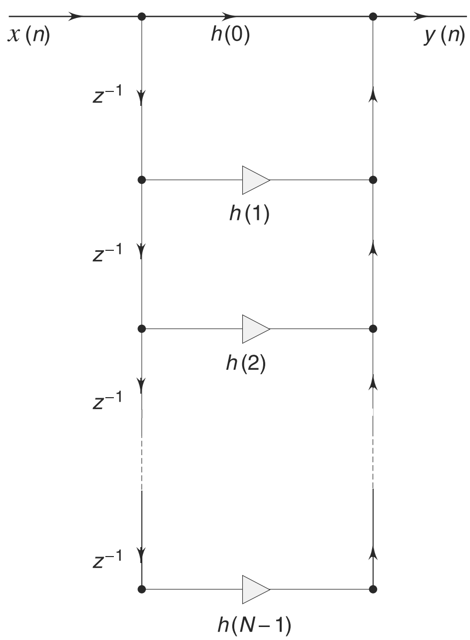


Fig. 11.14 *Signal-flow Graph Representation of an FIR Filter*

FIR filters are normally designed with linear phase. Hence the impulse response is symmetric given by

$$h(k) = h(N-1-k)$$

With this property, the number of multiplications can be reduced by a factor of two. If N is even,

$$y(n) = \sum_{k=0}^{\frac{N}{2}-1} h(k)\left\{x(n-k) + x\left(n-\left(N-1-k\right)\right)\right\}$$

11.5.2 IIR Direct Form Structure

Consider an IIR filter with the difference equation represented by

$$y(n) = \sum_{k=1}^{D} a_k y(n-k) + \sum_{k=0}^{N-1} b_k x(n-k) \tag{11.27}$$

Figure 11.15 shows the signal-flow graph for the IIR filter. The system equation of the IIR filter is given by

$$H(z)=\frac{Y(z)}{X(z)}=\frac{\sum_{k=0}^{N-1} b_k z^{-k}}{1-\sum_{k=1}^{D} a_k z^{-k}}=\frac{N(z)}{D(z)} \tag{11.28}$$

where $N(z)$ and $D(z)$ are numerator and denominator polynomials of $H(z)$ respectively. The signal-flow graph in Fig. 11.15 assumes $D = N - 1$. Feedforward part of the structure represents the numerator and feedback branches represent the denominator.

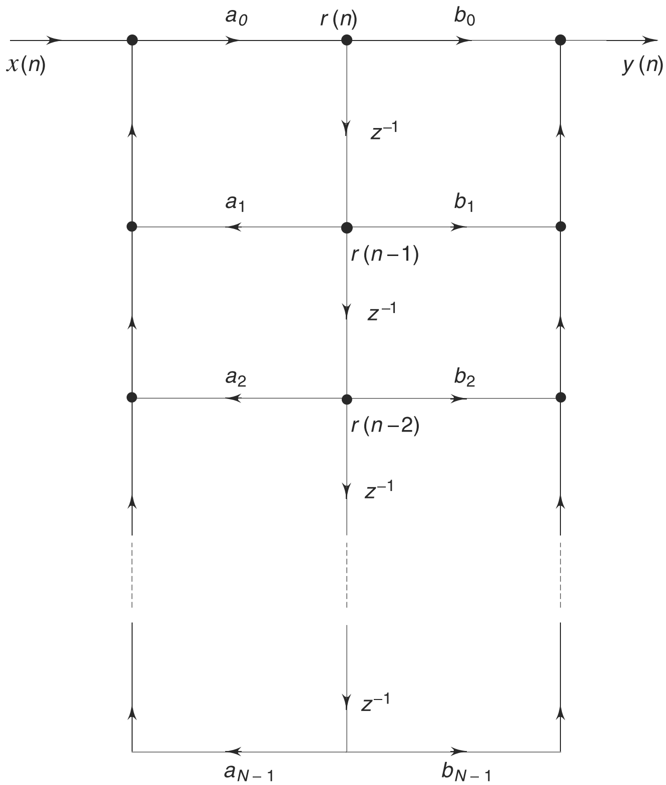


Fig. 11.15 *Signal-flow Graph for an IIR Filter*

11.5.3 Structures for FIR Decimators and Interpolators

Figure 11.16 shows the model of an M to 1 decimator. Let $x(n)$ be the input signal which is sampled with the sampling frequency of F. Let $y(m)$ be the output signal which is obtained by reducing the original sampling rate by a factor M. The filter $h(n)$ also operates at the sampling rate F.

Out of every M input samples, only one sample will be available at the output and the rest $M-1$ samples will be discarded by the M to 1 sampling rate compressor. All multiplications and summations are performed at the rate F. By applying the commutative operations of the network,

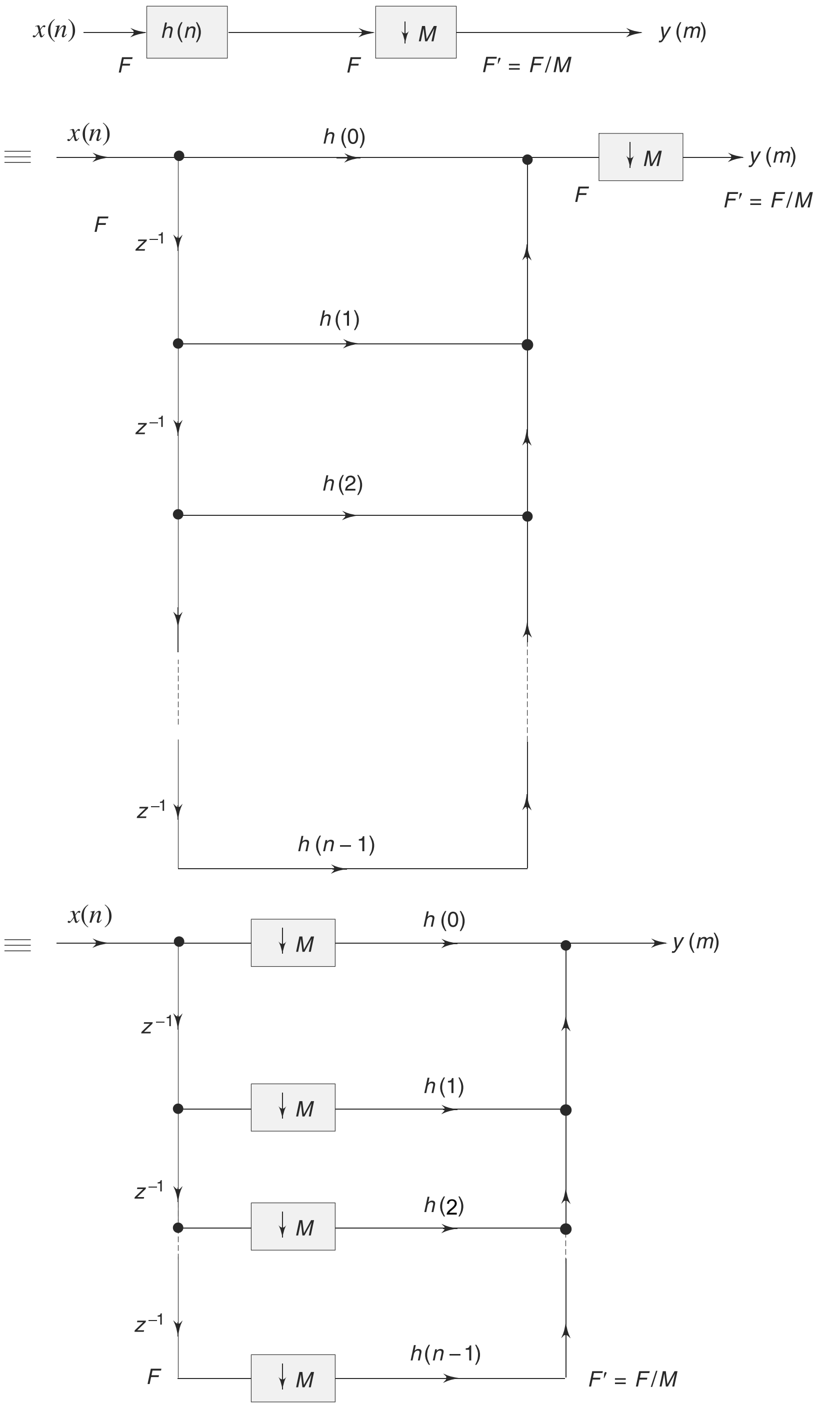


Fig. 11.16 *Direct Form Representation of M to 1 Decimator*

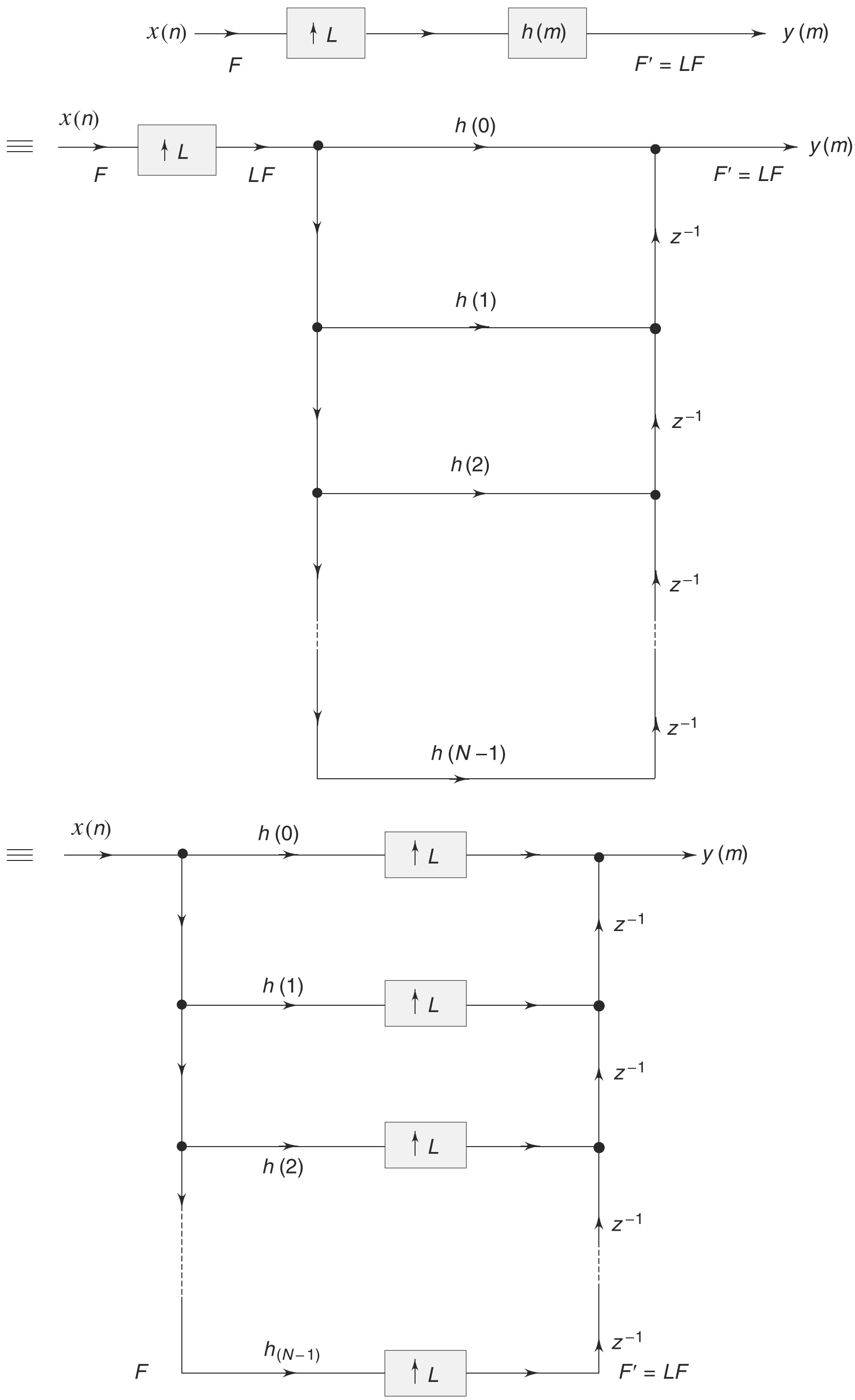


Fig. 11.17 *Direct Form Representation of the 1 to L Interpolator*

all multiplications and additions can be done at the low sampling rate of F/M. Hence, the total commutation rate is reduced by M.
The difference equation is given by

$$y(n) = \sum_{k=0}^{N-1} h(n)x(Mm-n) \tag{11.29}$$

Figure 11.17 shows the model of 1 to L interpolators. The output signal $y(m)$ is obtained by increasing the original sampling rate by a factor L.

Here, $h(M)$ is realised using the transposed direct form FIR structure. By applying the commutative property, the new structure has L times less computation when compared with the structure before commutation.

11.5.4 Frequency-Domain Characteristics for Decimation Filters

Frequency-Domain The spectral representation of the M to 1 decimator is shown in Fig. 11.18. Here, the sampling rate is reduced by a factor M. First, the input signal $x(n)$ is filtered by a low-pass filter to avoid aliasing during the decimation process (sampling rate compression).

The frequency response of the low-pass filter is given by

$$H\left(e^{j\omega}\right) = \begin{cases} 1, & 0 \le \omega < \pi / M \\ 0, & \text{otherwise} \end{cases} \tag{11.30}$$

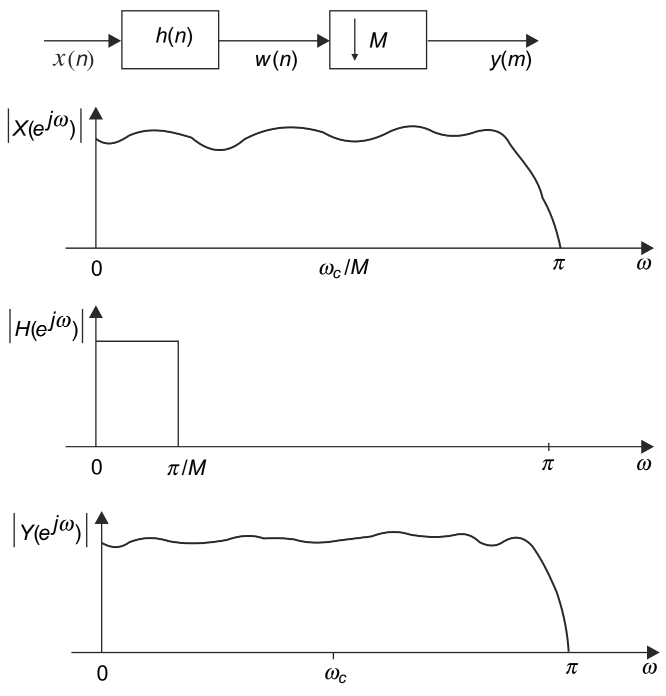


Fig. 11.18 *The Spectral Representation of the M to 1 Decimator*

11.5.5 Frequency Domain and Time-Domain Characteristics for Interpolation Filters

Frequency Domain The spectral representation of the 1 to L interpolator is given in Fig. 11.19. The input signal $x(n)$ is interpolated by a factor L to get the signal $y(m)$. $L - 1$ zero-valued samples are introduced between each pair of samples of $x(n)$. The resultant signal $\omega(m)$ contains along with the baseband spectrum, periodic repetitions of this spectrum. The signal $\omega(m)$ is now filtered with the filter with impulse response $h(m)$ to remove the repetitive spectrum of $\omega(m)$ and to retain only the baseband spectrum.

The frequency response of the filter is given by

$$H\left(e^{j\omega'}\right) = \begin{cases} L, & |\omega'| < \pi / L \\ 0, & \text{otherwise} \end{cases} \tag{11.31}$$

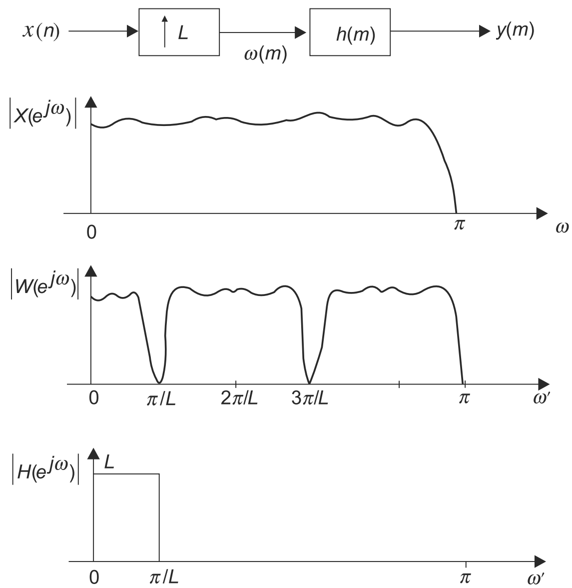


Fig. 11.19 *The Spectral Representation of the 1 to L Interpolator*

Time Domain By taking the inverse Fourier transform of $H(e^{j\omega})$ the time-domain representation, i.e. the impulse response of the filter can be obtained.

$$h(k) = \sin(\pi k / L) / (\pi k / L), \quad k = 0, \pm 1, \pm 2, \ldots \tag{11.32}$$

11.6 POLYPHASE DECOMPOSITION

The reduction of computational complexity in FIR filter realisation is possible by using the polyphase decomposition. It is also possible to reduce the computational complexity in the realisation of IIR decimation and interpolation filters in polyphase forms. The polyphase decomposition and its application in the realisation of the decimator and the interpolator are discussed in this section.

The z-transform of a filter with impulse response $h(n)$ is given by

$$H(z) = h(0) + z^{-1}h(1) + z^{-2}h(2) + \ldots \tag{11.33}$$

Rearranging the above equation, we get

$$H(z) = h(0) + z^{-2}h(2) + z^{-4}h(4) + \cdots + z^{-1}\left(h(1)\right) + z^{-3}h(3) + z^{-4}h(5) + \cdots) \tag{11.34}$$

Type I Polyphase Decomposition

$$H(z) = E_0(z^2) + z^{-1}E_1(z^2) \tag{11.35}$$

where $E_0(z^2)$ and $E_1(z^2)$ are polyphase components for a factor of two.

Type II Polyphase Decomposition

$$H(z) = z^{-1} R_0(z^2) + R_1(z^2) \tag{11.36}$$

where $R_0(z^2)$ and $R_1(z^2)$ are polyphase components for a factor of two.

Equations (11.35) and (11.36) represent two-branch polyphase decomposition of $H(z)$. In general, an M-branch type I polyphase decomposition is given by

$$H(z)=\sum_{k=0}^{M-1} z^{-k} E_k\left(z^M\right)$$

The transpose representation can be obtained by using

$$R_l\left(z^M\right)=E_{M-1-L}\left(z^M\right), \qquad L=0,1,\ldots,M-1$$

The Type II polyphase decomposition is given by

$$H(z)=\sum_{l=0}^{M-1} z^{-(M-1-L)} R_l\left(z^M\right)$$

The Type I and Type II polyphase decomposition representation is shown in Fig. 11.20.

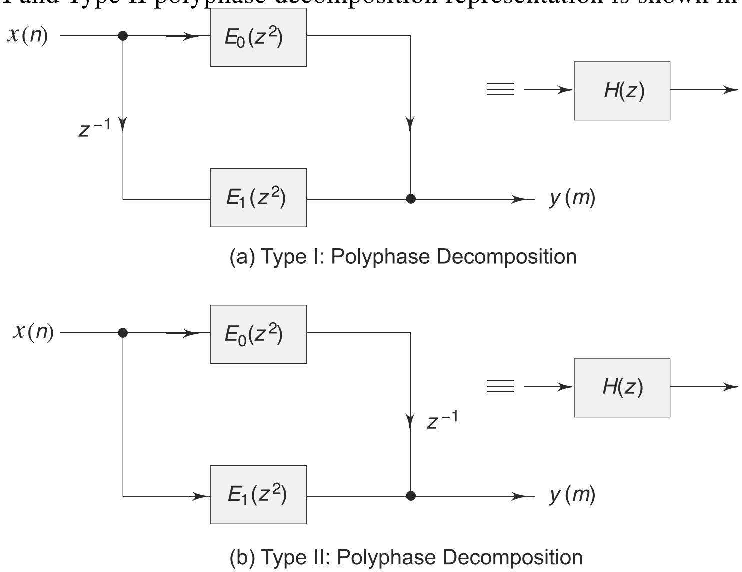


Fig. 11.20 *Types I and II Polyphase Decomposition Representation*

Example 11.7 *Develop a computationally efficient realisation of a factor-of-4 interpolator shown in Fig. E11.7(a), employing a length-16 linear-phase FIR filter.*

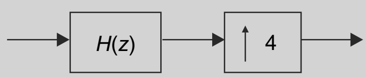


Fig. E11.7(a)

Solution

A computationally efficient realisation of the factor-of-4 interpolator is obtained by applying a 4-branch polyphase decomposition to $H(z)$ is shown in Fig. E11.7(b):

$$H(z)=E_0(z^4)+z^{-1}E_1(z^4)+z^{-2}E_2(z^4)+z^{-3}E_3(z^4)$$

Then moving the down-sampler through the polyphase filters resulting in further reduction in computational complexity is achieved by sharing common multipliers if $H(z)$ is a linear-phase FIR filter. For example, for a length-16 Type II FIR transfer function, a computationally efficient factor-of-4 interpolator structure based on the above equation is as shown in Fig. E11.7(c).

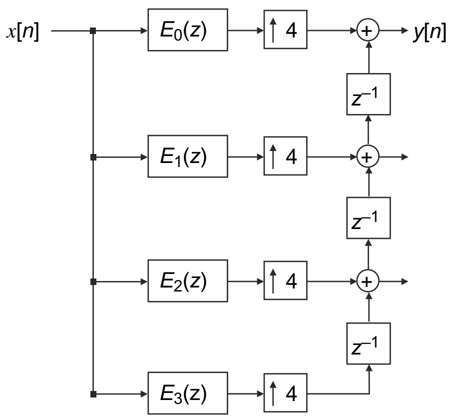


Fig. E11.7(b)

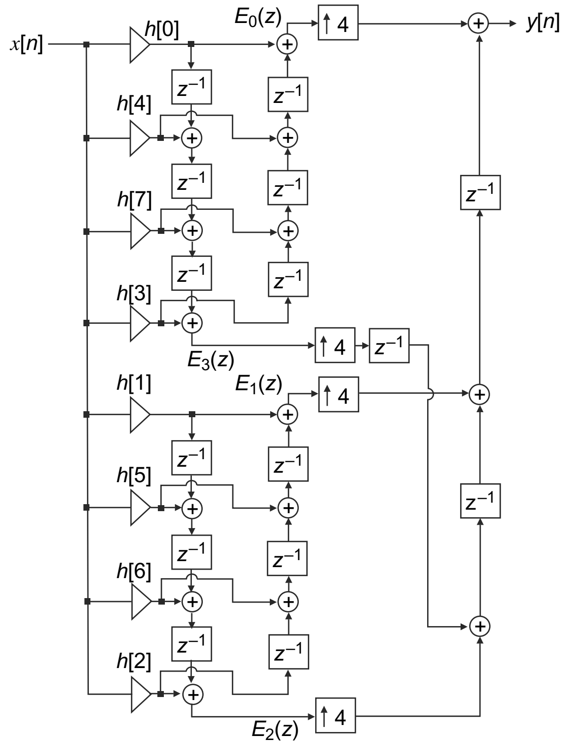


Fig. E11.7(c)

Example 11.8 *The transfer function of an FIR filter is given by*

$$H(z) = 0.2 + 0.7z^{-1} + 0.8z^{-2} + 0.15z^{-3} + 0.6z^{-4} + 0.32z^{-5} + 0.5z^{-6} + 0.4z^{-7} + 0.9z^{-8}$$

Obtain the polyphase decomposition of H(z) to decompose into (a) 2 sections, (b) 3 sections, and (c) 4 sections.

Solution

(a) Polyphase decomposition into 2 sections

Given $H(z) = 0.2 + 0.7z^{-1} + 0.8z^{-2} + 0.15z^{-3} + 0.6z^{-4} + 0.32z^{-5} + 0.5z^{-6} + 0.4z^{-7} + 0.9z^{-8}$

$H(z)$ can be decomposed into 2 sections as

$$H(z) = \left[0.2 + 0.8z^{-2} + 0.6z^{-4} + 0.5z^{-6} + 0.9z^{-8}\right] + \left[0.7z^{-1} + 0.15z^{-3} + 0.32z^{-5} + 0.4z^{-7}\right]$$

$$= \left[0.2 + 0.8z^{-2} + 0.6z^{-4} + 0.5z^{-6} + 0.9z^{-8}\right] + z^{-1}\left[0.7 + 0.15z^{-2} + 0.32z^{-4} + 0.4z^{-6}\right]$$

$$= E_o\left(z^2\right) + z^{-1}E_1\left(z^2\right)$$

Hence the polyphase components of 2 sections are

$$E_0\left(z^2\right) = 0.2 + 0.8z^{-2} + 0.6z^{-4} + 0.5z^{-6} + 0.9z^{-8}$$

$$E_1\left(z^2\right) = 0.7 + 0.15z^{-2} + 0.32z^{-4} + 0.4z^{-6}$$

(b) Polyphase decomposition into 3 sections

Given $H(z) = 0.2 + 0.7z^{-1} + 0.8z^{-2} + 0.15z^{-3} + 0.6z^{-4} + 0.32z^{-5} + 0.5z^{-6} + 0.4z^{-7} + 0.9z^{-8}$

$H(z)$ can be decomposed into 3 sections as

$$H(z) = \left[0.2 + 0.15z^{-3} + 0.5z^{-6}\right] + \left[0.7z^{-1} + 0.6z^{-4} + 0.4z^{-5} + 0.4z^{-7}\right] + \left[0.8z^{-2} + 0.32z^{-5} + 0.9z^{-8}\right]$$

$$= \left[0.2 + 0.15z^{-3} + 0.5z^{-6}\right] + z^{-1}\left[0.7 + 0.6z^{-3} + 0.4z^{-6}\right] + z^{-2}\left[0.8 + 0.32z^{-3} + 0.9z^{-6}\right]$$

$$= E_0\left(z^3\right) + z^{-1}E_1\left(z^3\right) + z^{-2}E_2\left(z^3\right)$$

Hence, the polyphase components of 3 sections are

$$E_0\left(z^3\right) = 0.2 + 0.15z^{-3} + 0.5z^{-6}$$

$$E_1\left(z^3\right) = 0.7 + 0.6z^{-3} + 0.4z^{-6}$$

$$E_2\left(z^3\right) = 0.8 + 0.32z^{-3} + 0.9z^{-6}$$

(c) Polyphase decomposition into 4 sections

Given $H(z) = 0.2 + 0.7z^{-1} + 0.8z^{-2} + 0.15z^{-3} + 0.6z^{-4} + 0.32z^{-5} + 0.5z^{-6} + 0.4z^{-7} + 0.9z^{-8}$

$H(z)$ can be decomposed into 4 sections as

$$H(z) = \left[0.2 + 0.6z^{-4} + 0.9z^{-8}\right] + \left[0.7z^{-1} + 0.32z^{-5}\right] + \left[0.8z^{-2} + 0.5z^{-6}\right] + \left[0.15z^{-3} + 0.4z^{-7}\right]$$

$$= E_0\left(z^4\right) + z^{-1}E_1\left(z^4\right) + z^{-2}E_2\left(z^4\right) + z^{-3}E_3\left(z^4\right)$$

Hence, the polyphase components of 4 sections are

$$E_0(z^4) = 0.2 + 0.6z^{-4} + 0.9z^{-8}$$

$$E_1(z^4) = 0.7 + 0.32z^{-4}$$

$$E_2(z^4) = 0.8 + 0.5z^{-4}$$

$$E_3(z^4) = 0.15 + 0.4z^{-4}$$

Example 11.9 *The transfer function of an IIR filter is given by*

$$H(z) = \frac{1 + 0.7z^{-1}}{1 - 0.9z^{-1}}$$

Obtain the polyphase decomposition of H(z) to decompose into (a) 2 sections, and (b) 4 sections.

Solution Given $H(z) = \dfrac{1 + 0.7z^{-1}}{1 - 0.9z^{-1}}$

$$= \frac{1 + 0.7z^{-1}}{1 - 0.9z^{-1}} \times \frac{1 + 0.9z^{-1}}{1 + 0.9z^{-1}} = \frac{1 + 0.9z^{-1} + 0.7z^{-1} + 0.63z^{-2}}{1 - 0.81z^{-2}}$$

$$= \frac{1 + 1.6z^{-1} + 0.63z^{-2}}{1 - 0.81z^{-2}}$$

(a) Polyphase decomposition into 2 sections

$$H(z)=\frac{1+1.6z^{-1}+0.63z^{-2}}{1-0.81z^{-2}}=\frac{1+0.63z^{-2}}{1-0.81z^{-2}}+z^{-1}\frac{1.6}{1-0.81z^{-2}}=E_0\left(z^2\right)+z^{-1}E_1\left(z^2\right)$$

Hence, the polyphase components of 2 sections are

$$E_0\left(z^2\right)=\frac{1+0.63z^{-2}}{1-0.81z^{-2}} \text{ and } E_1\left(z^2\right)=\frac{1.6}{1-0.81z^{-2}}$$

(b)Polyphase decomposition into 4 sections

$$H\left(z\right)=\frac{1+1.6z^{-1}+0.63z^{-2}}{1-0.81z^{-2}}$$

$$=\frac{1+1.6z^{-1}+0.63z^{-2}}{1-0.81z^{-2}}\times\frac{1+0.81z^{-2}}{1+0.81z^{-2}}$$

$$=\frac{1+1.6z^{-1}+1.44z^{-2}+1.296z^{-3}+0.5103z^{-4}}{1-0.6561z^{-4}}$$

$$=\frac{\left(1+0.5103z^{-4}\right)+z^{-1}\left(1.6\right)+z^{-2}\left(1.44\right)+z^{-3}\left(1.296\right)}{1-0.6561z^{-4}}$$

$$=\frac{1+0.5103z^{-4}}{1-0.6561z^{-4}}+z^{-1}\frac{1.6}{1-0.6561z^{-4}}+z^{-2}\frac{1.44}{1-0.6561z^{-4}}+z^{-3}\frac{1.296}{1-0.6561z^{-4}}$$

$$=E_0\left(z^4\right)+z^{-1}E_1\left(z^4\right)+z^{-2}E_2\left(z^4\right)+z^{-3}E_3\left(z^4\right)$$

Hence, the polyphase components of 4 sections are

$$E_0\left(z^4\right)=\frac{1+0.5103z^{-4}}{1-0.6561z^{-4}};\qquad E_1\left(z^4\right)=\frac{1.6}{1-0.6561z^{-4}}$$

$$E_2\left(z^4\right)=\frac{1.44}{1-0.6561z^{-4}};\qquad E_3\left(z^4\right)=\frac{1.296}{1-0.6561z^{-4}}$$

11.6.1 General Polyphase Framework

The z-transform of an anti-aliasing filter shown in Fig. 11.21(a) with impulse response $h(n)$ is given by

$$H(z)=\sum_{n=0}^{\infty}h(n)z^{-n}$$
$$= h(0) + h(1)z^{-1} + h(2)z^{-2} + \cdots$$

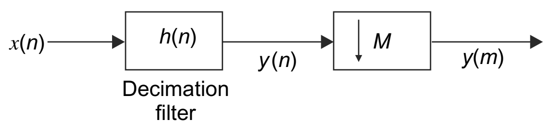


Fig. 11.21(a) *Decimation by a Factor M*

which can be partitioned into M subsignals where M represents decimation factor. Hence,

$$\begin{aligned}H\left(z\right)&=h\left(0\right)+h\left(M\right)z^{-M}+h\left(2M\right)z^{-2M}+\cdots\\&+z^{-1}\left\{h\left(1\right)+h\left(M+1\right)z^{-M}+h\left(2M+1\right)z^{-2M}+\cdots\right\}\\&+z^{-(M-1)}\left\{h\left(M-1\right)+h\left(2M-1\right)z^{-M}+\cdots\right\}\end{aligned}\tag{11.37}$$

Equation (11.37) can be written as

$$H(z) = \sum_{k=0}^{M-1}\sum_{m=0}^{\infty} h(mM+k)z^{-(mM+k)} = \sum_{k=0}^{M-1} z^{-k}\sum_{m=0}^{\infty} h(mM+k)z^{-mM}$$
$$= E_0\left(z^M\right) + z^{-1}E_1\left(z^M\right) + \cdots + z^{-(M-1)}E_{M-1}\left(z^M\right) \quad (11.38)$$

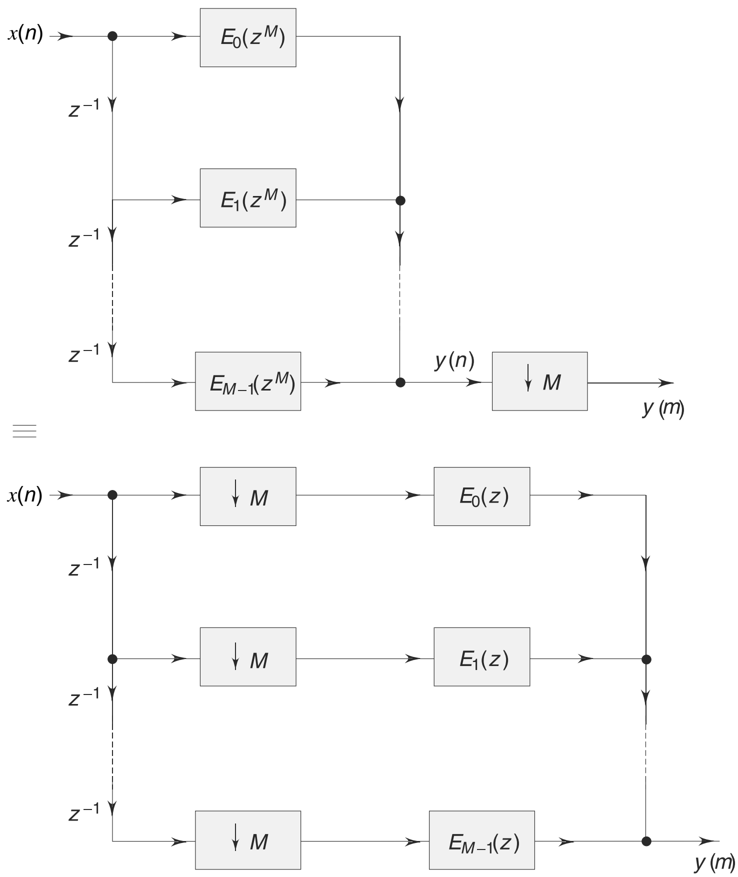


Fig. 11.21(b) *Polyphase Representation of Decimators*

where the polyphase components

$E_0(z^M) = H(z)|_k = 0$, $E_1(z^M) = H(z)|_k = 1$, and so on.

The matrix representation is

$$H(z) = \left[1, z^{-1}, \ldots, z^{-(M-1)}\right]\begin{pmatrix} E_0\left(z^M\right) \\ E_1\left(z^M\right) \\ \vdots \\ \vdots \\ E_{M-1}\left(z^{M-1}\right) \end{pmatrix}$$

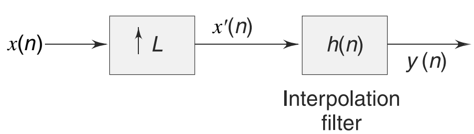


Fig. 11.21(c) *Interpolation by a Factor L*

With these polyphase components, the anti-aliasing filter and its equivalent using decimator identity can be represented as shown in Fig. 11.21(b). Similarly, with the polyphase components $R_0(z^L)$, $R_1(z^L)$,…, $R_{L-1}(z^L)$ of the interpolation filter shown in Fig. 11.21(c) and its equivalent using interpolator identity are represented in Fig. 11.21(d).

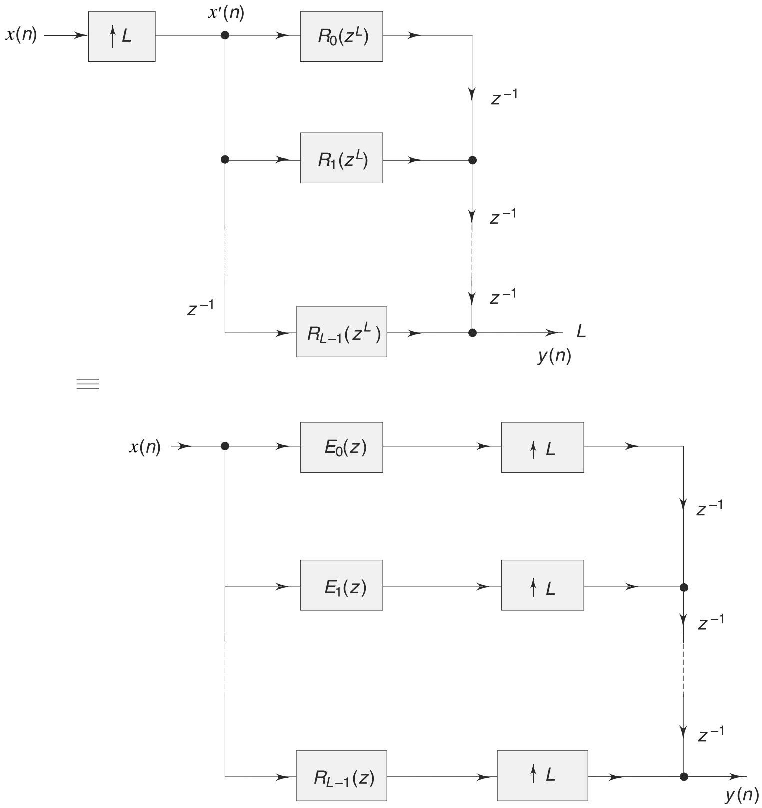


Fig. 11.21(d) *Polyphase Representation of Interpolators*

11.6.2 Polyphase FIR Filter Structures for Decimators and Interpolators

Decimator The decimators consisting of an anti-aliasing filter $h(n)$ and a downsampler by a factor two is shown in Fig. 11.22(a).

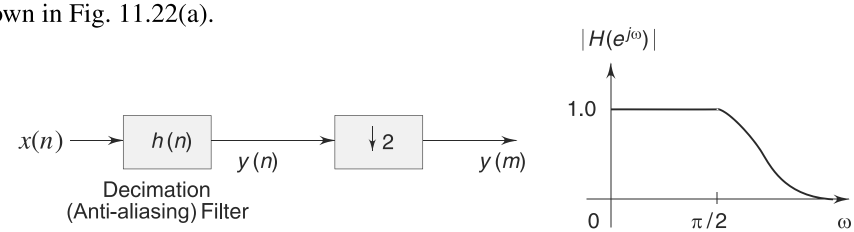


Fig. 11.22(a) *Decimation by a Factor Two and its Frequency Response*

Assuming that the anti-aliasing FIR filter with N coefficients, the filtering described by the convolution $y(n) = x(n) * h(n) = \sum_{k=0}^{N-1} x(k)h(n-k)$ and the downsampling as $y(m) = y(2n)$. Figure 11.22(b) represents the direct implementation of an FIR filter in transversal structure. This structure computes all the values of $y(n)$. But $y(1)$, $y(3)$, ···, and so on, are not necessary for further process, i.e. they have been calculated unnecessarily which leads to more number of multiplications and additions.

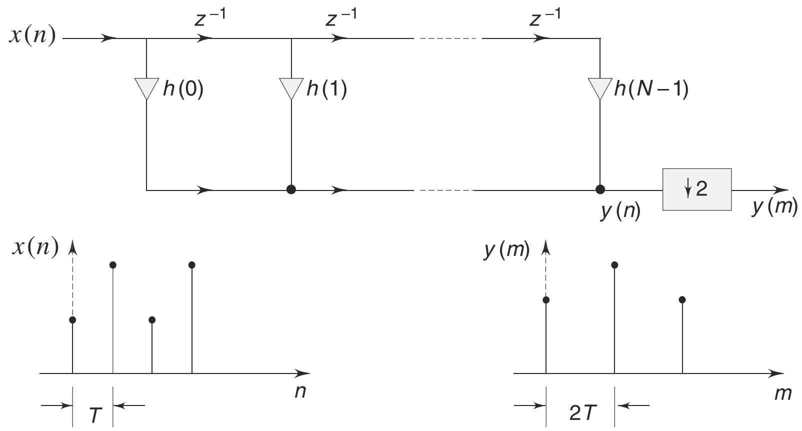


Fig. 11.22(b) *Direct Implementation of the Decimator by a Factor of Two*

As an alternative to the efficient transversal filter structures, polyphase filter structures can be used. The polyphase structures introduced here are of fundamental importance for analysis and synthesis of filter banks. The polyphase implementation using decimator identities as shown in Fig. 11.22(c) results in a reduced number of multiplications and additions.

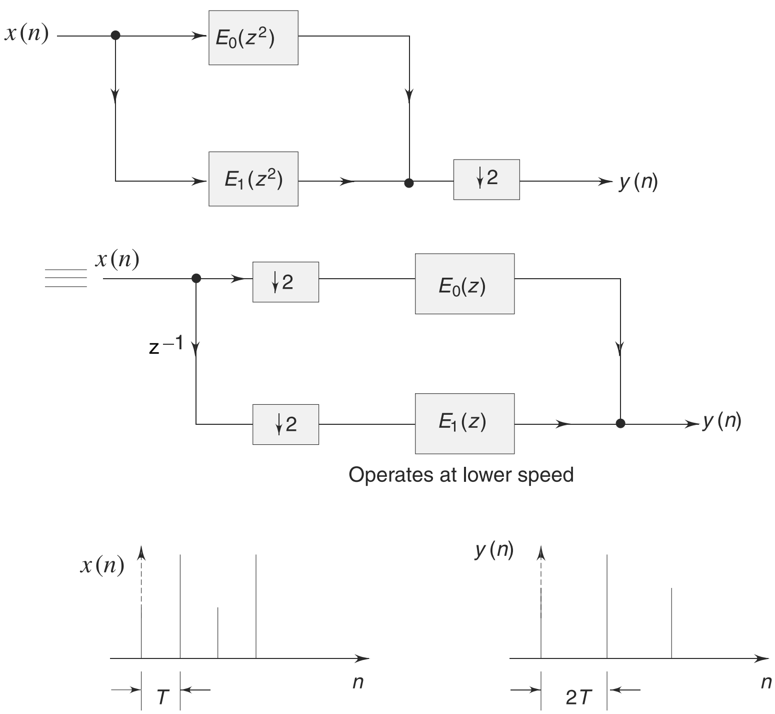


Fig. 11.22(c) *Polyphase Implementation of Decimator by a Factor of Two*

Interpolator The interpolator is the dual of decimator. The respective signal-flow graph can be derived form each other with the direction of all signals reversed, downsamplers and upsamplers interchanged and inputs and outputs swapped. Because of this, the interpolator structures derived in this section are of similar form to the decimator structures. The interpolator consisting of an interpolation (anti-imaging) filter $h(n)$ and an upsampler by a factor two is shown in Fig. 11.22(d).

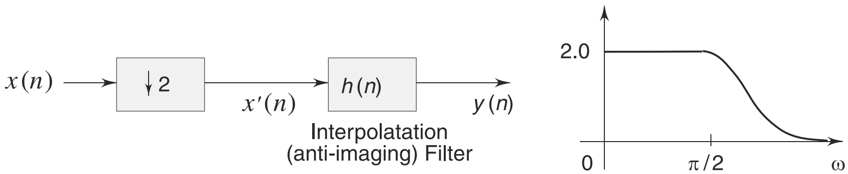


Fig. 11.22(d) *Interpolating by Factor Two and its Frequency Response*

Assuming that the anti-imaging FIR filter has N coefficients, the filtering described by $y(n) = x'(n)^{*}\, h(n) = \sum_{k=0}^{N-1} x'(k)h(n-k)$. Figure 11.22(e) shows the direct implementation of the FIR filter in transversal structure. The polyphase implementation structure with its interpolator identity is shown in Fig. 11.22(f).

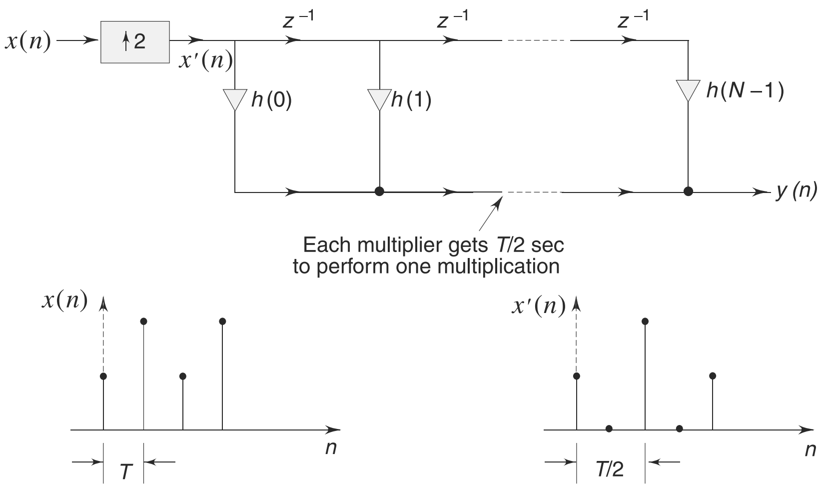


Fig. 11.22(e) *Direct Implementation in Transverse Structure*

IIR Structures for Decimators

The IIR filter is represented by the difference equation,

$$y(n) = \sum_{k=1}^{D} a_k y(n-k) + \sum_{k=0}^{N-1} b_k x(n-k) \tag{11.39}$$

The system function for the above difference equation is given by

$$H(z) = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{1 - \sum_{k=1}^{D} a_k z^{-k}} = \frac{N(z)}{D(z)} \tag{11.40}$$

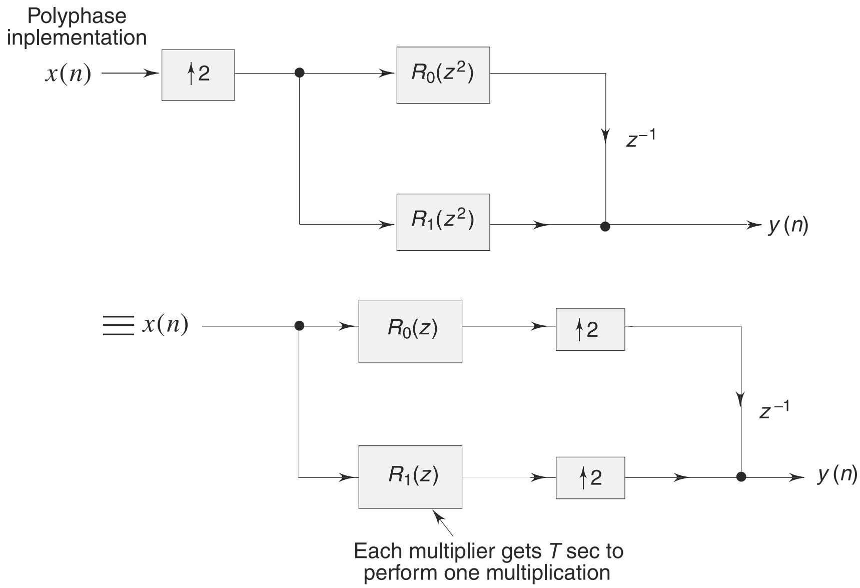


Fig. 11.22(f) *Polyphase Implementation of an Interpolator by a Factor of Two*

Let $D = N - 1$, so that the numerator and denominator orders are same. Figure 11.23 shows the direct form of the IIR structure for an M to 1 decimator.

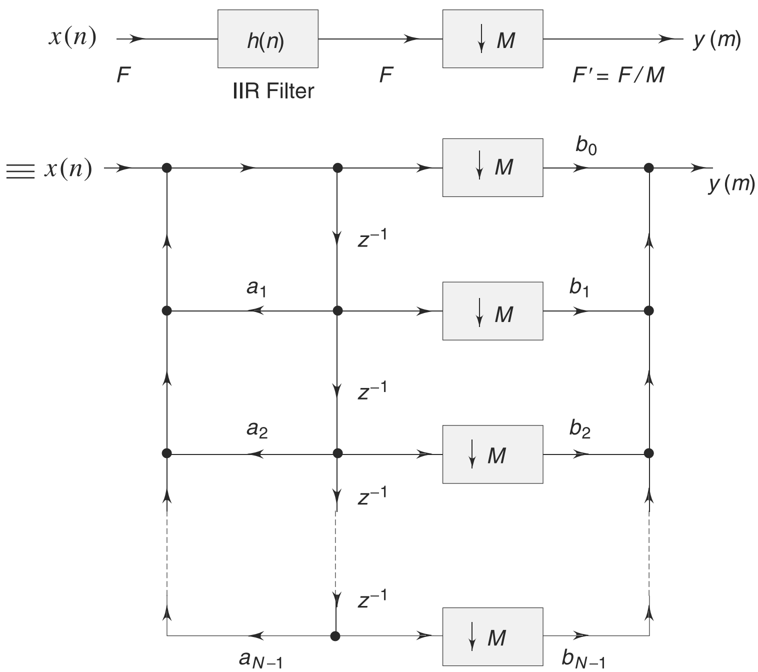


Fig. 11.23 *Direct Form of the IIR Structure for an M to 1 Decimator*

Polyphase IIR Filter Structures for Decimators

Consider the IIR transfer function $H(z) = \dfrac{N(z)}{D(z)}$. The polyphase decomposition is obtained as follows.

Express $H(z)$ in the form $\dfrac{N'(z)}{D'\left(z^M\right)}$ by multiplying and dividing $H(z)$ with a properly chosen polynominal, apply polyphase decomposition to $N'(z)$. Figure 11.24 shows the polyphase IIR filter structures for decimators.

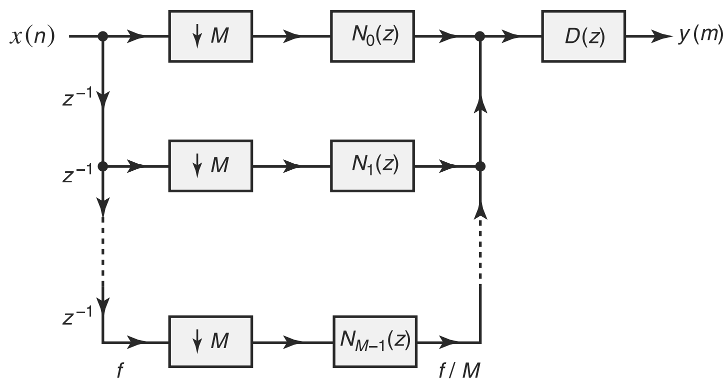


Fig. 11.24 *Polyphase IIR Filter Structures for Decimators*

DIGITAL FILTER DESIGN 11.7

The basic model for sampling rate conversion by a rational factor L/M is shown in Fig. 11.25.

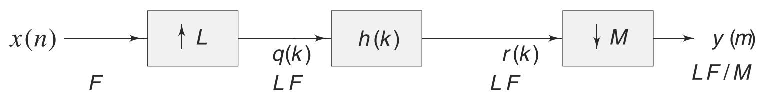


Fig. 11.25 *Basic Model for Sampling Rate Conversion by a Rational Factor L/M*

11.7.1 Frequency and Time Domain Characteristics

The various methods for FIR filter design are given below:

(i) Window method
(ii) Optimal, equiripple linear phase method
(iii) Half-band designs
(iv) FIR interpolator design based on-time domain filter specifications
(v) Classical interpolation designs: linear and Lagrangian

Let us consider the equiripple FIR filter design. The equiripple response of the filter is shown in Fig. 11.26.

The design equations for calculating the stopband and passband frequencies are discussed below. Let the highest frequency of the decimated signal or the total bandwidth of the interpolated signal be $\omega_c \leq \pi$, then the passband frequency is given by

$$\omega_p \leq \begin{cases} \omega_c / L, & 1 \text{ to } L \text{ interpolator} \\ \omega_c / M, & M \text{ to } 1 \text{ decimator} \\ \min\left(\omega_c / L, \omega_c / M\right), & \text{conversion by } L / M \end{cases} \tag{11.41}$$

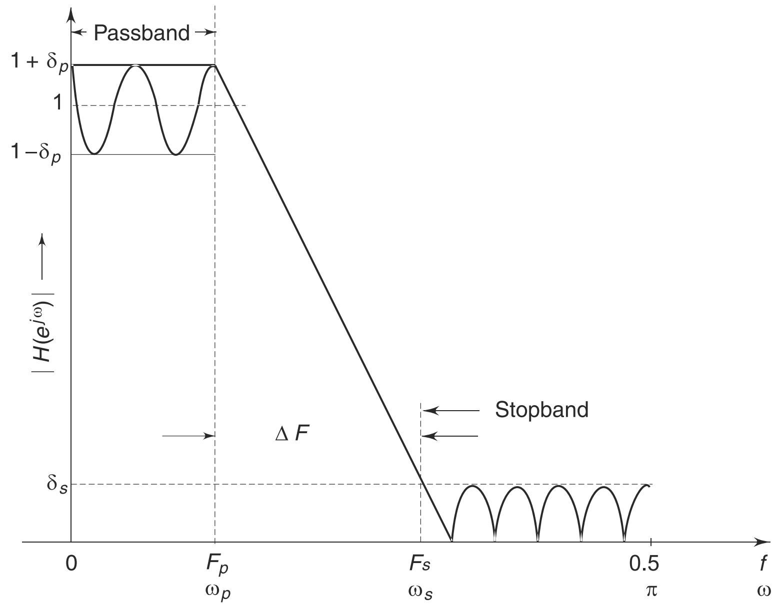


Fig. 11.26 *Equiripple Response of the Filter*

The stopband frequency is given by

$$\omega_s \le \begin{cases} \pi/L, & 1 \text{ to } L \text{ interpolator} \\ \pi/M, & M \text{ to 1 decimator} \\ \min(\pi/L, \pi/M), & \text{conversion by } L/M \end{cases} \tag{11.42}$$

The assumption is that there is no aliasing in the decimator or imaging in the interpolator. If aliasing is allowed in the decimator or interpolator, then the stopband frequency is given by

$$\omega_s = \begin{cases} (2\pi-\omega_c)/L, & 1 \text{ to } L \text{ interpolator} \\ (2\pi-\omega_c)/M, & M \text{ to 1 decimator} \\ \min\left((2\pi-\omega_c)/L, (2\pi-\omega_c)/M\right), & \text{conversion by } L/M \end{cases} \tag{11.43}$$

11.7.2 Filter Design for IIR Interpolators and Decimators

The ideal characteristic for the IIR prototype filter $h(k)$ assuming that no constraints are set for the phase, is given by

$$H\left(e^{j\omega}\right) = \begin{cases} e^{j\phi}(\omega'), & |\omega'| \le \pi/M \\ 0, & \text{otherwise} \end{cases} \tag{11.44}$$

This is for an M to 1 decimator. For an interpolator, the ideal characteristic becomes,

$$H\left(e^{j\omega}\right) = \begin{cases} e^{j\phi}(\omega'), & |\omega'| \le \pi/L' \\ 0, & \text{otherwise} \end{cases} \tag{11.45}$$

The system function for an IIR filter is given by

$$H(z) = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{1 - \sum_{k-1}^{D} a_k z^{-k}} = \frac{N(z)}{D(z)} \tag{11.46}$$

Represent the denominator polynomial as a polynomial of order R in z^M, where M is the decimation factor. Replace

$$\sum_{k=1}^{D} a_k z^{-k} \text{ by } \sum_{k=1}^{R} c_k z^{-rM}$$

For IIR filter design, the following approximations are used.

(i) The Butterworth approximation
(ii) The Bessel approximation
(iii) The Chebyshev approximation
(iv) The Elliptic approximation

MULTISTAGE DECIMATORS AND INTERPOLATORS 11.8

In practical applications, mostly sampling-rate conversion by a rational factor L/M is required. Figure 11.27 represents the general structure of system where this conversion is used.

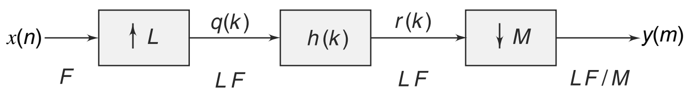


Fig. 11.27 *Sampling Rate Conversion by a Rational Factor L/M*

Consider a system for decimating a signal by an integer factor M. Let the input signal sampling frequency be f_0, then the decimated signal frequency will be f_0/M.

The decimation factor can be factorised as

$$M = \prod_{i=1}^{I} M_i \tag{11.47}$$

The resultant network is shown in Fig. 11.28.

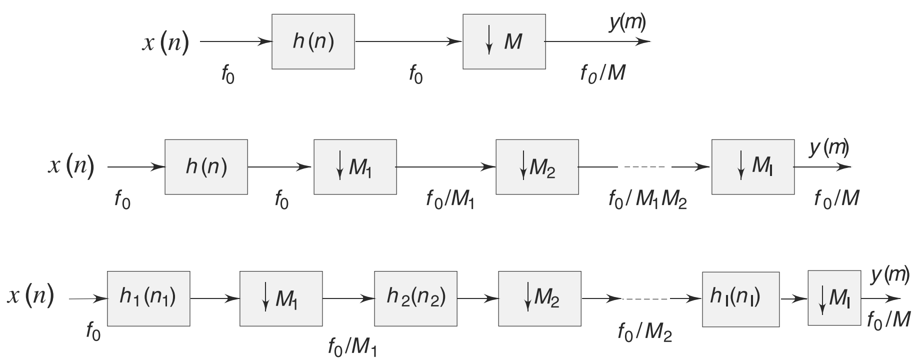


Fig. 11.28 *Multistage Decimator*

Similarly, the multistage interpolator is shown in Fig. 11.29. The 1 to L interpolator, has its interpolation factor represented by

$$L = L_1 L_2 \dots L_i = \prod_{i=1}^{I} L_i \tag{11.48}$$

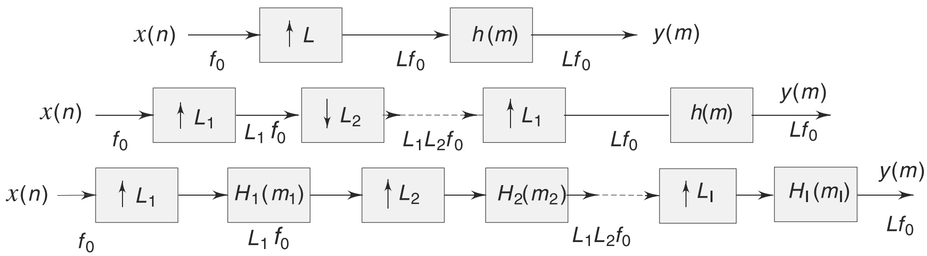


Fig. 11.29 *Multistage Interpolators*

If the sampling rate alteration system is designed as a cascade system, there is a significant savings in computational complexity. Then the computational efficiency is improved significantly.

The reasons for using multistage structures are

(i) Overall filter length is reduced
(ii) Reduction in the multiplication rate and hence, multistage systems require reduced computation
(iii) Storage space required is less
(iv) Filter design problem is simple
(v) Finite word length effects are less

The demerits of the systems are that proper control structure is required in implementing the system and proper values of L should be chosen.

Example 11.10 *Implement a two-stage decimator for the following specifications.*

Sampling rate of the input signal = 20 kHz
M = 100
Passband = 0 *to* 40 Hz
Transition band = 40 to 50 Hz
Passband ripple = 0.01
Stopband ripple = 0.002

Solution The implementation of the system is shown in Fig. E11.10(a).

$F_p = 40$ Hz $\qquad F_s = 50$ Hz
$\delta_p = 0.01 \qquad \delta_s = 0.002$
$F_T = 20$ kHz $\qquad M = 100$

For an equiripple linear phase FIR filter, the length N is given by

$$N = \frac{-20\log_{10}\sqrt{\delta_p \delta_s} - 13}{14.6\Delta f}$$

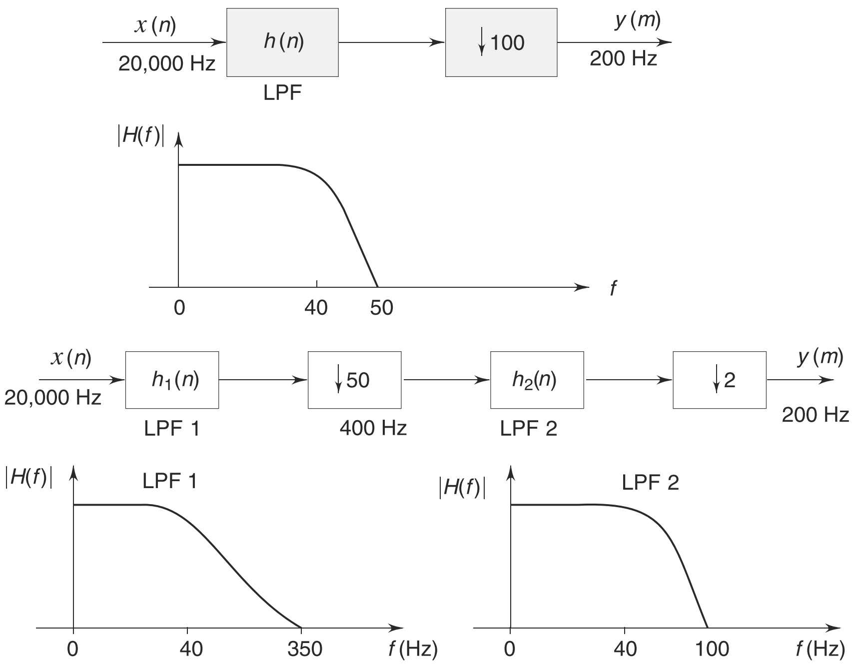


Fig. E11.10(a) *Single-stage and Two-stage Network for Decimator*

where $\Delta f = \dfrac{F_s - F_p}{F_T}$ is the normalised transition bandwidth.

$$N = \frac{-20\log_{10}\sqrt{(0.01)(0.002)} - 13}{14.6\left(\dfrac{50-40}{20,000}\right)} = 4656$$

In the single-stage implementation, the number of multiplications per second is

$$R_{M,H} = 4656 \times \frac{20,000}{100} = 9,31,200$$

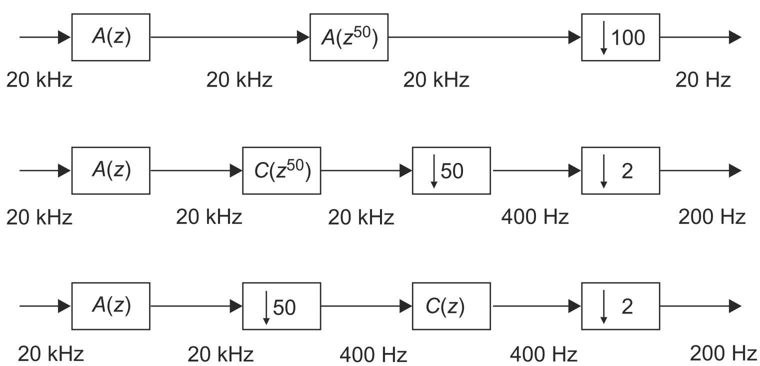


Fig. E11.10(b) *Two-stage Realisation of the Decimator Structure*

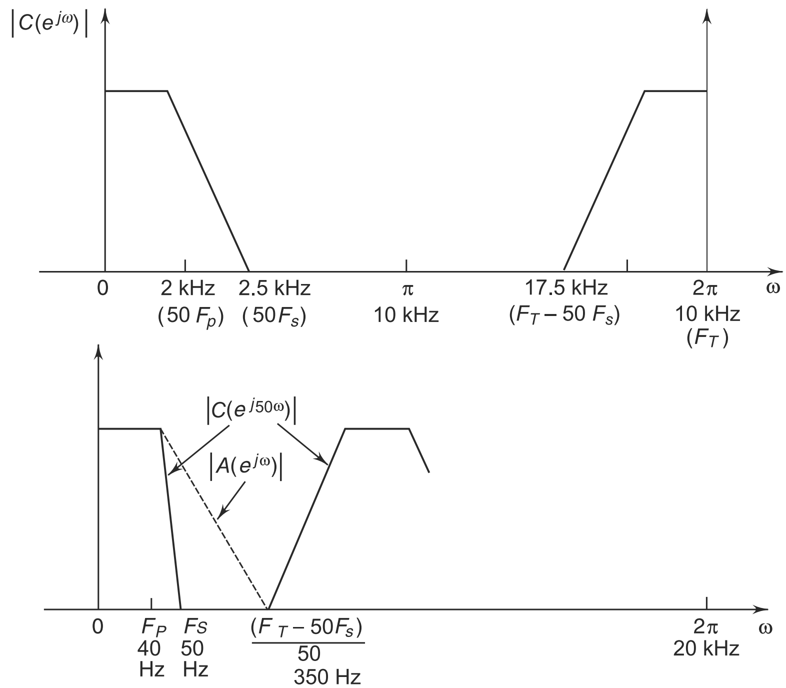


Fig. E11.10(c) *Magnitude Response for a Two-state Decimator*

Two-stage Realisation

$H(z)$ can be implemented as a cascade realisation in the form of $C(z^{50})\,A(z)$. The steps in the two-stage realisation of the decimator structure is shown in Fig. E11.10(b) and the magnitude response are shown in Fig. E11.10(c).

For the cascade realisation, the overall ripple is the sum of the passband ripples of $A(z)$ and $C(z^{50})$. To maintain the stopband ripple at least as good as $A(z)$ or $C(z^{50})$, δ_s for both can be 0.002.

For $C(z)$, $\delta_p = 0.005$, $\delta_s = 0.002$

$$\Delta f = \frac{500}{20,000} = 0.025$$

The filter length 'N' for $C(z)$ is calculated as

$$N = \frac{-20\log_{10}\sqrt{(0.005)(0.002)} - 13}{14.6\Delta f} = \frac{(50-13)}{14.6\times 0.025} = 102$$

For $A(z)$, $\delta_p = 0.005$, $\delta_s = 0.002$

$$\Delta f = \frac{350-40}{20,000} = 0.0155$$

The filter length 'N' for $A(z)$ is calculated as

$$N = \frac{-20\log_{10}\sqrt{(0.005)(0.002)} - 13}{14.6\Delta f} = \frac{(50-13)}{14.6\times 0.0155} = 164$$

The length of the overall filter in cascade is

$$= 164 + (50 \times 102) + 2 = 5{,}266$$

The filter length in cascade realisation has increased but the number of multiplications per second (computational complexity) is reduced.

$$R_{M,C} = 102 \times \frac{400}{2} = 20{,}400$$

$$R_{M,A} = 164 \times \frac{20{,}000}{50} = 65{,}600$$

Total number of multiplications per second is

$$R_{M,C} + R_{M,A} = 20{,}400 + 65{,}600 = 86{,}000$$

Example 11.11 *Compare the single-stage, two-stage, three-stage and multistage realisation of the decimator with the following specifications.*

Sampling rate of a signal has to be reduced from 10 kHz *to* 500 Hz. *The decimation filter* $H(z)$ *has the passband edge* (F_p) *to be* 150 Hz, *stopband edge* (F_s) *to be* 180 Hz, *passband ripple* (δ_p) *to be* 0.002 *and stopband ripple* (δ_s) *to be* 0.001.

Solution

Single-stage Realisation

The length N of an equiripple linear phase FIR filter is given by

$$N = \frac{-20\log_{10}\sqrt{\delta_p\delta_s} - 13}{14.6\Delta f}$$

where $\Delta f = \dfrac{F_s - F_p}{F_T}$ is the nomalised transition bandwidth.

$$F_T = 10 \text{ kHz}$$

$$N = \frac{-20\log_{10}\sqrt{(0.002)(0.001)} - 13}{14.6\left(\dfrac{180-150}{10{,}000}\right)} = 1{,}004$$

For the single-stage implementation of the decimator with a factor of 20, the number of multiplications per second is given by

$$R_{M,H} = 1004 \times \frac{10{,}000}{20} = 5{,}02{,}000$$

Two-stage Realisation

$H(z)$ can be implemented as a cascade realisation in the form of $C(z^{10})\,A(z)$. The steps in the two-stage realisation of the decimator structure are shown in Fig. E11.11(a) and the magnitude response is shown in Fig. E11.11(b).

For the cascade realisation, the overall ripple is the sum of the passband ripples of $A(z)$ and $C(z^{10})$. To maintain the stopband ripple at least as good as $A(z)$ or $C(z^{10})$, δ_s for both can be 0.001. The specifications for the interpolated FIR filters (IFIR) is given by

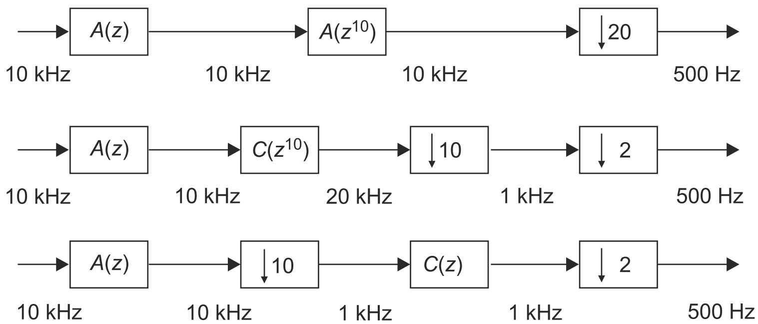


Fig. E11.11(a) *Two-stage Realisation of the Decimator Structure*

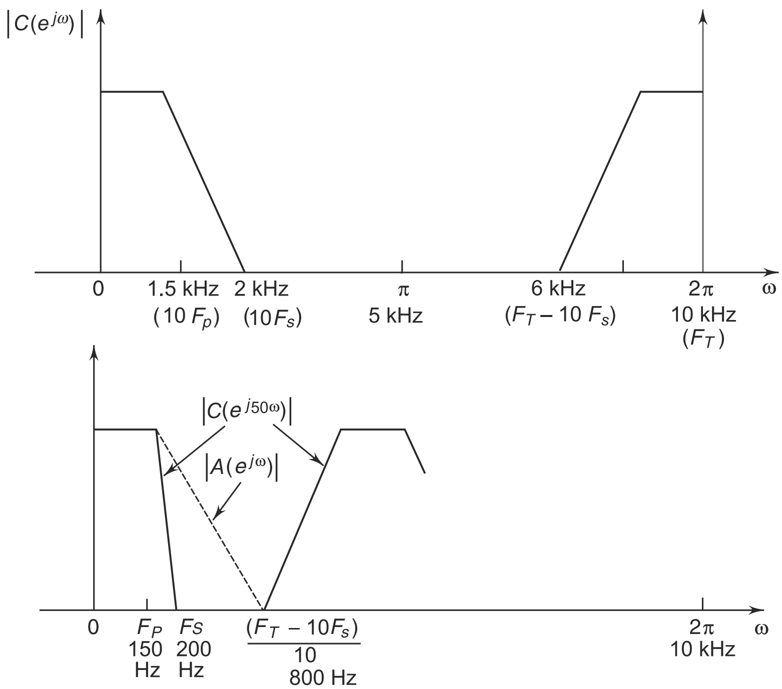


Fig. E11.11(b) *Magnitude Response for a Two-stage Decimator*

For $C(z)$, $\delta_p = 0.001$, $\delta_s = 0.001$

$$\Delta f = \frac{10F_s - 10F_p}{F_T} = \frac{2{,}000 - 1{,}500}{10{,}000} = 0.05$$

The filter lengths for $C(z)$ is calculated as

$$N = \frac{-20\log_{10}\sqrt{(0.001)(0.001)} - 13}{14.6\,\Delta f} = 65$$

For $A(z)$, $\delta_p = 0.001$, $\delta_s = 0.001$

$$\Delta f = \frac{800-150}{10,000} = \frac{650}{10,000} = 0.065$$

The filter length for $A(z)$ is calculated as

$$N = \frac{-20\ \log_{10}\sqrt{(0.001)(0.001)}-13}{14.6\,\Delta f} = 50$$

The length of the overall filter in cascade is

$$= 50 + (10 \times 65) + 2 = 702$$

The filter length in cascade realisation has increased but the number of multiplications per second (computerised complexity) can be reduced.

$$R_{M,C} = 65 \times \frac{1000}{2} = 32,500$$

$$R_{M,A} = 50 \times \frac{10000}{10} = 50,000$$

Total number of multiplications per second is

$$R_{M,C} + R_{M,A} = 32,500 + 50,000 = 82,500$$

Three-stage Realisation

The decimation filter $F(z)$ can be realised in the cascade form $R(z)\ S(z^5)$. The specifications are given as follows

For $S(z)$, $\delta_p = 0.0005$ and $\delta_s = 0.001$

$$\Delta f = \frac{(800-150)}{10,000} \times 5 = 0.325$$

Therefore, the filer length is $N = \dfrac{-20\ \log_{10}\sqrt{(0.0005)(0.001)}-13}{14.6 \times 0.325} = 11$

For $R(z)$, $\delta_p = 0.0005$ and $\delta_s = 0.001$

$$\Delta f = \frac{((800 \times 2)-150)}{10,000} = 0.145$$

Therefore, the filter length is $N = \dfrac{-20\ \log_{10}\sqrt{(0.0005)(0.001)}-13}{14.6 \times 0.145} = 24$

The three-stage realisation is shown in Fig. E11.11(c).

The number of multiplications per second is given by

$$R_{M,S} = 11 \times \frac{2,000}{2} = 11,000$$

$$R_{M,R} = 24 \times \frac{10,000}{5} = 48,000$$

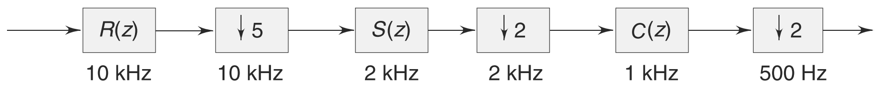


Fig. E11.11(c) *Frequency Response for a Three-stage Decimation*

The overall number of multiplications per second for a three-stage realisation is given by

$$R_{M,C} + R_{M,S} + R_{M,R} = 32,500 + 11,000 + 48,000 = 91,500$$

The number of multiplications per second for a three-stage realisation is more than that of a two-stage realisation. Hence, beyond two stages, the realisation may not be efficient.

Number of Stages and Decimation Factors

A multistage system requires reduced computation and less storage space. This saving depends on the number of stages and the decimation factor used at each stage. Optimal number of stages refer to less computational effort, e.g. number of Multiplications per Second (MPS) or Total Storage Requirements (TSR) for their coefficients

$$MPS = \sum_{i=1}^{l} N_i F_i$$

$$TSR = \sum_{i=1}^{l} N_i$$

where N_i is the number of filter coefficients for the stage *i*.

For practical purpose, the number of stages does not go beyond 3 or 4. Also, for a given M, there are limited sets of probable integer factors. Hence, with a selected set of M values, MPS and TSR can be determined.

Comb Filters

The impulse response of a comb filter (FIR filter) is given by

$$h(n) = \begin{cases} 1, & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases} \tag{11.49}$$

where N is the number of taps in the filter.

The frequency response of the filter is given by

$$H\left(e^{j\omega}\right) = \left\{\left(\sin\left(\omega N/2\right)\right) / \left(\sin\left(\omega/2\right)\right)\right\} e^{-j(N-1)\omega/2} \tag{11.50}$$

Figure 11.30 shows the comb filter for $N = 5$. The sampling frequency of the input signal is f_i and that of the output is $5 f_i$.

In the frequency response, at the following frequencies, the values are zero.

$$\omega = 2\pi k/N,\ k = 1, 2, ..., N-1$$

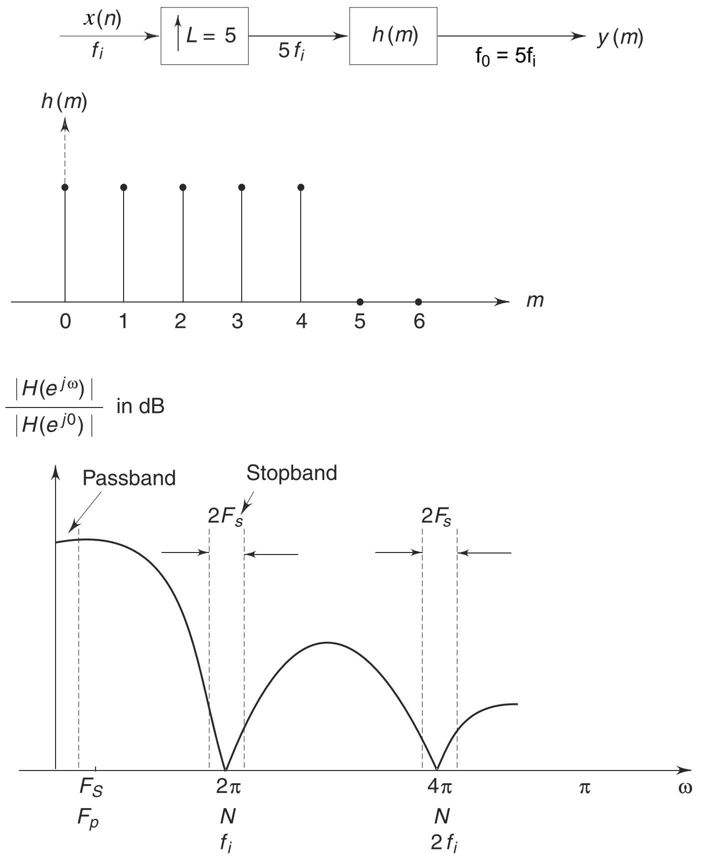


Fig. 11.30 *Comb Filter for N = 5*

Example 11.12 *Consider the multirate system shown in Fig. E11.12. Find y(n) as a function of x(n).*

Solution From Fig. E11.12, the outputs of the downsampler are

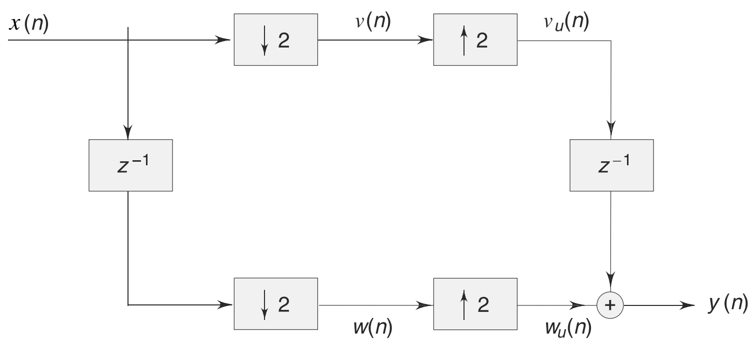


Fig. E11.12

$$V(Z) = \frac{1}{2}X\left(z^{\frac{1}{2}}\right) + \frac{1}{2}X\left(-z^{\frac{1}{2}}\right)$$

$$W(Z) = \frac{z^{\frac{-1}{2}}}{2}X\left(z^{\frac{1}{2}}\right) - \frac{z^{\frac{-1}{2}}}{2}X\left(-z^{\frac{1}{2}}\right)$$

The outputs of the upsampler are

$$V_u(z) = \frac{1}{2}X(z) + \frac{1}{2}X(-z)$$

$$W_u(z) = \frac{z^{-1}}{2}X(z) - \frac{z^{-1}}{2}X(-z)$$

$Y(z)$ is given by

$$Y(z) = z^{-1}V_u(z) + W_u(z)$$

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

Hence, $y(n) = x(n-1)$.

DIGITAL FILTER BANKS 11.9

Multirate filters give a useful idea of filter bank without filters. The complete operation is based only on the upsampling and downsampling functions. A filter bank without filters is shown in Fig. 11.31(a), in which both the analysis and synthesis sections are together.

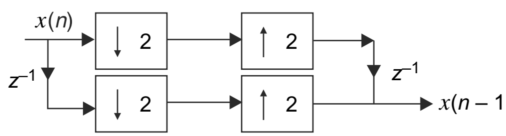


Fig. 11.31(a) *Filter Bank without Filters*

The top branch downsamples the analysis part and upsamples in the reconstruction part. The bottom branch also does the same operation in order but on the delayed version of the signal. The top branch acts as a low-pass filter and the bottom one acts as a high-pass filter. The reverse process is done for the reconstruction of the signal. Both the streams are upsampled but when adding together, a delay is introduced. In this structure, a genuine filter of definite taps before the downsampler on the analysis side and after the upsampler in the synthesis side is introduced as shown in Fig. 11.31(b). By appropriately choosing the filter coefficients, a very good decomposition can be achieved.

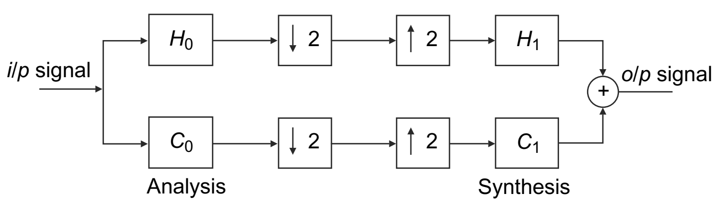


Fig. 11.31(b) *Typical Filter Bank*

Filter banks are used for performing spectrum analysis and signal synthesis. They are of two types, namely (i) analysis filter banks, and (ii) synthesis filter banks. An analysis filter bank consists of a set of filters with system function $H_k(z)$ arranged in a parallel bank shown in Fig. 11.32. The frequency response characteristic of this filter bank splits into a corresponding number of sub-bands.

The digital filter bank can be represented as a set of digital bandpass filters that can have either a common input or a summed output. The signal $x(n)$ can be decomposed into a set of M sub-band signals $v_k(n)$ with the M-band analysis filter bank. Each sub-band signal $v_k(n)$ occupies a portion of the original frequency band by filtering process done with the sub-filters $H_k(z)$ called the analysis filters. Similarly, the sub-band signals $\hat{v}_k(n)$ which belong to contiguous frequency bands can be combined to get $y(n)$ by means of the synthesis filter bank. This can be done by combining various synthesis filters with system function $A_k(z)$ as shown in Fig. 11.32(a) and (b).

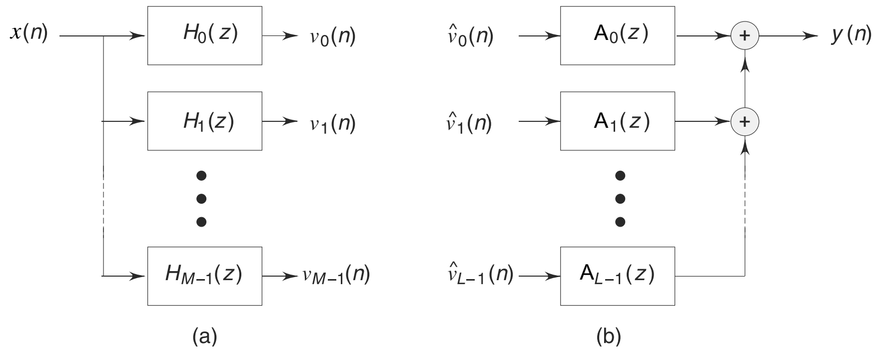


Fig. 11.32(a) *Analysis Filter Bank (b) Synthesis Filter Bank*

11.9.1 Uniform DFT Filter Banks

If a filter bank is used in the computation of the Discrete Fourier Transform (DFT) of a sequence $[x(n)]$, then the filter bank is called a DFT filter bank.

Consider the design of a class of filter banks with equal passbands. The causal low-pass IIR digital filter $H_0(z)$ is given by

$$H_0(z) = \sum_{n=0}^{\infty} h_0(n) z^{-n}$$

The passband edge is ω_p and the stopband edge is ω_s around π/M, where M is an integer. Let the transfer function $H_k(z)$ and the impulse response $h_k(n)$ be defined as

$$h_k(n) = h_0(n) W_M^{-kn}, \quad k = 0,1\ldots, M-1$$

where $W_M = e^{-j\frac{2\pi}{M}}$

$$H_k(z) = \sum_{n=0}^{\infty} h_k(n) z^{-n} = \sum_{n=0}^{\infty} h_0(n)\left(W_M^k z\right)^{-n}$$

$$= \sum_{n=0}^{\infty} h_0(n)\left(W_M^k z\right)^{-n}, \qquad k = 0,1,\ldots, M-1$$

$$= H_0\left(zW_M^k\right), \qquad k = 0,1,\ldots, M-1 \tag{11.51}$$

The frequency response is

$$H_k\left(e^{j\omega}\right) = H_0\left(e^{j\left(\omega - \frac{2\pi k}{M}\right)}\right), \qquad k = 0,1,\ldots, M-1$$

If the frequency response of $H_0(z)$ is shifted by $\dfrac{2\pi k}{M}$ to the right, the frequency response of $H_k(z)$ is obtained as shown in Fig. 11.33. If the impulse responses $h_k(n)$ are complex, then it is not necessary that $|H_k(e^{j\omega})|$ to exhibit symmetry with respect to zero frequency.

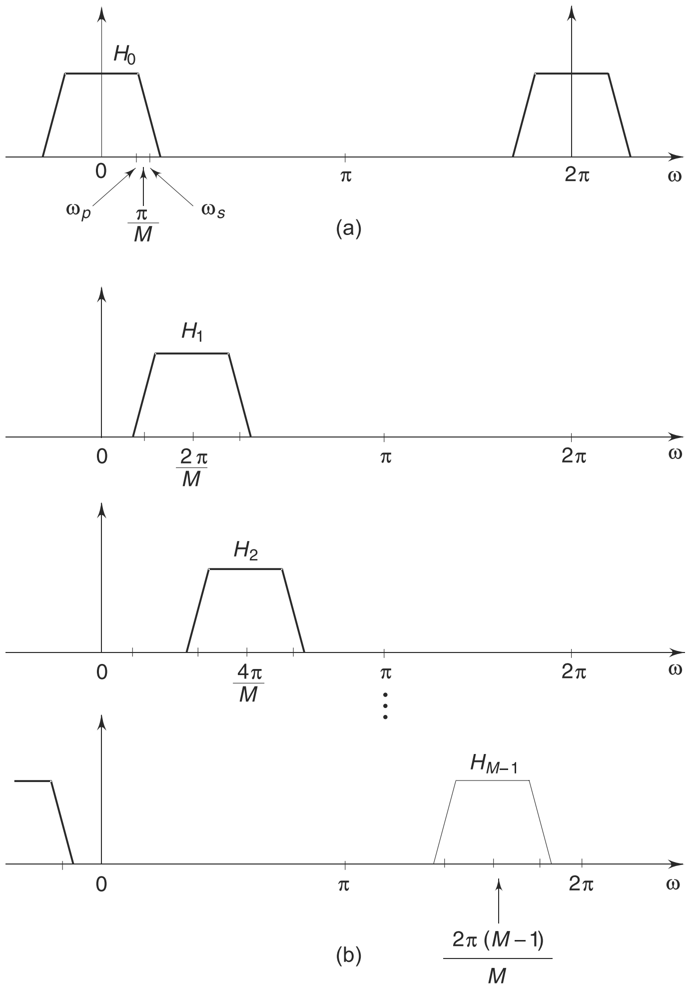


Fig. 11.33 *Frequency Responses for M-band Filter $H_k(z)$*

The M filters $H_k(z)$ is the uniformly shifted version of $H_0(z)$. The magnitude responses $|H_k(e^{j\omega})|$ for the M-band filters are uniformly shifted versions of $|H_0(e^{j\omega})|$. Hence, the filter bank is called a uniform filter bank. The M-band filter $H_k(z)$ can be used as analysis filters or synthesis filters.

Polyphase Implementation

The efficient realisation of the uniform filter bank can be developed using polyphase decomposition. The low-pass filter $H_0(z)$ can be expressed in polyphase form as

$$H_0(z) = \sum_{l=0}^{M-1} z^{-l} E_l\left(z^M\right)$$

where $E_l(z)$ is the polyphase component of $H_0(z)$.

$$E_l(z)=\sum_{n=0}^{\infty} e_l(n)z^{-n}=\sum_{n=0}^{\infty} h_0(l+nM)z^{-n},\quad 0\le l\le M-1$$

The M-band polyphase decomposition of $H_k(z)$ can be obtained by replacing z with zW_M^k. Therefore,

$$H_k(z)=\sum_{l=0}^{M-1}\left(zW_M^k\right)^{-l}E_l\left(\left(zW_M^k\right)^M\right)=\sum_{l=0}^{M-1}z^{-l}W_M^{-kl}E_l\left(z^M W_M^{kM}\right)$$

$$=\sum_{l=0}^{M-1}z^{-l}W_M^{-kl}E_l\left(z^M\right),\quad k=0,1,\ldots,M-1 \tag{11.52}$$

Since $W_M^{kM}=1$, $H_k(z)$ can be expressed in matrix form as

$$H_k(z)=\begin{bmatrix}1 & W_M^{-k} & W_M^{-2k} & \ldots & W_M^{-(M-1)k}\end{bmatrix}\begin{vmatrix}E_0\left(z^M\right)\\ z^{-1}E_1\left(z^M\right)\\ z^{-2}E_2\left(z^M\right)\\ .\\ .\\ .\\ z^{-(M-1)}E_{M-1}\left(z^M\right)\end{vmatrix}$$

where $k = 0, 1, ..., M - 1$.

$$\begin{bmatrix}H_0(z)\\ H_1(z)\\ H_2(z)\\ .\\ .\\ .\\ H_{(M-1)}(z)\end{bmatrix}=\begin{bmatrix}1 & 1 & 1 & \ldots & 1\\ 1 & W_M^{-1} & W_M^{-2} & \ldots & W_M^{-(M-1)}\\ 1 & W_M^{-2} & W_M^{-4} & \ldots & W_M^{-2(M-1)}\\ . & . & . & \ldots & .\\ . & . & . & \ldots & .\\ . & . & . & \ldots & .\\ 1 & W_M^{-(M-1)} & W_M^{-2(M-1)} & \ldots & W_M^{-(M-1)^2}\end{bmatrix}\begin{bmatrix}E_0\left(z^M\right)\\ z^{-1}E_1\left(z^M\right)\\ z^{-2}E_2\left(z^M\right)\\ .\\ .\\ .\\ z^{-(M-1)}E_{M-1}\left(z^M\right)\end{bmatrix}=M\mathbf{D}^{-1}\begin{bmatrix}E_0\left(z^M\right)\\ z^{-1}E_1\left(z^M\right)\\ z^{-2}E_2\left(z^M\right)\\ .\\ .\\ .\\ z^{-(M-1)}E_{M-1}\left(z^M\right)\end{bmatrix}$$

where **D** represents DFT matrix, which is given by

$$\mathbf{D}=\begin{bmatrix}1 & 1 & 1 & \ldots & 1\\ 1 & W_M^{1} & W_M^{2} & \ldots & W_M^{(M-1)}\\ 1 & W_M^{2} & W_M^{4} & \ldots & W_M^{2(M-1)}\\ . & . & . & \ldots & .\\ . & . & . & \ldots & .\\ . & . & . & \ldots & .\\ 1 & W_M^{(M-1)} & W_M^{2(M-1)} & \ldots & W_M^{(M-1)^2}\end{bmatrix}$$

The uniform DFT analysis filter bank structure is shown in Fig. 11.34.

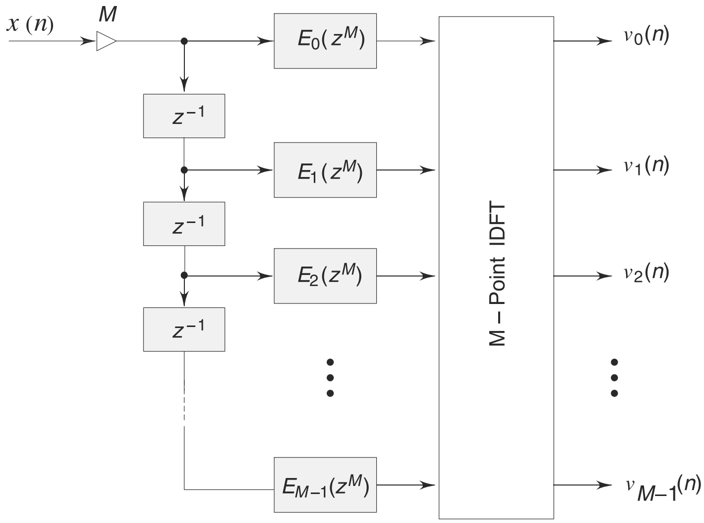


Fig. 11.34 *Uniform DFT Analysis Filter Bank Polyphase Implementation*

The direct implementation requires NM multiplications. The M-band uniform DFT analysis filter bank with N-tap low-pass filter requires $(M/2)\log_2 M + N$ multipliers.

Similarly, the structure for a uniform DFT synthesis filter bank using types I and II polyphase decomposition can be obtained as shown in Fig. 11.35(a) and (b).

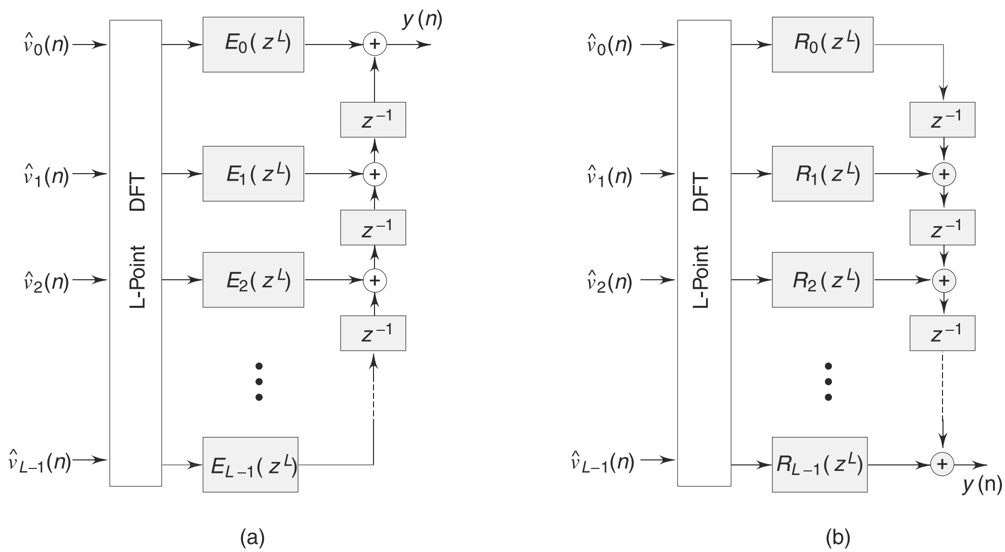


Fig. 11.35 *Uniform DFT Synthesis Filter Bank: (a) Type I Polyphase Decomposition (b) Type II Polyphase Decomposition*

11.9.2 Nyquist Filters

A type of low-pass filter which is computationally more efficient than other low-pass filters of the same order has certain zero-valued coefficients. When they are used as interpolator filters, the non-zero samples of the upsampler output are preserved. These type of filters are called L^{th} bank filters or Nyquist filters.

L^{th} Band Filters

The output of the interpolator with factor L is given by

$$Y(z) = H(z)\,X(z^L)$$

Polyphase Realisation

The interpolation filter $H(z)$ in polyphase form is given by

$$H(z) = E_0(z^1) + z^{-1}E_1(z^L) + z^{-2}E_2(z^L) + \ldots + z^{-(L-1)}E_{L-1}(z^L)$$

Let the k^{th} polyphase component $E_k(z)$ of $H(z)$ be a constant. Therefore

$$E_k(z) = \alpha$$

$$H(z) = E_0(z^L) + z^{-1}E_1(z^L) + \cdots + z^{-(k-1)}E_{k-1}(z^L) + \alpha z^{-k} + z^{-(k+1)}E_{k+1}(z^L) + \cdots + z^{-(L-1)}E_{L-1}(z^L)$$

Therefore,

$$Y(z) = \alpha z^{-k} X\left(z^L\right) + \sum_{\substack{l=0 \\ l\neq k}}^{L-1} z^{-1} E_l\left(Z^L\right) X\left(z^L\right) \tag{11.53}$$

In the time-domain, $y(Ln) = \alpha_x(n-k)$. The input samples will appear at the output without any distortion at $n = k,\ k \pm L,\ k \pm 2L, \ldots$. The in-between $(L-1)$ samples can be determined by interpolation.

Filters which satisfy the above specified property are called Nyquist filters or L^{th} band filters. The impulse response of the L^{th} band filter for $k = 0$ is given by

$$h(Ln) = \begin{cases} \alpha, & n = 0 \\ 0, & \text{otherwise} \end{cases}$$

The impulse response of a fourth band filter ($L = 4$) is shown in Fig. 11.36. If $k = 0$, $E_0(z) = \alpha$, then

$$\sum_{k=0}^{L-1} H\left(zW_L^k\right) = L\alpha = 1, \quad \text{if } \alpha = \frac{1}{L} \tag{11.54}$$

L^{th} band filters can be FIR or IIR filters, since the summed frequency response of $H(zW_L^k)$ of L uniformly shifted version of $H(e^{j\omega})$ is a constant. The frequency response of $H(zW_L^k)$ is shown in Fig. 11.37.

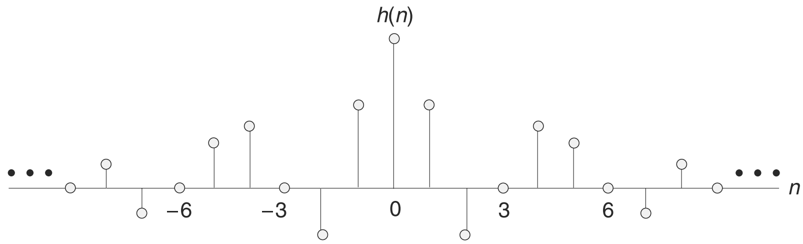


Fig. 11.36 *Impulse Response of a Fourth Band Filter*

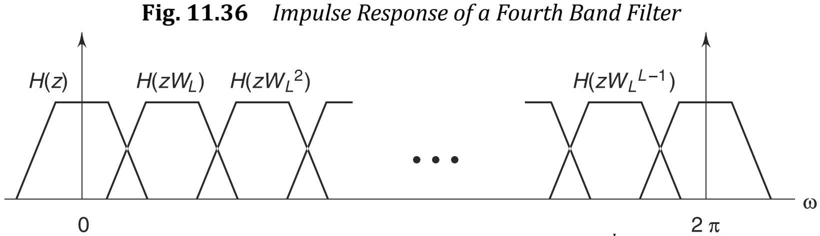


Fig. 11.37 *Frequency Response of $H(zW_L^k)$*

Linear Phase L^{th} Band FIR Filters

Consider a low-pass linear phase Nyquist L^{th} band FIR filter with a cut-off frequency $\omega_c = \frac{\pi}{L}$. Let us consider the windowing method for the filter design. The impulse response coefficients of the low-pass filter is

$$h(n) + h_{LP}(n)w(n)$$

where $h_{LP}(n)$ is the impulse response of an ideal low-pass filter and $w(n)$ is a suitable window function. If $h_{LP}(n) = 0$ for $n = \pm L, \pm 2L, \ldots$, then the property for L^{th} band filter is satisfied. The various choices are

(i) $$h_{Lp}(n) = \frac{\sin(\pi n / L)}{\pi n}, \quad -\infty \le n \le \infty \tag{11.55}$$

(ii) $$h_{Lp}(n) = \begin{cases} \dfrac{1}{L}, & n = 0 \\ \dfrac{2\sin(\Delta\omega n/2)}{\Delta\omega n}\dfrac{\sin(\pi n/L)}{\pi n}, & |n| > 0 \end{cases} \tag{11.56}$$

(iii) Raised cosine filter.

TWO-CHANNEL QUADRATURE MIRROR FILTER BANK 11.10

In applications where sub-band filtering is used, the processes are carried out in the following sequence.

(i) The signal $x(n)$ is split into a number of sub-band signals $\{v_k(n)\}$ by using an analysis filter bank.
(ii) The sub-band signals are processed.
(iii) The processed signals are combined using synthesis filter bank to obtain the output signal $y(n)$.

The sub-band signals are downsampled before processing. The signals are upsampled after processing. The structure is called Quadrature Mirror Filter (QMF) bank. In a critically sampled filter bank, the decimation and interpolation factors are equal and the characteristics of $x(n)$ will be available in $y(n)$, provided the filter structures are properly selected. The applications of QMF filters are

(i) Efficient coding of the signal, and
(ii) Analog voice privacy system in secure telephone communication.

The two-channel QMF filter bank is shown in Fig. 11.38. This is a multirate digital filter structure that employs two decimators in the signal-analysis part and two interpolators in the signal-synthesis part. The analysis filter $H_0(z)$ is a low-pass filter and $H_1(z)$ is a mirror image high-pass filter. In frequency domain, $H_0(\omega) = H(\omega)$ and $H_1(\omega) = H(\omega - \pi)$, where $H(\omega)$ is the frequency response of a low-pass filter. As a consequence, $H_0(\omega)$ and $H_1(\omega)$ have mirror image symmetry about the frequency $\omega = \dfrac{\pi}{2}$.

In time domain, the corresponding relations are

$$h_0(n) = h(n) \text{ and } h_1(n) = (-1)^n\, h(n).$$

The cut-off frequency is $\pi/2$ for these filters. The sub-band signals $\{v_k(n)\}$ are downsampled. After downsampling, these signals are processed (encoded). In the receiving side, the signals are decoded, upsampled and then passed through the synthesis filters $C_0(z)$ and $C_1(z)$ to get the output $y(n)$.

The low-pass filter $C_0(z)$ is selected such that $C_0(\omega) = 2H(\omega)$ and the high-pass filter $C_0(\omega)$ as $C_1(\omega) = -2H(\omega - \pi)$. In the time domain, $c_0(n) = -2h(n)$ and $c_1(n) = -2(-1)^n h(n)$. The scale factor of

2 corresponds to the interpolation factor used to normalise the overall frequency response of the QMF. Thus, the aliasing occurring due to decimation in the analysis section of the QMF bank is perfectly cancelled by the image signal spectrum due to interpolation. As a result, the two-channel QMF behaves as a linear, time-invariant system.

The encoding and decoding processes are not shown in Fig. 11.39. For perfect reconstruction, the QMF filter banks should be properly selected.

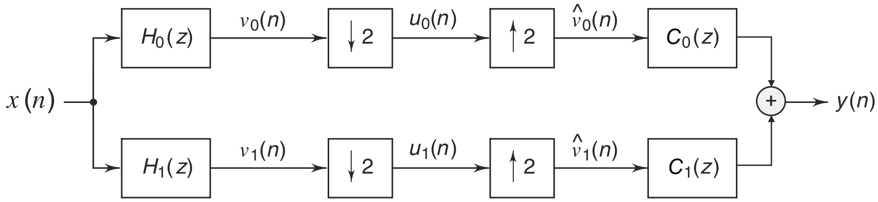


Fig. 11.38 *Two-channel Quadrature Mirror Filter Bank*

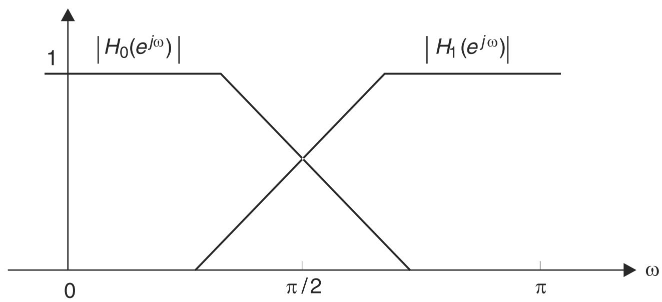


Fig. 11.39 *Frequency Response of Analysis Filters $H_0(z)$ and $H_1(z)$*

Analysis

The input-output relationships of the two-channel QMF filter bank are given by

$$V_k(z) = H_k(z)X(z)$$

$$U_k(z) = \frac{1}{2}\left\{V_k\left(z^{\frac{1}{2}}\right) + V_k\left(-z^{\frac{1}{2}}\right)\right\} \tag{11.57}$$

$$\hat{V}_k(z) = U_k\left(z^2\right) \quad \text{for} \quad k = 0,1.$$

Combining the above equations,

$$\hat{V}(z) = \frac{1}{2}\{V_k(z) + V_k(-z)\} = \frac{1}{2}\{H_k(z)X(z) + H_k(-z)X(-z)\} \tag{11.58}$$

The output is expressed as

$$Y(z) = C_0(z)\hat{V}_0(z) + C_1(z)\hat{V}_1(z) \tag{11.59}$$

$$= \frac{1}{2}\{H_0(z)C_0(z)X(z) + H_0(-z)X(-z)C_0(z)\}$$

$$+ \frac{1}{2}\{H_1(z)C_1(z)X(z) + H_1(-z)X(-z)C_1(z)\}$$

$$= \frac{1}{2}\{H_0(z)C_0(z) + H_1(z)C_1(z)\}X(z) + \frac{1}{2}\{H_0(-z)C_0(z) + H_1(-z)C_1(z)\}X(-z)$$

$$Y(z) = T(z)X(z) + B(z)X(-z) \tag{11.60}$$

where $T(z)$ is the distortion transfer function given by

$$T(z)=\frac{1}{2}\{H_0(z)C_0(z)+H_1(z)C_1(z)\} \tag{11.61}$$

and

$$B(z)=\frac{1}{2}\{H_0(-z)C_0(z)+H_1(-z)C_1(-z)\} \tag{11.62}$$

In matrix form,

$$Y(z)=\frac{1}{2}\left[X(z)X(-z)\right]\begin{bmatrix} H_0(z) & H_1(z) \\ H_0(-z) & H_1(-z)\end{bmatrix}\begin{bmatrix} C_0(z) \\ C_1(z)\end{bmatrix} \tag{11.63}$$

The matrix

$$H(z)=\begin{bmatrix} H_0(z) & H_1(z) \\ H_0(-z) & H_1(-z)\end{bmatrix} \tag{11.64}$$

is referred as the aliasing component (AC) matrix.

Errors in QMF Filter Bank

The QMF filter structure is a linear time-varying system since the upsampler and downsampler are linear time varying. The analysis and synthesis filters banks should be properly selected to cancel the aliasing effect because of linear time invariant operation. To achieve this, we have

$$2B(z) = H_0(-z)C_0(z) + H_1(-z)C_1(z) = 0 \tag{11.65}$$

One possible solution for this is,

$$C_0(z) = H_1(-z),\ \ C_1(z) = -H_0(-z) \tag{11.66}$$

Then the output becomes

$$Y(z) = T(z)X(z) \tag{11.67}$$

where

$$T(z)=\frac{1}{2}\{H_0(z)H_1(-z)-H_1(z)H_0(-z)\} \tag{11.68}$$

If analysed over a unit circle,

$$\begin{aligned} Y(e^{j\omega}) &= T(e^{j\omega})X(e^{j\omega}) \\ &= |T(e^{j\omega})|e^{j\phi(\omega)}X(e^{j\omega}) \end{aligned} \tag{11.69}$$

Let $T(z)$ be an all-pass filter, i.e., $|T(e^{j\omega}) = d \neq 0$ where d is a constant. Therefore,

$$|Y(e^{j\omega})| = d\,|X(e^{j\omega})| \tag{11.70}$$

Let $T(z)$ have linear phase

$$\phi(\omega) = \alpha\omega + \beta \tag{11.71}$$

Then

$$\angle Y(e^{j\omega}) = \angle X(e^{j\omega}) + \alpha\omega + \beta \tag{11.72}$$

Hence, the filter bank is said to be both magnitude and phase preserving. If the QMF filter bank has no distortion in amplitude and phase, then the QMF filter bank is called perfect reconstruction QMF filter bank. Then

$$T(z) = dz^{-n_0}$$

Therefore,

$$Y(z) = dz^{-n_0}X(z) \tag{11.73}$$

In the time-domain,

$$y(n) = dx(n - n_0)$$

The output signal $y(n)$ is the scaled and delayed signal of $x(n)$.

11.10.1 Alias Free Realisation

To achieve alias free QMF filter bank, we have

$$H_1(z) = H_0(-z) \tag{11.74}$$

Frequency responses are,

$$\left|H_1\left(e^{j\omega}\right)\right| = \left|H_0\left(e^{j(\pi-\omega)}\right)\right| \tag{11.75}$$

where $H_0(z)$ is a low-pass filter and $H_1(z)$ is a high-pass filter.

$$C_0(z) = H_0(z) \tag{11.76}$$

$$C_1(z) = -H_1(z) = -H_0(-z) \tag{11.77}$$

where $G_0(z)$ is a low-pass filter and $G_1(z)$ is a high-pass filter. Therefore,

$$T(z) = \frac{1}{2}\left\{H_0^2(z) - H_1^2(z)\right\} = \frac{1}{2}\left\{H_0^2(z) - H_0^2(-z)\right\} \tag{11.78}$$

Polyphase Representation

Using Type I polyphase realisation,
For analysis filters,

$$H_0(z) = E_0(z^2) + z^{-1}E_1(z^2) \tag{11.79}$$

$$H_1(z) = E_0(z^2) - z^{-1}E_1(z^2) \tag{11.80}$$

In matrix form,

$$\begin{bmatrix} H_0(z) \\ H_1(z) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} E_0\left(z^2\right) \\ z^{-1}E_1\left(z^2\right) \end{bmatrix} \tag{11.81}$$

For synthesis filters,

$$\begin{bmatrix} C_0(z) & C_1(z) \end{bmatrix} = \begin{bmatrix} z^{-1}E_1\left(z^2\right) & E_0\left(z^2\right) \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{11.82}$$

The distortion transfer function in terms of polyphase components is given by

$$T(z) = 2z^{-1}E_0(z^2)E_1(z^2) \tag{11.83}$$

Figure 11.40 shows the polyphase realisation of the two-channel QMF filter bank.

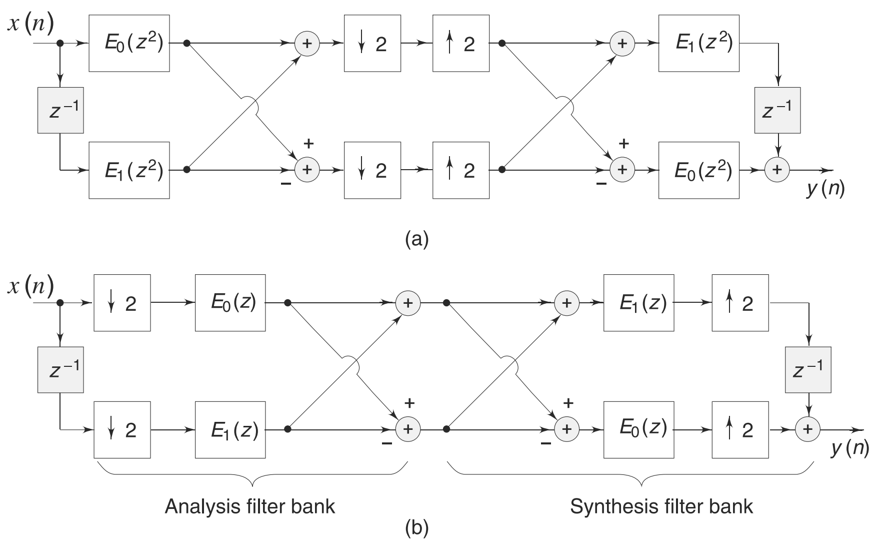


Fig. 11.40 *Polyphase Realisation of Two-channel QMF Filter Bank (a) Direct Realisation and (b) Computationally Efficient Realisation*

11.10.2 Alias-Free QMF Filter Bank

In the linear-phase FIR filter, the transfer function $|T(e^{j\omega})|$ should be a constant for obtaining perfect reconstruction without any phase disortion. But in the practical case, $|T(e^{j\omega})|$ is not a constant and hence, the filter bank introduces amplitude distortion.

For the prototype low-pass filter, the transfer function is given by

$$H_0(z) = \sum_{n=0}^{N-1} h_0(n) z^{-n}$$

The impulse response should satisfy the condition

$$h_0(n) = h_0(N-1-n)$$

Hence, $H_0(e^{j\omega}) = e^{-j\omega(N-1)/2} \breve{H}_0(\omega)$ (11.84)

where $\breve{H}_0(\omega)$ is the amplitude function, which is a real function of ω. The distortion transfer function becomes,

$$T(e^{j\omega}) = \frac{e^{-j(N-1)\omega}}{2}\left\{\left|H_0(e^{j\omega})\right|^2 - (-1)^{N-1}\left|H_0(e^{j(\pi-\omega)})\right|^2\right\} \quad (11.85)$$

If N is odd, then $T(e^{j\omega}) = 0$, at $\omega = \pi/2$.

This results in severe amplitude distortion at the output. Therefore, N is usually taken to be even. Then,

$$T(e^{j\omega}) = \frac{e^{-j(N-1)\omega}}{2}\left\{\left|H_0(e^{j\omega})^2\right| + \left|H_0(e^{j(\pi-\omega)})\right|^2\right\}$$

$$= \frac{e^{-j(N-1)\omega}}{2}\left\{\left|H_0(e^{j\omega})\right|^2 + \left|H_0(e^{j(\pi-\omega)})^2\right|\right\} \quad (11.86)$$

To reduce amplitude distortion, the following filter characteristics should be met

(i) $H_0(z)$, the low-pass filter should have
$|H_0(e^{j\omega})| \approx 1$ in the passband
$|H_0(e^{j\omega})| \approx 0$ in the stopband

(ii) $H_1(z)$, the high-pass filter should have its passband coinciding with the stopband of $H_0(z)$ and vice versa.

(iii) $|T(e^{j\omega})| \approx 1/2$ in the passbands of $H_0(z)$ and $H_1(z)$. Amplitude distortion occurs in the transition band which can be reduced by properly selecting the passband edge of $H_0(z)$.

Another method for reducing the amplitude distortion is by properly selecting the coefficients $h_0(n)$.

IIR Filter Bank

For the alias-free two-channel QMF bank IIR filters, the distortion transfer function is given by

$$T(z) = 2z^{-1}E_0(z^2)E_1(z^2) \quad (11.87)$$

If $T(z)$ represents an all-pass filter, the magnitude response is constant.

Analysis Filters

The polyphase components of $H_0(z)$ are given by

$$E_0(z) = \frac{1}{2}B_0(z); \quad E_1(z) = \frac{1}{2}B_1(z)$$

where $B_0(z)$ and $B_1(z)$ represent the stable functions of all-pass filters.

$$H_0(z)=\frac{1}{2}\left\{B_0\left(z^2\right)+z^{-1}B_1\left(z^2\right)\right\}$$

$$H_1(z)=\frac{1}{2}\left\{B_0\left(z^2\right)-z^{-1}B_1\left(z^2\right)\right\}$$

In matrix form,

$$\begin{bmatrix}H_0(z)\\H_1(z)\end{bmatrix}=\frac{1}{2}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}B_0\left(z^2\right)\\z^{-1}B_1\left(z^2\right)\end{bmatrix} \tag{11.88}$$

Synthesis Filters

$$\begin{bmatrix}C_0(z) & C_1(z)\end{bmatrix}=\frac{1}{2}\begin{bmatrix}z^{-1}B_1\left(z^2\right) & B_0\left(z^2\right)\end{bmatrix}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix} \tag{11.89}$$

$$C_0(z)=\frac{1}{2}\left\{B_0\left(z^2\right)+z^{-1}B_1\left(z^2\right)\right\}=H_0(z)$$

$$C_1(z)=\frac{1}{2}\left\{-B_0\left(z^2\right)+z^{-1}B_1\left(z^2\right)\right\}=-H_1(z)$$

The realisation of the above filter is shown in Fig. 11.41.

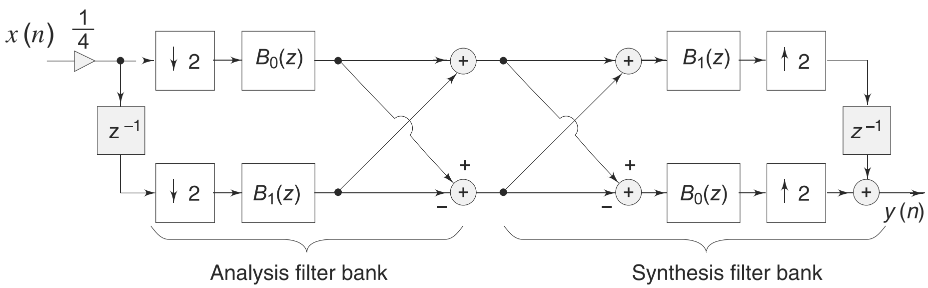


Fig. 11.41 *Magnitude Preserving Two-channel QMF Filter Bank*

11.10.3 *L*-Channel QMF Filter Bank

Figure 11.42 shows the *L*-channel QMF filter bank structure. The filter equations are given by

$$V_k(z)=H_k(z)X(z) \tag{11.90}$$

$$U_k(z)=\frac{1}{L}\sum H_k\left(z^{1/L}W_L^l\right)X\left(z^{1/L}W_L^l\right)$$

$$\hat{V}_k(z)=U_k\left(z^L\right),\quad k=0,1,\ldots L-1$$

The output $Y(z)$ is

$$Y(z)=\sum_{k=0}^{L-1}C_k(z)\hat{V}_k(z) \tag{11.91}$$

Combining the above equations,

$$Y(z)=\frac{1}{L}\sum_{l=0}^{L-1}X\left(zW_L^l\right)\sum_{k=0}^{L-1}H_k\left(zW_L^l\right)C_k(z) \tag{11.92}$$

$$= \sum_{l=0}^{L-1} a_l(z) X\left(zW_L^l\right)$$

where $$a_l(z) = \frac{1}{L}\sum_{k=0}^{L-1} H_k\left(zW_L^l\right) C_k(z), 0 \le l \le L-1$$

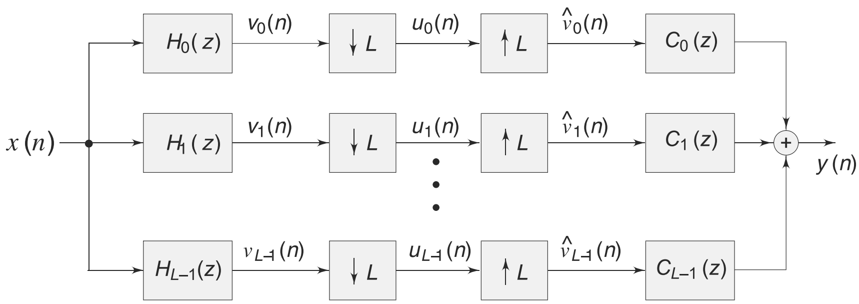


Fig. 11.42 *L-channel QMF Filter Bank*

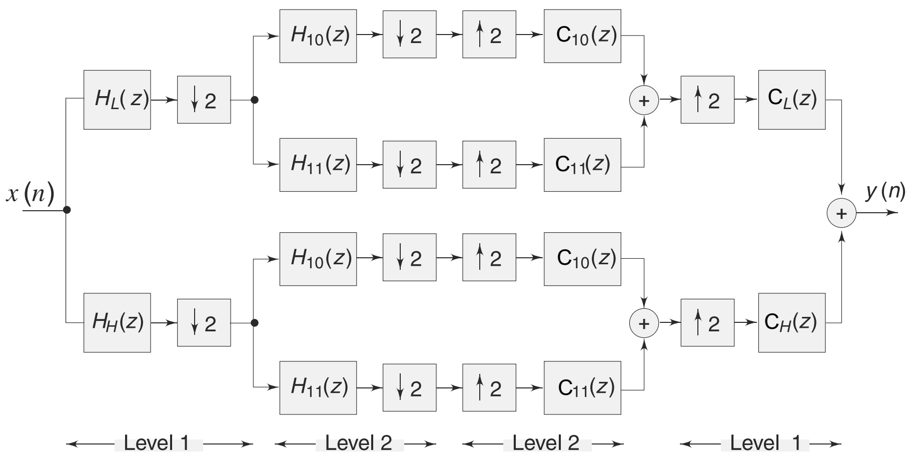
MULTILEVEL FILTER BANKS 11.11

Multiband analysis and sythesis filter bank can be obtained by combining many two-channel QMF filter banks. If the individual QMF filter bank satisfies the perfect reconstruction property, then the entire structure exhibits the same property.

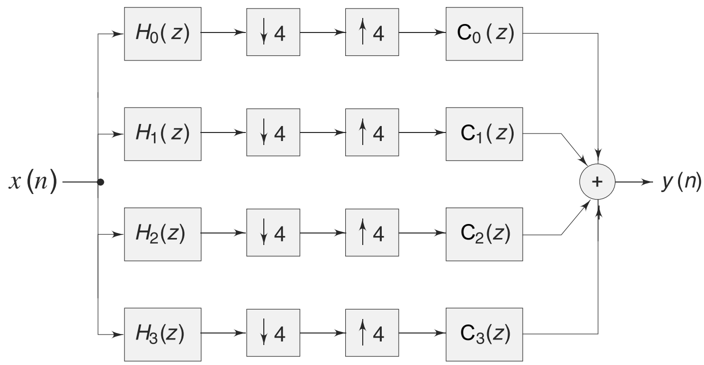
11.11.1 Filter Banks with Equal Passbands

If two-channel QMF filter bank is inserted between the downsampler and upsampler of another two-channel QMF filter, then the resultant is four-channel QMF filter. Because of its tree structure, it is also called as tree structured filter bank. Figure 11.43(a) and (b) show the four-channel QMF filter bank. The relationships between the two-level filter and its equivalent structure is given by

$$H_0(z) = H_L(z)H_{10}(z^2) \qquad C_0(z) = C_L(z)C_{10}(z^2)$$
$$H_1(z) = H_L(z)H_{11}(z^2) \qquad C_1(z) = C_L(z)C_{11}(z^2)$$
$$H_2(z) = H_H(z)H_{10}(z^2) \qquad C_2(z) = C_H(z)C_{10}(z^2)$$
$$H_3(z) = H_H(z)H_{11}(z^2) \qquad C_3(z) = C_H(z)C_{11}(z^2)$$



(a)



(b)

Fig. 11.43(a) *Two-level Four-channel QMF Filter Bank (b) Equivalent Representation*

The analysis filter $H_k(z)$ which are in cascade has the following characteristics:

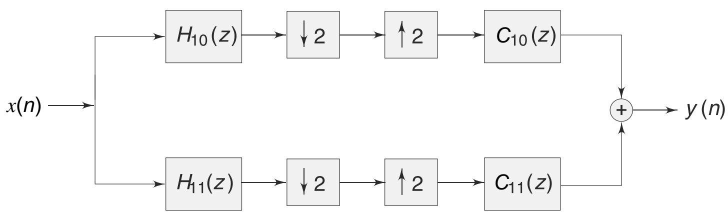
(i) One filter has a single passband and single stopband

(ii) Another filter has two passbands and two stopbands.

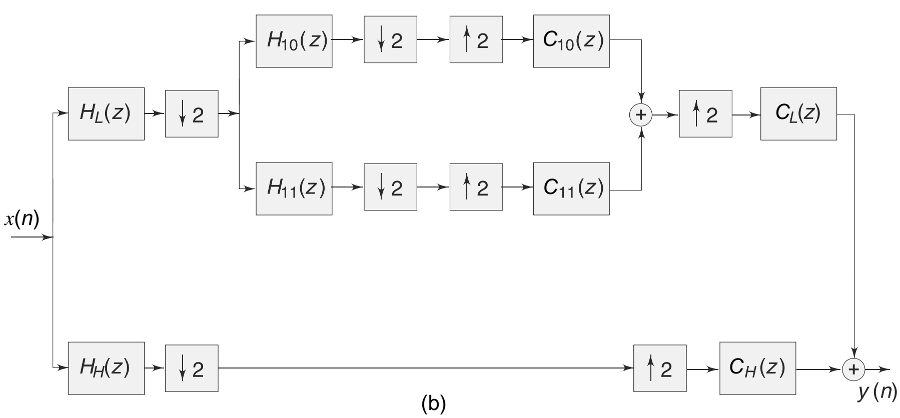
Similarly, the multilevel QMF filter can be arrived with passbands of equal widths given by π/L.

11.11.2 Filter Banks with Unequal Passbands

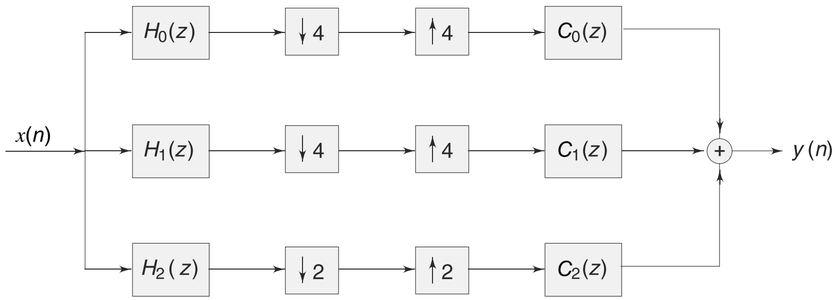
If a two-channel QMF filter bank is introduced in one sub-band channel of another two-channel QMF filter bank, the resultant structure has filters with unequal passband. Figure 11.44(a), (b) and (c) show the three-channel QMF filter bank derived from a two-channel QMF filter bank. The relationship



(a)



(b)



(c)

Fig. 11.44 *(a) Two-channel QMF Filter Bank (b) Three-channel QMF Filter Bank (c) Equivalent Three-channel QMF Filter Bank Realisation*

between the three-channel QMF filter bank and its equivalent structure is given by

$$H_0(z) = H_L(z)H_L(z^2) \qquad C_0(z) = C_L(z)C_L(z^2)$$
$$H_1(z) = H_L(z)H_H(z^2) \qquad C_1(z) = C_L(z)C_H(z^2)$$
$$H_2(z) = H_H(z) \qquad C_2(z) = C_H(z)$$

Similarly, a four-channel QMF filter bank can be derived from a three-channel QMF filter bank. These structures come under the class of non-uniform QMF filter banks. These types of filters find applications in speech and image coding.

REVIEW QUESTIONS

11.1 What is multirate DSP?
11.2 What is the need for multirate signal processing?
11.3 Give some examples of multirate systems.
11.4 List some applications of multirate signal-processing system.
11.5 List the advantages of multirate processing.
11.6 What is a decimator? Draw the symbolic representation of a decimator.
11.7 Show that the decimator is a time-variant system.
11.8 Show that the decimator is a linear system.
11.9 What is anti-aliasing filter?
11.10 What is an interpolator? Draw the symbolic representation of an interpolator.
11.11 Show that the interpolator is a time-variant system.
11.12 Show that the interpolator is a linear system.
11.13 What is an anti-imaging filter?
11.14 Explain the decimation process with an example.
11.15 Explain the interpolation process with an example.
11.16 Write the input-output relationship for the decimation by a factor of five.
11.17 Explain the sampling process with an example.
11.18 What is meant by aliasing?
11.19 How can aliasing be avoided?

11.20 The signal $x(n)$ is defined by $x(n) = \begin{cases} a^n, & n > 0 \\ 0, & \text{otherwise} \end{cases}$

(a) Obtain the decimated signal with a factor of three.

(b) Obtain the interpolated signal with a factor of three.

11.21 Explain polyphase decomposition process.

11.22 How can sampling rate be converted by a rational factor M/L?

11.23 Draw the block diagram of a multistage decimator and integrator.

11.24 Discuss the computationally efficient implementation of decimator in an FIR filter.

11.25 Discuss the computationally efficient implementation of interpolator in an FIR filter.

11.26 What are the characteristics of a comb filter?

11.27 Explain with block diagram the general polyphase framework for decimators and interpolators.

11.28 What is a signal-flow graph?

11.29 Draw the signal-flow graph of a system represented by the input-output relationship
$y(n) = 3x(n) + 5x(n-1) - 4x(n-2) - 6y(n-2)$

11.30 Design a set of interpolators using equiripple design method for the following specifications: $L = 5$, $\delta_p \le 0.002$, $\delta_s \le 0.0002$, $0 \le \omega_c \le \pi$ and $\omega_p = \omega_c / L$

11.31 Design a decimator with the following specifications:

$M = 5$, $\delta_p = 0.0035$, $\omega_s = 0.2\pi$ and $\Delta f = (\omega_s - \omega_p)/2\pi$

11.32 What is the need for multistage filter implementation?

11.33 What are the drawbacks in multistage filter implementations?

11.34 Design a 4-stage decimator where the sampling rate has to be reduced from 20 kHz to 500 Hz. The specification for the decimator filter $H(z)$ are as follows:

Passband edge = 20 Hz, Stopband edge = 220 Hz, Passband ripple = 0.004, Stopband ripple = 0.002. Determine the filter lengths and the number of multiplications per second.

11.35 Obtain the polyphase structure of the IIR filter with the transfer functions given by

(a) $H(z) = \dfrac{1 - 3z^{-1}}{1 + 4z^{-1}}$ (b) $H(z) = \dfrac{1 - 2z^{-1}}{1 + 3z^{-1}}$

11.36 What are the advantages of polyphase decomposition?

11.37 Explain the polyphase decomposition for

(a) FIR filter structure (b) IIR filter structure

11.38 The analysis filter of a two-channel QMF filter bank is given by $H_0(z) = 1 + z^{-1}$. Give the expression for the distortion transfer function.

11.39 What are digital filter banks? Give some applications where these filter banks are used.

11.40 What are uniform DFT filter banks? Explain in detail.

11.41 Give the polyphase implementation of uniform filter banks.

11.42 What are Nyquist filters?

11.43 Give the input-output relationship of L^{th} band filter.

11.44 Explain the design of linear phase L^{th} band FIR filters.

11.45 What are quadrature mirror filter banks?

11.46 Give the input-output analysis for a two-channel QMF filter band.

11.47 What are the errors in QMF filter bank?

11.48 Explain how alias-free QMF realisation is achieved.

11.49 Explain how alias-free FIR QMF filter bank and (ii) IIR QMF filter bank realisation is achieved.

11.50 Design a five-channel filter bank.

11.51 What are multilevel filter banks?

11.52 Explain filter banks with equal and non-equal filter passbands.

11.53 Prove that the transpose of M-channel analysis filter bank is an M-channel synthesis bank.

11.54 Obtain a two-band polyphase decomposition of the following IIR transfer functions.

$$H(z) = \frac{a_0 + a_1 z^{-1}}{1 + b_1 z^{-1}}, |\ b_i\ | < 1$$

11.55 Derive the input-output relation for an uniform DFT synthesis filter bank with

(a) Type-I decomposition (b) Type-II decomposition

11.56 Decompose $C(z) = \dfrac{\left(1+z^{-1}\right)^3}{6+2z^{-2}}$ in the form $C(z) = \dfrac{1}{2}\left\{B_0(z) + B_1(z)\right\}$, where $B_0(z)$ and $B_1(z)$ are stable all-pass transfer functions.

11.57 Derive the four-channel QMF filter bank from three-channel QMF filter bank with an unequal passband.

11.58 Write a short note on sampling-rate conversion by a rational factor.

11.59 What is polyphase decomposition?

11.60 Derive an expression for the spectrum of output signal of an interpolator.

11.61 Discuss the concept of imaging in the spectrum of output signal of an interpolator with an example.

11.62 Discuss the multistage implementation of sampling rate conversion.

11.63 Write a detailed note on polyphase decomposition of filters.

11.64 Draw and explain the polyphase structure of a decimator.

11.65 Draw and explain the polyphase structure of an interpolator.

11.66 Explain the digital filter banks with suitable sketches.

11.67 Discuss the sub-band coding of speech signal with a suitable diagram.

Chapter 12

Discrete-Time Random Signal Processing

INTRODUCTION 12.1

The deterministic signals are classified as weighted sum of complex exponentials and are highly predictable. Practically, signals that are encountered are not exactly predictable in nature. Such signals are not deterministic and are termed as *random signals* or *noise* . However, using concepts from probability theory, such noise signals can also be analyzed. Many important practical signals like music and speech can be characterised using concepts from probability and statistics.

In this chapter, the fundamentals of random processes and their statistical properties are discussed. The concepts of various statistical averages like mean, variance, autocorrelation etc. are considered. The notion of ergodic processes, stationary processes and power spectrum of a random process is also explored.

RANDOM VARIABLES 12.2

The basic characteristics of a random variable are defined which serve as the starting point for the development of random processes.

In a random experiment, the outcome cannot be predicted with certainty, even though all the possible outcomes are known in advance. The set of all possible outcomes of a random experiment is known as the sample space and denoted by S.

Consider an example of tossing an unbiased coin. The outcome of the experiment can be either head (H) or tail (T). Thus,the sample space can be written as

$$S = \{H, T\}$$

A random variable is defined as a function or a mapping which maps every possible outcome of a random experiment to a real number.

In the above example, define the random variable X such that $X(H) = 1$ and $X(T) = -1$. Hence, in this case, X takes only two values 1 and -1. The corresponding probability will be

$$P\{X = 1\} = 1/2 \text{ and } P\{X = -1\} = 1/2$$

In general, a random variable follows a rule that assigns a real number to all the outcomes of a random experiment as shown in Fig. 12.1.

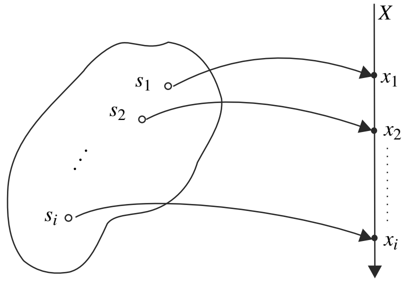


Fig. 12.1 *Mapping of Outcomes in a Sample Space to Real Numbers Using Random Variables*

Here, S is the sample space which contains all possible outcomes namely $s_1, s_2, ..., s_n$ and X is the random variable which takes the values $x_1, x_2, ..., x_n$ for each outcome. If the random variable takes countably finite number of values, then it is called discrete random variable. On the other hand, if the random variable takes all values in an interval, then it is called continuous random variable.

In signal processing applications, it is important to describe the random variable with its probabilistic description. The probability distribution function of a random variable X is defined and denoted by $F_X(x)$ as

$$F_X(x) = P[X \le x] \tag{12.1}$$

For example, for the coin-tossing experiment, the probability distribution function is defined as

$$F_X(x) = \begin{cases} 0 & ; \quad x < -1 \\ 1/2 & ; \quad -1 \le x < 1 \\ 1 & ; \quad x \ge 1 \end{cases} \tag{12.2}$$

Another characterization for a discrete random variable is the probability mass function (pmf) $P[X = x_i]$ and probability density function $f_X(x)$ for a continuous random variable. They are defined as

$$P[X = x_i] = F_X(x_i) - F_X(x_{i-1})$$

$$f_X(x) = \frac{d}{dx} F_X(x) \tag{12.3}$$

The probability distribution function and probability mass function of a coin-tossing experiment are plotted in Fig. 12.2(a) and (b) respectively.

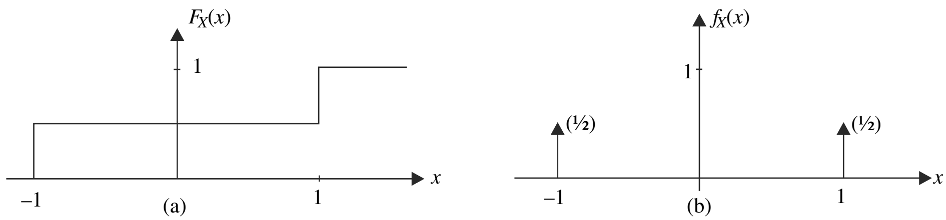


Fig. 12.2 *(a) Probability Distribution Function, (b) Probability Density Function*

ENSEMBLE AVERAGES OF A RANDOM VARIABLE 12.3

In many applications, the complete statistical characterisation using probability distribution function and probability density function may not be necessary. In such cases, the average behaviours of the random variable may be sufficient. Thus, the expected value of a random variable is defined.

If X is a continuous random variable, with probability density function $f_X(x)$, then the expected value of the random variable X is defined as

$$E[X] = \int_{-\infty}^{\infty} x\, f_X(x)\, dx \tag{12.4}$$

which is called the mean (μ) of the random variable.

Similarly, if $Y = g(X)$, then the expected value of Y is given by

$$E[Y] = \int_{-\infty}^{\infty} g(X) f_X(x)\, dx \tag{12.5}$$

If $Y = X^2$, then

$$E\left[X^2\right] = \int_{-\infty}^{\infty} x^2 f_X(x)\, dx \tag{12.6}$$

which is called the mean square value of the random variable.

The variance (σ^2) of a random variable is defined as

$$\text{var}(X) = \sigma^2 = E\lfloor (X-\mu)^2 \rfloor \tag{12.7}$$

$$= \int_{-\infty}^{\infty} (x-\mu)^2 f_X(x)\,dx$$

$$\sigma^2 = E[X^2] - \{E[X]\}^2 \tag{12.8}$$

TWO-DIMENSIONAL RANDOM VARIABLES 12.4

If there are two or more continuos random variables, it is necessary to consider the statistical dependencies between them. These relationships can be defined by joint probability distribution and density functions. If X_1 and X_2 are two random variables, their joint distribution function is defined as

$$F_{X_1X_2}(x_1, x_2) = P\{X_1 \le x_1, X_2 \le x_2\} \tag{12.9}$$

The joint density function is defined as

$$f_{X_1,X_2}(x_1,x_2) = \frac{\partial^2}{\partial x_1 \partial x_2}\left\{F_{X_1,X_2}(x_1,x_2)\right\} \tag{12.10}$$

Just as in the single random variable, the ensemble averages for two random variables X_1 and X_2 are defined. If X_1 and X_2 are two random variables with probability density function $f_{X_1X_2}(x_1, x_2)$, the expectations of X_1 and X_2 are defined as

$$E[X_1] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1\, f_{X_1X_2}(x_1,x_2)\,dx_1\,dx_2 \tag{12.11}$$

and

$$E[X_2] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_2\, f_{X_1X_2}(x_1,x_2)\,dx_1\,dx_2 \tag{12.12}$$

The covariance of X_1 and X_2 is defined as

$$\text{cov}(X_1, X_2) = E\{[X_1 - E(X_1)]\,[X_2 - E(X_2)]\}$$

$$\text{cov}(X_1, X_2) = E[X_1 X_2] - E[X_1]\,E[X_2] \tag{12.13}$$

The coefficient of correlation is given by

$$\rho_{x_1x_2} = \frac{cov(X_1, X_2)}{\sigma_{X1}\sigma_{X2}} \tag{12.14}$$

where $\sigma_{X1} = \sqrt{\sigma_{X1}^2} = \sqrt{var(X_1)}$

and $\sigma_{X2} = \sqrt{var(X_2)}$

Two random variables are said to be independent if its joint probability function is separable.

$$f_{X_1X_2}(x_1, x_2) = f_{X_1}(x_1)\,f_{X_2}(x_2) \tag{12.15}$$

BIAS AND CONSISTENCY FOR PARAMETER ESTIMATION 12.5

In signal processing applications, there are many examples in which the value of an unknown parameter has to be estimated from a set of observations of a random variable. One such example is estimating the value of mean and variance from a set of observations that follow Gaussian distribution. The estimated values will be the function of the set of observations and they themselves will be random variables.

It is important to characterise the effectiveness of the estimated values. Thus, the effectiveness of any estimator is described using *bias* and *consistency* which are defined below.

Bias Let θ be a parameter which is to be estimated from a set of random variables x_n for $n = 1, 2, \ldots N$. The estimated value is denoted by $\hat{\theta}$. Now, bias denoted by B, is defined as the difference between the expected value of the estimate and the actual value

$$B = \theta - E(\hat{\theta})$$

(i) If $B = 0$, then the estimate is said to be unbiased.
(ii) If $B \neq 0$, then the estimate is said to be biased.
(iii) If an estimate is biased and it goes to zero from infinite number of observations, then the bias is said to be asymptotically unbiased.

$$\underset{N\to\infty}{Lt}\ E\left(\hat{\theta}\right) = \theta$$

It is desirable that any estimate should be unbiased or asymptotically unbiased.

In order for the estimate of a parameter to converge to its true value, it is necessary that the variance of the estimate should go to zero for infinite number of observations.

$$\underset{N\to\infty}{Lt}\ \mathrm{var}\left(\hat{\theta}\right) = \underset{N\to\infty}{Lt}\ E\left\{\left|\hat{\theta} - E\left[\hat{\theta}\right]\right|\right\} = 0$$

Another form of convergence is given by mean square convergence of the estimator as,

$$\underset{N\to\infty}{Lt}\ E\left\{\left|\hat{\theta} - \theta\right|^2\right\} = 0$$

However, for an unbiased estimator, both the conditions are equivalent. Finally, an estimate is said to be consistent if it converges to the true value in some sense. We say that an estimate is consistent if it is asymptotically unbiased and has a variance that goes to zero as N tends to infinity.

The use of bias, convergence and consistency is well studied in Chapter 13 for spectral estimation techniques.

RANDOM PROCESSES 12.6

A random process denoted by $X(t)$ is a mapping from a sample space to a set of time-varying signals or waveforms. If the time-varying signal is a continuous time signal, then the random process is a continuous time signal, and a continuous random process. Otherwise, if it is a discrete-time signal, then the associated random process is a discrete-time random process. The analysis of discrete-time random processes is considered in detail.

A discrete-time random process, denoted by $X(n)$, is an ensemble or collection of discrete time signals $x(n)$. It is nothing but a mapping of sample space S to a set of discrete-time signals as shown in Fig. 12.3.

At time $n = n_0$, the random process may be viewed as a random variable having values,

$$X\{n_0\} = X = \{x_1(n_0), x_2(n_0), \ldots, x_m(n_0)\}.$$

The random process $x_i(n)$ can also be viewed as a random variable.

The probability distribution function of the collection of random variables is given by

$$F_{x(n_1)x(n_2)\ldots x(n_k)}(x_1, x_2, \ldots, x_k) = P\left[x(n_1) \le x_1,\ x(n_2) \le x_2, \ldots, x(n_k) \le x_k\right] \tag{12.16}$$

and the joint density function is given by

$$f_{x(n_1),x(n_2)\ldots x(n_k)}\left(x_1, x_2, \ldots, x_k\right) = \frac{\partial^k}{\partial x_1 \partial x_2 \ldots \partial x_k}\left[F_{x(n_1)x(n_2)\ldots x(n_k)}\left(x_1, x_2, \ldots, x_k\right)\right] \tag{12.17}$$

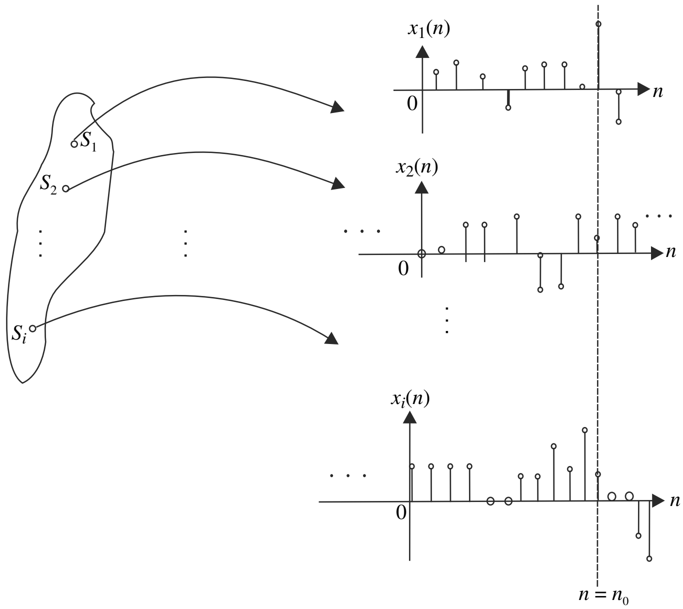


Fig. 12.3 *Mapping of Events in a Sample Space to Discrete-time Signals*

The discrete-time random process is an indexed set of random variables, and the mean of these random variables is defined as

$$\mu_x(n) = E[x(n)] \tag{12.18}$$

and the variance of each random variable is the process as

$$\sigma_x^2(n) = E\left[\left|x(n) - \mu_x(n)\right|^2\right] \tag{12.19}$$

Additionally, the autocovariance and autocorrelation of the random variables are also defined, which are very useful in the study of a random process.

The autocovariance,

$$c_{xx}(k,\ell) = E\left[\left[\left(x(k)\right) - \mu_x(k)\right]\left[x(\ell) - \mu_x(\ell)\right]^*\right] \tag{12.20}$$

and the autocorrelation

$$r_{xx}(k,\ell) = E\,[x(k)x^*(\ell)] \tag{12.21}$$

for the random variables $x(k)$ and $x(\ell)$. If $k = \ell$, the autocovariance becomes the variance.

The autocorrelation and autocovariance give information about two different random variables derived from the same random process. If more than one random process is involved for any application, cross-covariance and cross-correlation for them are defined.

The cross-covariance is

$$c_{xy}(k,\ell) = E\,[\{x(k) - \mu_x(k)\}\{y(\ell) - \mu_y(\ell)\}^*] \tag{12.22}$$

The cross-correlation is

$$r_{xy}(k,\ell) = E\,[x(k)\,y^*(\ell)] \tag{12.23}$$

Two random processes are said to be uncorrelated if $c_{xy}(k,\ell) = 0$

Two random processes are said to be orthogonal if $r_{xy}(k,\ell) = 0$

Although orthogonal random processes are not necessarily uncorrelated, zero mean processes that are uncorrelated are orthogonal.

STATIONARY PROCESS 12.7

In signal processing applications, the statistical averages are often independent of time. This represents the property of statistical time invariance, also called *stationarity*.

If the joint probability density function of a random process does not vary with respect to time, then the random process is said to be *stationary* or *strict-sense stationary*.

A random process is said to be stationary in the wide sense, i.e. *Wide Sense Stationary (WSS)* if the following conditions are true.

(i) The mean of the process is a constant, i.e., $\mu_{xx}(n) = \mu_{xx}$

(ii) The autocorrelation depends only on the time difference, i.e. $r_{xx}(k,\ell) = r_{xx}(k-\ell)$

(iii) The variance of the process is finite, i.e. $c_{xx}(0) < \infty$

The important properties of the autocorrelation of a wide sense stationary process are

(i) The autocorrelation sequence of a WSS random process is a conjugate symmetric function,

$$r_{xx}(k) = r_{xx}^*(-k)$$

For a real process,

$$r_{xx}(k) = r_{xx}(-k)$$

(ii) The autocorrelation sequence of a WSS process at $k = 0$ is equal to the mean square value of the process.

$$r_{xx}(0) = E\left[\left(x(n)\right)^2\right] \geq 0$$

(iii) The magnitude of the autocorrelation sequence of a process at 'k' is bounded by its value at $k = 0$,

$$r_{xx}(0) \geq r_{xx}(k)$$

Example 12.1 *A periodic process is given by $x(n) = A\cos(n\omega_0 + \phi)$ where ϕ is the random process that uniformly varies between 0 and 2π. Check whether the process is wide sense stationary.*

Solution For a process to be wide sense stationary,

$$E[x(n)] = \text{constant}$$

and $r_{xx}(k, l) = r_{xx}(k-l)$, should be function of time difference

$$x(n) = A\cos(n\omega_0 + \phi)$$

and $$f(\phi) = \begin{cases} \dfrac{1}{2\pi}; & 0 \leq \phi \leq 2\pi \\ 0; & \text{elsewhere} \end{cases}$$

Therefore, $$E[x(n)] = \int_{-\infty}^{\infty} x(n) f(\phi)\, d\phi$$

$$= \int_0^{2\pi} A\cos(n\omega_0 + \phi)\frac{1}{2\pi}\, d\phi = \frac{A}{2\pi}\left[\sin(n\omega_0 + \phi)\right]_0^{2\pi}$$

$$= \frac{A}{2\pi}\left[\sin(n\omega_0 + 2\pi) - \sin(n\omega_0)\right] = \frac{A}{2\pi}\left[\sin n\omega_0 - \sin n\omega_0\right] = 0$$

Therefore, $E[x(n)] = 0$, which is a constant.

$$r_{xx}(k,l) = E[x(k)\, x^*(l)]$$

$$= E[A\sin(k\omega_0 + \phi)\, A\sin(l\omega_0 + \phi)]$$

$$= \frac{A^2}{2} E\left[\cos\left[(k-l)\,\omega_0\right] - \cos\left[(k+l)\omega_0 + 2\phi\right]\right]$$

$$= \frac{A^2}{2}\left[\cos(k-l)\omega_0 - \int_0^{2\pi} \cos\left((k+l)\,\omega_0 + 2\phi\right) f(\phi)\, d\phi\right]$$

$$= \frac{A^2}{2}\left[\cos(k-l)\omega_0 - \frac{1}{2\pi}\left[\frac{\sin\left((k+l)\omega_0 + 2\phi\right)}{2}\right]_0^{2\pi}\right]$$

$$= \frac{A^2}{2}\left[\cos(k-l)\omega_0 - \frac{1}{2\pi}\left[\frac{\sin\left((k+l)\omega_0 + 4\pi\right)}{2} - \frac{\sin(k+l)\omega_0}{2}\right]\right]$$

$$= \frac{A^2}{2}\cos\left[(k-l)\omega_0\right]$$

Therefore, $r_{xx}(k-l)$ is a function of time difference. Thus, the given process is a wide sense stationary process.

12.8 THE AUTOCORRELATION AND AUTOCOVARIANCE MATRICES

The important second-order characterisations like autocorrelation and autocovariance can be conveniently represented in the form of matrices. Let us consider a discrete-time random process represented by the vector,

$$\mathrm{x} = [x(0), x(1), ..., x(p)]^T$$

Its outer product can be obtained by multiplying with a Hermitian transpose which is defined as the conjugate transpose of a matrix.

$$\mathrm{xx}^H = \left[x(0), x(1), \ldots, x(p)\right]^T \left[x^*(0), x^*(1), \ldots, x^*(p)\right]$$

$$= \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(p) \end{bmatrix} \left[x^*(0)\; x^*(1)\; \ldots\; x^*(p)\right]$$

$$= \begin{bmatrix} x(0)\,x^*(0) & x(0)\,x^*(1) & \ldots & x(0)\,x^*(p) \\ x(1)\,x^*(0) & x(1)\,x^*(1) & \ldots & x(1)\,x^*(p) \\ \vdots & \vdots & & \vdots \\ x(p)\,x^*(0) & x(p)\,x^*(1) & \ldots & x(p)\,x^*(p) \end{bmatrix}$$

Taking expectation on both sides, we get

$$E\left[\mathrm{xx}^H\right] = E\begin{bmatrix} x(0)\,x^*(0) & x(0)\,x^*(1) & \ldots & x(0)\,x^*(p) \\ x(1)\,x^*(0) & x(1)\,x^*(1) & \ldots & x(1)\,x^*(p) \\ \vdots & \vdots & & \vdots \\ x(p)\,x^*(0) & x(p)\,x^*(1) & \ldots & x(p)\,x^*(p) \end{bmatrix}$$

$$= \begin{bmatrix} E[x(0)\,x^*(0)] & E[x(0)x^*(1)] & \dots & E[x(0)x^*(p)] \\ E[x(1)\,x^*(0)] & E[x(1)x^*(1)] & \dots & E[x(1)x^*(p)] \\ \vdots & \vdots & & \vdots \\ E[x(p)\,x^*(0)] & E[x(p)x^*(1)] & \dots & E[x(p)x^*(p)] \end{bmatrix}$$

$$= \begin{bmatrix} r_{xx}(0) & r_{xx}(-1) & \cdots & r_{xx}(-p) \\ r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(1-p) \\ \vdots & \vdots & & \vdots \\ r_{xx}(p) & r_{xx}(p-1) & \cdots & r_{xx}(0) \end{bmatrix}$$

As $r_{xx}(-k) = r^*_{xx}(k)$, we have

$$R_{xx} = E\left[\mathrm{xx}^H\right] = \begin{bmatrix} r_{xx}(0) & r^*_{xx}(1) & \cdots & r^*_{xx}(p) \\ r_{xx}(1) & r^*_{xx}(0) & \cdots & r^*_{xx}(p-1) \\ \vdots & \vdots & & \vdots \\ r_{xx}(p) & r_{xx}(p-1) & \cdots & r_{xx}(0) \end{bmatrix} \tag{12.24}$$

is a $(p + 1) \times (p + 1)$ matrix of autocorrelation values which is called *autocorrelation matrix of the process* $x(n)$.

Similarly, the autocovariance matrix is represented as,

$$C_{xx} = E\,[(x - m_x)\,(x - m_x)^H] \tag{12.25}$$

where $m_x = [m_x\, m_x \,...\, m_x]^T$ is a vector having $(p + 1)$ mean values of the process.

The autocorrelation and autocovariance matrices are related by

$$C_{xx} = R_{xx} - m_x\, m_x^H \tag{12.26}$$

For a process with zero mean,

$$C_{xx} = R_{xx}$$

The importance of the autocorrelation matrix of a WSS process is that

(i) It is a Hermitian matrix (any matrix is said to be Hermitian if its entries are equal to its own conjugate transpose. For any matrix A, $A = (A^*)^T$)

(ii) All the terms along each of the diagonal elements are equal.

Thus R_{xx} is a *Hermitian Toeplitz matrix*. (A Toeplitz matrix is a matrix in which the entries in the diagonal and the sub-diagonal elements are same.)

The properties of the autocorrelation matrix are mentioned below.

Property 1 The autocorrelation matrix of a WSS random process *x(n)* is a Hermitian Toeplitz matrix.

$$R_{xx} = \text{Toep}\,\{r_{xx}(0), r_{xx}(1), ..., r_{xx}(p)\}$$

But, the converse is not true, i.e. every Hermitian Toeplitz matrix is not a valid autocorrelation matrix. For example,

$$R_{xx} = \begin{bmatrix} -4 & 2 \\ 2 & -4 \end{bmatrix} \text{ and } R_{xx} = \begin{bmatrix} 1 & -5 & 2 \\ -5 & 1 & -5 \\ 2 & -5 & 1 \end{bmatrix}$$

cannot be a valid autocorrelation matrix of a WSS process because $r_{xx}(0)$ is always non-negative and satisfies $r_{xx}(0) \geq r_{xx}(k)$.

Property 2 The autocorrelation matrix of a WSS process is non-negative definite, that is,

$$R_{xx} > 0$$

Property 3 The Eigen values λ_k of the autocorrelation matrix of a WSS process are real and non-negative.

Example 12.2 *Find the 2 × 2 autocorrelation matrix of a random phase sinusoid, whose autocorrelation sequence is given by*

$$r_{xx}(k) = \frac{A^2}{2}\cos(k\omega_0)$$

Also show that their Eigen values are non-negative.

Solution

$$R_{xx} = \begin{bmatrix} r_{xx}(0) & r_{xx}^*(1) \\ r_{xx}(1) & r_{xx}(0) \end{bmatrix}$$

Given $r_{xx}(k) = \frac{A^2}{2}\cos(k\omega_0)$

Therefore, $r_{xx}(0) = \frac{A^2}{2}$

$$r_{xx}(1) = r_{xx}^*(1) = \frac{A^2}{2}\cos\omega_0$$

Hence, $R_{xx} = \frac{A^2}{2}\begin{bmatrix} 1 & \cos\omega_0 \\ \cos\omega_0 & 1 \end{bmatrix}$

The Eigen values of R_{xx} are determined as follows:

$$|R_{xx} - \lambda I| = 0$$

$$\left| \frac{A^2}{2}\begin{bmatrix} 1 & \cos\omega_0 \\ \cos\omega_0 & 1 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right| = 0$$

$$\frac{A^2}{2}\begin{vmatrix} 1-\lambda & \cos\omega_0 \\ \cos\omega_0 & 1-\lambda \end{vmatrix} = 0$$

$$\frac{A^2}{2}\left[(1-\lambda)^2 - \cos^2\omega_0\right] = 0$$

$$(1-\lambda)^2 = \cos^2\omega_0$$

$$1-\lambda = \pm\cos\omega_0$$

Therefore, $\lambda = 1 \pm \cos\omega_0$

Therefore, $\lambda_1 = 1 + \cos\omega_0$ and $\lambda_2 = 1 - \cos\omega_0$ are the Eigen values R_{xx}.

It is clear that the Eigen values are non-negative.

Example 12.3 *A complex-valued random phase process is given by*

$$y(n) = Ae^{j(n\omega_1 + \phi_1)} + Ae^{j(n\omega_2 + \phi_2)}$$

where ϕ_1 and ϕ_2 are uniformly distributed between 0 and 2π. Find its 2 × 2 autocorrelation matrix.

Solution The autocorrelation sequence of a single complex exponential can be computed,

Let $x(n) = Ae^{j(n\omega_0 + \phi_0)}$

Therefore, $r_{xx}(k) = E\,[x(n + k)\, x^*(n)]$

$= E\,[Ae^{j(n + k)\omega_0 + \phi_0)} \,.\, A^* e^{-j(n\omega_0 + \phi_0)}] = |A|^2\, E\,[e^{jk\omega_0}] = |A|^2\, e^{jk\omega_0}$

Hence, $y(n) = Ae^{j(n\omega_1 + \phi_1)} + Ae^{j(n\omega_2 + \phi_2)}$

Thus, for the sum of the two uncorrelated processes $y(n)$, the autocorrelation sequence is given by

$$r_{yy}(k) = |A|^2 e^{jk\omega_1} + |A|^2 e^{jk\omega_2}$$

and the 2 × 2 autocorrelation matrix is given by

$$R_{yy}\,|A|^2 \begin{bmatrix} 2 & e^{-j\omega_1} + e^{-j\omega_2} \\ e^{j\omega_1} + e^{j\omega_2} & 2 \end{bmatrix}$$

ERGODIC PROCESSES 12.9

The mean and autocorrelation of a WSS random process are called *ensemble averages* that are defined over the ensemble of all possible discrete-time signals. But these ensemble averages are generally unknown a priori, whereas they are essential for problems such as filtering, spectrum estimation, etc. Therefore, to estimate these averages for any random process, we approximate it from their time averages. The time averages can be calculated if a single realisation of a process, i.e. N samples of a process are known. For example, the same mean taken as the average over time for a process $x(n)$ is given by

$$\hat{m}_x(N) = \frac{1}{N}\sum_{n=0}^{N-1} x(n) \tag{12.27}$$

where $x(n) = [x(0), x(1), \cdots, x(N-1)]$

This sample mean is called the time average value of process. A process in which the time averaged values converge to the ensemble averages is called *ergodic process*, that is,

i.e. $$\underset{N\to\infty}{\text{Lt}}\ \hat{m}_x(N) = m_x \tag{12.28}$$

Then the process is said to be ergodic in the mean.

A process is said to be autocorrelation ergodic if

$$\underset{N\to\infty}{\text{Lt}}\ \hat{r}_{xx}(k, N) = r_{xx}(k) \tag{12.29}$$

where $\hat{r}_{xx}(k, N)$ is the time averaged autocorrelation.

Example 12.4 *Given the harmonic random phase sinusoid $x(n) = A\cos(n\omega_0 + \phi)$, where ϕ varies uniformly between 0 and 2π. Check whether $x(n)$ is mean ergodic.*

Solution The ensemble mean is given by

$$m_x(n) = E\,[x(n)]$$

and the time averaged mean is given by

$$\hat{m}_x(N) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)$$

Now, $$m_x(n) = E\,[A\cos(n\omega_0 + \phi)] = A\,\frac{1}{2\pi}\int_0^{2\pi}\cos(n\omega_0 + \phi)\,d\phi = A\left[\sin(n\omega_0 + \phi)\right]_0^{2\pi} = 0$$

Similarly,

$$\hat{m}_x(N) = \frac{1}{N}\sum_{n=0}^{N-1} A\cos\left(n\omega_0 + \phi\right)$$

$$= \frac{A}{N}\left[\cos\phi + \cos\left(\omega_0 + \phi\right) + \cos\left(2\omega_0 + \phi\right) + \ldots + \cos\left(N-1\right)\omega_0 + \phi\right]$$

If $N \to \infty$,

$$\underset{N\to\infty}{\text{Lt}}\ \hat{m}_x(N) = 0$$

Hence, $x(n)$ is ergodic in the mean.

POWER SPECTRUM 12.10

The spectral analysis plays an important role in the analysis of a random process. Autocorrelation sequence of a WSS process provides a time-domain representation of the second order moment of a process, which is a deterministic sequence. So, if we take Fourier transform for the autocorrelation sequence, we have

$$P_x(e^{j\omega}) = \text{DTFT}\ [r_{xx}(k)]$$

$$P_x(e^{j\omega}) = \sum_{k=-\infty}^{\infty} r_{xx}(k)e^{-jk\omega} \tag{12.30}$$

which is defined as the power spectrum or power spectral density of the process. Similarly, if the power spectral density is known, the autocorrelation sequence can be determined as,

$$r_{xx}(k) = \text{IDTFT}\ [P_x(e^{j\omega})]$$

Therefore,

$$r_{xx}(k) = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x(e^{j\omega})\, e^{jk\omega} d\omega \tag{12.31}$$

The properties of power spectrum are mentioned below.

Property 1: Symmetry The power spectrum of a *WSS* process $x(n)$ is real valued.

$$P_x(e^{j\omega}) = P_x^*(e^{j\omega})$$

If $x(n)$ is real, $P_x(e^{j\omega}) = P_x(e^{-j\omega})$, i.e. power spectrum is even.

Property 2: Positivity The power spectrum of a *WSS* process is non-negative.

$$P_x(e^{j\omega}) \geq 0$$

Property 3: Total Power The power in a zero mean WSS process is proportional to the area under the graph of power spectral density.

$$E\left[\left|x(n)\right|^2\right] = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x\left(e^{j\omega}\right) d\omega$$

Property 4: Eigen value external property For a zero mean WSS process, the eigen values of the autocorrelation matrix are upper bounded by the maximum value of power spectrum and lower bounded by the minimum value of power spectrum

$$\min_{\omega} P_x\left(e^{j\omega}\right) \leq \lambda_i \leq \max_{\omega} P_x\left(e^{j\omega}\right)$$

White Noise

White noise is an important and fundamental discrete-time process which is frequently encountered in the analysis of discrete-time random process. A wide sense stationary process $v(n)$ is said to be white if its autocovariance function is given by

$$c_v(k) = \sigma_v^2 \delta(k).$$

Example 12.5 *The autocorrelation sequence of a zero mean process is given by, $r_{xx}(k) = \sigma_x^2\, \delta(k)$, where σ_x^2 is the variance of the process. Find its power spectrum.*

Solution The power spectrum,

$$P_x\left(e^{j\omega}\right) = \sum_{K=-\infty}^{\infty} \sigma_x^2 \delta(k)\, e^{-jk\omega}$$

Hence, $P_x(e^{j\omega}) = \sigma_x^2$ is a constant.

Example 12.6 *The random phase sinusoid $x(n)$ has an autocorrelation sequence $r_{xx}(k) = \frac{A^2}{2}\cos(k\omega_0)$. Find its power spectrum.*

Solution

$$P_x\left(e^{j\omega}\right) = \sum_{K=-\infty}^{\infty} r_{xx}(k) e^{-jk\omega}$$

$$= \sum_{K=-\infty}^{\infty} \frac{A^2}{2}\cos(k\omega_0).e^{-jk\omega} = \sum_{K=-\infty}^{\infty} \frac{A^2}{2}\left(\frac{e^{jk\omega_0} + e^{-jk\omega_0}}{2}\right) e^{-jk\omega}$$

We know that,

$$\text{DTFT } [e^{jk\omega_0}] = 2\pi\delta(\omega - \omega_0)$$

Therefore, $P_x\left(e^{j\omega}\right) = \frac{A^2}{4}.\left[2\pi\delta(\omega-\omega_0) + 2\pi\delta(\omega+\omega_0)\right] = \frac{\pi A^2}{2}\left[\delta(\omega-\omega_0) + \delta(\omega+\omega_0)\right].$

Example 12.7 *Find the power spectrum of the process whose autocorrelation sequence is given by $r_x(k) = a^{|k|}$, where $|a|<1$.*

Solution The power spectrum of the process is given by

$$P_x\left(e^{j\omega}\right) = \sum_{K=-\infty}^{\infty} a^{|k|} e^{-jk\omega}$$

$$= \sum_{K=-\infty}^{-1} a^{-k} e^{-jk\omega} + \sum_{k=0}^{\infty} a^k e^{-jk\omega} = \sum_{K=\infty}^{1} a^k e^{jk\omega} + \sum_{k=0}^{\infty} a^k\, e^{-jk\omega}$$

$$= \left(\sum_{k=0}^{\infty} a^k e^{jk\omega} - 1\right) + \sum_{k=0}^{\infty} a^k e^{-jk\omega} = \sum_{k=0}^{\infty} \left(ae^{j\omega}\right)^k + \sum_{k=0}^{\infty} \left(a\, e^{-j\omega}\right)^k - 1$$

$$= \frac{1}{1 - a\, e^{j\omega}} + \frac{1}{1 - a\, e^{-j\omega}} - 1$$

$$= \frac{\left(1-a\,e^{-j\omega}\right)+\left(1-a\,e^{j\omega}\right)-\left(1-a\,e^{-j\omega}\right)\left(1-a\,e^{j\omega}\right)}{\left(1-a\,e^{j\omega}\right)\left(1-a\,e^{-j\omega}\right)}$$

$$= \frac{1-a\,e^{-j\omega}+1-a\,e^{j\omega}-\left(1-a^2-a\,e^{-j\omega}-a\,e^{j\omega}\right)}{1-ae^{j\omega}-ae^{-j\omega}+a^2}$$

$$P_x\left(e^{j\omega}\right) = \frac{1-a^2}{1-2a\cos\omega+a^2}$$

FILTERING RANDOM PROCESSES 12.11

The inputs to linear time invariant systems are often random processes and it is mandatory to find how the statistics of these processes change when filtered. This section determines the relationship between the mean and autocorrelation of the input process to that of the output process.

Let $x(n)$ be a WSS process with mean m_k and autocorrelation $r_{xx}(k)$. Let $x(n)$ be filtered by an LTI filter with impulse response $h(n)$. Now the output $y(n)$ is also a random process given by

$$y(n) = x(n) * h(n)$$

Therefore,
$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) \tag{12.32}$$

The mean of the output process is found as

$$m_y = E\left[y(n)\right] = E\left[\sum_{k=-\infty}^{\infty} x(k)h(n-k)\right]$$

$$= E\sum_{k=-\infty}^{\infty} h(k)x(n-k) = \sum_{k=-\infty}^{\infty} h(k)\; E\left[x(n-k)\right]$$

$$= m_x \sum_{k=-\infty}^{\infty} h(k) = m_x H(e^{j0})$$

Therefore,
$$m_y = m_x\, H(e^{j0}) \tag{12.33}$$

Thus, the mean of the output process is also a constant and is equal to the mean of the input process scaled by the frequency response of the filter at $\omega = 0$.

To find the autocorrelation of the output process, the cross-correlation of $x(n)$ and $y(n)$ is first computed.

$$r_{yx}(n+k,n) = E[y(n+k)x^*(n)] \tag{12.34}$$

$$= E\left[\sum_{\ell=-\infty}^{\infty} h(\ell)x(n+k-\ell)\; x^*(n)\right]$$

$$= \sum_{\ell=-\infty}^{\infty} h(\ell)\; E\left[x(n+k-\ell)x^*(n)\right]$$

$$= \sum_{\ell=-\infty}^{\infty} h(\ell) r_{xx}(k-\ell)$$

Thus,
$$r_{yx}(k) = r_{xx}(k) * h(k) \tag{12.35}$$

Now, the autocorrelation of $y(n)$ can be found as

$$r_{yy}(n+k,n) = E[y(n+k)y^*(n)]$$

$$= E\left[y(n+k)\sum_{\ell=-\infty}^{\infty} x^*(\ell)h^*(n-\ell)\right]$$

$$= \sum_{\ell=-\infty}^{\infty} h^*(n-\ell)\ E\left[y(n+k)\ x^*(\ell)\right] = \sum_{\ell=-\infty}^{\infty} h^*(n-\ell)\ r_{yx}(n+k-\ell)$$

Letting $m = n - \ell$, we have

$$r_{yy}(n+k,n) = \sum_{m=-\infty}^{\infty} h^*(m)r_{yx}(m+k) = r_{yx}(k) * h^*(-k)$$

Therefore, $\quad r_{yy}(k) = r_{yx}(k)*h^*(-k) \quad$ (12.36)

Substituting (12.36) in (12.35), we get

$$r_{yy}(k) = r_{xx}(k)*h(k)*h^*(-k) \tag{12.37}$$

Thus, if $x(n)$ is WSS, then $y(n)$ will also be a WSS random process.

The power spectrum of output process $y(n)$ is then given by

$$P_y(e^{j\omega}) = \text{DTFT}\,[r_{yy}(k)] = \text{DTFT}\,[r_{xx}(k)*h(k)*h^*(-k)]$$

$$= P_x(e^{j\omega})\ H(e^{j\omega})\ H^*(e^{j\omega})$$

$$P_y(e^{j\omega}) = P_x(e^{j\omega})|\text{H}(e^{j\omega})|^2 \tag{12.38}$$

In terms of z-transform,

$$P_y(z) = P_x(z)H(z)H^*\left(\frac{1}{z^*}\right) \tag{12.39}$$

If $h(n)$ is real, $\quad H(z) = H^*(z^*)$

Then, $\quad P_y(z) = P_x(z)H(z)H\left(\frac{1}{z}\right) \quad$ (12.40)

Example 12.8 *Let x(n) be a random process that is generated by filtering a unit variance white noise v(n) by an LTI filter having transfer function.*

$$H(z) = \frac{1}{1-0.5z^{-1}}$$

Find the autocorrelation of the output process x(n).

Solution

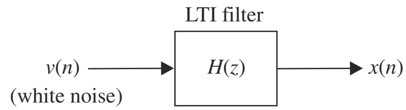


Fig. E12.8 *Random Process*

The power spectrum of the output process $x(n)$ is

$$P_x(z) = \sigma_v^2\ H(z)\ H(z^{-1})$$

Given $\sigma_v^2 = 1$

Therefore, $\quad P_x(z) = H(z)\ H(z^{-1})$

$$= \frac{1}{\left(1-0.5z^{-1}\right)}\frac{1}{(1-0.5z)} = \frac{z}{(z-0.5)(1-0.5z)}$$

$$= \frac{z}{-0.5(z-2)(z-0.5)} = \frac{-2z}{(z-0.5)(z-2)}$$

$$\frac{P_x(z)}{z} = \frac{-2}{(z-0.5)(z-2)}$$

Using partial fraction,

$$\frac{P_x(z)}{z} = \frac{A_1}{z-0.5} + \frac{A_2}{z-2}$$

Upon solving, we get

$$A_1 = \frac{4}{3} \text{ and } A_2 = -\frac{4}{3}$$

Therefore, $P_x(z) = \dfrac{\frac{4}{3}z}{z-0.5} - \dfrac{\frac{4}{3}z}{z-2}$

Taking inverse Fourier transform, we get

$$r_{xx}(k) = \frac{4}{3}(0.5)^k u(k) - \frac{4}{3}(2)^k u(k-1)$$

Example 12.9 *Determine the impulse response of the filter which generates a random process having power spectrum of the form*

$$P_x\left(e^{j\omega}\right) = \frac{25+24\cos\omega}{26+10\cos\omega}$$

by filtering unit variance white noise.

Solution Given $P_x\left(e^{j\omega}\right) = \dfrac{25+24\cos\omega}{26+10\cos\omega} = \dfrac{25+24\left(\dfrac{e^{j\omega}+e^{-j\omega}}{2}\right)}{26+10\left(\dfrac{e^{j\omega}+e^{-j\omega}}{2}\right)}$

Letting $z = e^{j\omega}$, we have

$$P_x(z) = \frac{25+24\left(\dfrac{z+z^{-1}}{2}\right)}{26+10\left(\dfrac{z+z^{-1}}{2}\right)} = \frac{12z+25+12z^{-1}}{5z+26+5z^{-1}}$$

$$= \frac{12z+9+16+12z^{-1}}{5z+25+1+5z^{-1}} = \frac{3z\left(4+3z^{-1}\right)+4\left(4+3z^{-1}\right)}{5z\left(1+5z^{-1}\right)+1\left(1+5z^{-}\right)}$$

$$P_x(z) = \frac{(4+3z)\left(4+3z^{-1}\right)}{(1+5z)\left(1+5z^{-1}\right)}$$

From the factorisation $P_x(z) = \sigma_v^2 H(z)\, H(z^{-1})$, we have

$$H(z) = \frac{4+3z}{1+5z} = \frac{3z+4}{5z+1} = \frac{3z\left(1+\dfrac{4}{3}z^{-1}\right)}{5z\left(1+\dfrac{1}{5}z^{-1}\right)} = \frac{3}{5}\left[\frac{1}{1+\dfrac{1}{5}z^{-1}} + \frac{4}{3}\frac{z^{-1}}{1+\dfrac{1}{5}z^{-1}}\right]$$

Taking inverse z-transform, we get

$$h(n) = \frac{3}{5}\left(\frac{-1}{5}\right)^n u(n) + \frac{4}{5}\left(\frac{-1}{5}\right)^{n-1} u(n-1)\,.$$

SPECTRAL FACTORISATION THEOREM 12.12

If the power spectrum $P_x(e^{j\omega})$ of a wide sense stationary process is a real valued, positive and periodic function of ω, then $P_x(z)$ can be factorised as

$$P_x(z) = \sigma_0^2 Q(z) Q^*\left(\frac{1}{z^*}\right)$$

This is known as *spectral factorisation* of $P_x(z)$.

Consider that $x(n)$ is a WSS process, $r_{xx}(k)$ is autocorrelation and $P_x(e^{j\omega})$ is the power spectrum which is a continuous function of ω, and $x(n)$ contains no periodic components.

Hence, $$P_x(z) = \sum_{k=-\infty}^{\infty} r_{xx}(k) z^{-k}$$

Also, assume that $\ln[P_x(z)]$ is analytic in an annulus $\rho < |z| < 1/\rho$ which contains the unit circle. The logarithm of $P_x(z)$ and its derivatives are continuous functions of z, and hence $\ln[P_x(z)]$ can be expanded in a Laurent series of the form

$$\ln\left[P_x(z)\right] = \sum_{k=-\infty}^{\infty} c(k) e^{-jk\omega}$$

where $c(k)$ are the coefficients of the expansion. Thus, $c(k)$ is the sequence that has $\ln[P_x(z)]$ as its z-transform. Alternatively, $\ln[P_x(z)]$ is calculated on the unit circle as

$$\ln\left[P_x\left(e^{j\omega}\right)\right] = \sum_{k=-\infty}^{\infty} c(k) e^{-jk\omega}$$

Here $c(k)$ is also the Fourier coefficients of the periodic function $\ln[P_x(e^{j\omega})]$.

Therefore, $$c(k) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \ln\left[P_x\left(e^{j\omega}\right)\right] e^{jk\omega} d\omega \tag{12.41}$$

Since $P_x(e^{j\omega})$ is real, the coefficients $c(k)$ are conjugate symmetric, $c(-k) = c^*(k)$. Also, $c(0)$ is proportional to the area under the logarithm of the power spectrum,

$$c(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \ln\left[P_x\left(e^{j\omega}\right)\right] d\omega$$

Using the expansion given in Eq. (12.41), the power spectrum in factored form is expressed as

$$P_x(z) = \exp\left\{\sum_{k=-\infty}^{\infty} c(k) z^{-k}\right\} \tag{12.42}$$

$$= \exp\{c(0)\}\exp\left\{\sum_{k=1}^{\infty} c(k) z^{-k}\right\}\exp\left\{\sum_{k=-\infty}^{-1} c(k) z^{-k}\right\}$$

and $$Q(z) = \exp\left\{\sum_{k=1}^{\infty} c(k) z^{-k}\right\}, |z| > \rho$$

which is the z-transform of a causal and stable sequence, $q(k)$. Hence, $Q(k)$ is expanded in a power series of the form

$$Q(z) = 1 + q(1)z^{-1} + q(2)z^{-2} + ...$$

where $q(0) = 1$ since $Q(\infty) = 1$. As the functions $Q(z)$ and $\ln[Q(z)]$ are analytic for $|z| > \rho$, $Q(z)$ is a minimum phase filter. For a rational function of z, $Q(z)$ has no poles or zeros outside the unit circle and hence $Q(z)$ has a stable and causal inverse $1/Q(z)$. Using the conjugate symmetry of $c(k)$, the second factor in (12.42) in terms of $Q(z)$ is expressed as

$$\exp\left\{\sum_{k=-\infty}^{-1} c(k)z^{-k}\right\} = \exp\left\{\sum_{k=1}^{\infty} c^*(k)z^*\right\} = \exp\left\{\sum_{k=1}^{\infty} c(k)\left(1/z^*\right)^{-k}\right\}^* = Q^*\left(1/z^*\right)$$

Thus, the spectral factorisation of the power spectrum $P_x(z)$ can be written as

$$P_x(z) = \sigma_0^2\, Q(z)Q^*(1/z^*) \tag{12.43}$$

where $\sigma_0^2 = \exp\{c(0)\} = \exp\left\{\frac{1}{2\pi}\int_{-\pi}^{\pi} \ln P_x\left(e^{j\omega}\right)d\omega\right\}$

which is real and nonnegative. For a real-valued process, the spectral factorisation takes the form

$$P_x(z) = \sigma_0^2\, Q(z)Q(z^{-1}) \tag{12.44}$$

Using Eqs. (12.43) and (12.44), any WSS process can be factorised.

Figure 12.4 represents the WSS process using spectral factorisation, where $H(z)$ is a causal and

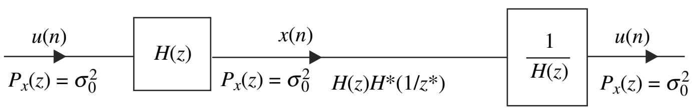


Fig. 12.4 *Representation of a WSS Process using Spectral Factorisation*

stable filter and $\dfrac{1}{H(z)}$ is the whitening filter. As $u(n)$ and $x(n)$ are related by an invertible transformation, either process may be derived from the other. Hence, both the processes contain the same information. As a specific case where $P_x(z)$ is a rational function, that is

$$P_x(z) = \frac{N(z)}{D(z)}$$

the spectral factorization given by Eq.(12.44) can be written as

$$P_x(z) = \sigma_0^2 Q(z)Q^*\left(\frac{1}{z^*}\right) = \sigma_0^2\left[\frac{A_2(z)}{A_1(z)}\right]\left[\frac{A_2^*\left(\frac{1}{z^*}\right)}{A_1^*\left(\frac{1}{z^*}\right)}\right]$$

where the roots of $A_1(z)$ and $A_2(z)$ lie inside the unit circle. Due to symmetry property of the power spectrum, for every pole or zero, there will be a matching conjugate pole or zero. To generalize, the representation of a WSS process can be decomposed into a sum of two orthogonal processes according to Wold decomposition theorem. Hence, any general process can be written as a sum of two processes given by

$$x(n) = x_p(n) + x_r(n)$$

where

$x_p(n)$ is a predictable process
$x_r(n)$ is a regular random process and

such that

$$E\left[x_p(n)x_r^*(m)\right] = 0$$

REVIEW QUESTIONS

12.1 What is a random variable?
12.2 What are ensemble averages?
12.3 Define autocorrelation.
12.4 State the properties of a zero mean process.
12.5 Compute the autocorrelation of the sequence, $x(n) = (0.5)^n\, u(n)$.
12.6 State Wiener–Khinchine theorem.
12.7 Compare power density spectrum with cross-power density spectrum.
12.8 What is a random process?
12.9 List the properties of a WSS process.
12.10 Find the power spectrum for each of the following WSS processes, given the autocorrelation
(a) $r_{xx}(k) = \delta(k) + 3(0.5)^{|k|}$
(b) $r_{xx}(k) = \delta(k) + j2\delta(k-1) - j2\delta(k+1)$
(c) $r_{xx}(k) = \begin{cases} 10-|k|, & |k| \le 10 \\ 0, & \text{otherwise} \end{cases}$
12.11 Find the autocorrelation of the WSS processes whose power spectral densities are
$P_x(e^{j\omega}) = 5 + 4\cos 2\omega$ and $P_x(e^{j\omega}) = \dfrac{1}{5+3\cos\omega}$
12.12 If the output random process $y(n)$ is obtained by filtering WSS process $x(n)$ with a stable LTI filter, $h(n)$, write down the relation between $y(n)$ and $x(n)$ in terms of autocorrelation $r_{xx}(k)$.
12.13 A linear shift invariant system is described by $H(z) = \dfrac{1-0.5z^{-1}}{1-0.33z^{-1}}$
which is excited by a zero mean exponentially correlated noise $x(n)$ with an autocorrelation sequence $r_{xx}(k) = (0.5)^{|k|}$. Let $y(n)$ be the output of the process, $y(n) = x(n)*h(n)$. Determine the (i) power spectrum $P_y(z)$ of $y(n)$, (ii) autocorrelation of $y(n)$ and (iii) cross power spectral density $P_{xy}(z)$.
12.14 Determine whether the following autocorrelation matrices are valid or not. Justify the answer.

(a) $R_1 = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$ (b) $R_2 = \begin{bmatrix} 4 & 2 & 2 \\ -2 & 4 & 2 \\ -2 & -2 & 4 \end{bmatrix}$ (c) $R_3 = \begin{bmatrix} 1 & 2 & 2 \\ 2 & 1 & 2 \\ 2 & 2 & 1 \end{bmatrix}$

(d) $R_4 = \begin{bmatrix} 1 & 1+j \\ 1-j & 1 \end{bmatrix}$ (e) $R_5 = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 3 & 1 \\ 1 & 1 & 2 \end{bmatrix}$

12.15 A random process is described by $x(n) = A\cos(n\omega_0 + \phi)$, where ϕ is a random variable uniformly distributed between $-\pi$ and π. Determine whether the process is near ergodic or not.
12.16 The power spectrum of a WSS process $x(n)$ is given by $P_x\left(e^{j\omega}\right) = \dfrac{25-24\cos\omega}{26-10\cos\omega}$. Find the whitening filter that produces unit variance white noise when the input is $x(n)$.
12.17 Determine the impulse response of the filter which generates a random process having power spectrum of the form
(i) $P_x\left(e^{j\omega}\right) = \dfrac{5+4\cos 2\omega}{10+6\cos\omega}$ and (ii) $P_x\left(e^{j\omega}\right) = \dfrac{5-4\cos 2\omega}{10-6\cos\omega}$
by filtering unit variance white noise.
12.18 State and prove the spectral factorisation theorem.

Chapter 13 Power Spectrum Estimation

INTRODUCTION 13.1

This chapter provides different methods that characterise the frequency content of a signal. All physical phenomena, whether it is electromagnetic, thermal, mechanical, and hydraulic or any other system has a unique spectrum associated with it. In a communication system, the received signal is always a random signal due to the fact that the transmitted signal is affected by a random noise. Such signals although cannot be described explicitly by mathematical relationship but may be characterised by certain statistical parameters. One of the most important parameters is the power spectral estimation that can be used for characterising both periodic and random signals. Power spectral estimation plays an important role in signal analysis, detection and recognition applications in the field of audio and speech, bio-signal, underwater acoustic, vibrations, astronomy, seismology, communication etc. In communication engineering, it is helpful in detecting signal component from noise. In radar and sonar, it is useful in detecting the targets.

ENERGY SPECTRAL DENSITY 13.2

13.2.1 Continuous Time Signal

Consider a complex-valued continuous time signal $x(t)$ and if $x(t)$ a finite energy signal, then, it satisfies the necessary condition as follows:

$$\int_{-\infty}^{\infty} |x(t)|^2 \, dt < \infty \tag{13.1}$$

Then, the Fourier transform (FT) of the signal $x(t)$ is given by

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \tag{13.2}$$

and the inverse Fourier transform (IFT) is given by

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df$$

By Parseval's relation,

$$\int_{-\infty}^{\infty} |x(t)|^2 \, dt = \int_{-\infty}^{\infty} |X(f)|^2 \, df \tag{13.3}$$

Equation (13.3) relates the total energy of the signal in time and frequency domain. The energy spectral density (ESD) is defined as the distribution of energy by the signal for a specific band of frequencies. For deterministic signals, the Fourier transform of the autocorrelation function gives energy spectral density $P_x(f)$ as

$$P_x(f) = \mathcal{F}\{r_{xx}(\tau)\} = \int_{-\infty}^{\infty} r_{xx}(\tau) \, e^{-j2\pi f\tau} d\tau \tag{13.4}$$

where $r_{xx}(\tau)$ be the autocorrelation function of the signal $x(t)$ for a lag τ, defined as

$$r_{xx}(\tau) = \int_{-\infty}^{\infty} x^*(t)\, x(t+\tau)\ \mathrm{d}t \tag{13.5}$$

Substituting $r_{xx}(\tau)$ in Eq.(13.4), we get

$$P_x(f) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x^*(t)\, x(t+\tau)\,\mathrm{d}t \right] e^{-j2\pi f\tau}\,\mathrm{d}\tau$$

$$= \int_{-\infty}^{\infty} x^*(t) \left[\int_{-\infty}^{\infty} x(t+\tau)\, e^{-j2\pi f\tau}\, d\tau \right] \mathrm{d}t = \int_{-\infty}^{\infty} x^*(t) X(f)\, e^{-j2\pi ft}\,\mathrm{d}t$$

$$= X(f) \left[\int_{-\infty}^{\infty} x^*(t)\, e^{-j2\pi ft}\,\mathrm{d}t \right] = X(f)\, X^*(f)$$

$$P_x(f) = |X(f)|^2 \tag{13.6}$$

Thus, the energy spectral density is described by the squared magnitude of the Fourier transform of the input signal.

13.2.2 Discrete Time Signal

Let the analog signal $x(t)$ be discretised by ideal periodic sampling function, which is called periodic impulse train with the sampling period of T_s. The resultant signal $x'(t)$ is represented by

$$x(t) = x(t) \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \tag{13.7}$$

The Fourier transform of the sampled signal $x'(t)$ is given by

$$X'(f) = \int_{-\infty}^{\infty} \left[x(t) \sum_{n=-\infty}^{\infty} \left[\delta(t - nT_s) \right] \right] e^{-j2\pi ft}\,\mathrm{d}t$$

$$= \sum_{n=-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} \left[\delta(t - nT_s) \right]\ \mathrm{d}t \right]$$

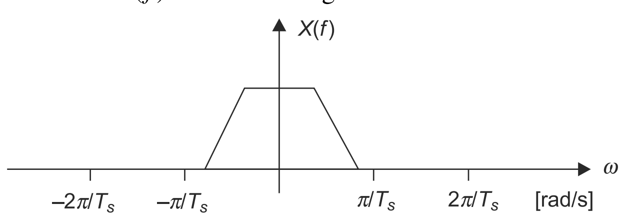
$$= \sum_{n=-\infty}^{\infty} x(nT_s) e^{-j2\pi fnT_s} \qquad \left[\text{since} \int_{-\infty}^{\infty} x(t)\delta(t - nT_s)\mathrm{d}t = x(nT_s) \right]$$

$$= f_s \sum_{l=-\infty}^{\infty} X(f - l f_s) \tag{13.8}$$

If aliasing is avoided, i.e. $x(t)$ is bandlimited to a frequency less than $1/2T_s$, then

$$X'(f) = f_s\, X(f) \text{ or } X(f) = T_s\, X'(f)$$

The graphical representation of $X(f)$ is shown in Fig. 13.1.



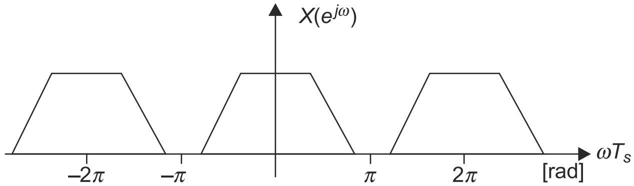


Fig. 13.1 *Spectrum of the sampled signal*

This means that the Fourier transform of the sampled signal is sum of the Fourier transform of the continuous time signal and its translated version scaled by $f_s = \frac{1}{T_s}$. The autocorrelation function $r_{xx}(k)$ of the discrete time signal $x(n)$ is given by

$$r_{xx}(k) = \sum_{n=-\infty}^{\infty} x^*(n)\, x(n+k) \tag{13.9}$$

The Fourier transform of $r_{xx}(k)$ gives the energy density spectrum

i.e.,
$$P_x(f) = \sum_{k=-\infty}^{\infty} r_{xx}(k)\, e^{-j2\pi kf}, \; -\infty \le f \le \infty \tag{13.10}$$

Another way of computing the energy density spectrum is from the Fourier transform of $x(n)$, given by

$$P_x(f) = |X(f)|^2$$
$$= \left| \sum_{n=-\infty}^{\infty} x(n)\, e^{-j2\pi fn} \right|^2 \tag{13.11}$$

Since finite energy signals possess Fourier transform, spectral analysis is done with the energy spectral density function. For signals which do not have finite energy, Fourier transform is not possible. But these signals have finite average power. For such signals, the quantity of interest is power spectral density function.

13.3 ESTIMATION OF THE AUTOCORRELATION AND POWER SPECTRUM OF RANDOM SIGNALS

A random signal or also referred to as random process is defined as a sequence of random variables indexed by time. It is characterised by a set of probability density function. Let $x(t)$ be a stationary random process. The statistical autocorrelation function for this signal is

$$r_{xx}(\tau) = E[x^*(t)\, x(t+\tau)] \tag{13.12}$$

where $E[.]$ denotes the expectation operator.

According to Wiener Khintchine theorem, the Fourier transform of the autocorrelation function of a stationary random process gives the power density spectrum,

$$P_x(f) = \mathcal{F}\left(r_{xx}(\tau)\right)$$
$$= \int_{-\infty}^{\infty} r_{xx}(\tau)\, e^{-j2\pi f\tau}\, d\tau \tag{13.13}$$

13.3.1 Estimate of Autocorrelation Function

In practice, only single realisation of the random process is considered, true autocorrelation function is not known and hence time average autocorrelation function is considered in the estimation process.

Let the observation interval be T_0. The time average autocorrelation function is $r_{xx}^T(\tau)$ can be defined as

$$r_{xx}^T(\tau) = \frac{1}{2T_0}\int_{-T_0}^{T_0} x^*(t)x(t+\tau)\,dt \tag{13.14}$$

The statistics or expectation of a stationary random process are not necessarily equal to the time averages. However, if a stationary random process whose statistics are equal to the time average and then the RP is said to be ergodic. If the random process is ergodic, then

$$\begin{aligned} r_{xx}(\tau) &= \underset{T_0\to\infty}{Lt}\; r_{xx}^T(\tau) \\ &= \underset{T_0\to\infty}{\text{Lt}}\; \frac{1}{2T_0}\int_{-T_0}^{T_0} x^*(t)\,x(t+\tau)\,\text{d}t \end{aligned} \tag{13.15}$$

The time average autocorrelation function is equal to the statistical autocorrelation function.

13.3.2 Estimation from Samples

Estimate 1 Let $x(n)$ be an N-point sequence of stationary obtained by sampling a continuous time random process. In practice, the autocorrelation function measures the self-similarity of a signal with the delayed version of the signal itself and it is estimated from an N-point sequence as

$$\hat{r}_{xx}(m) = \frac{1}{N}\sum_{n=0}^{N-|m|-1} x^*(n)\,x(n+m), \qquad 0 \le m \le N-1 \tag{13.16}$$

Mean and Variance of the Autocorrelation

Estimate 1 The mean value of the estimate $\hat{r}_{xx}(m)$ is

$$\begin{aligned} E\left[\hat{r}_{xx}(m)\right] &= \frac{1}{N}\sum_{n=0}^{N-|m|-1} E\left[x^*(n)x(n+m)\right] \\ &= \frac{N-|m|}{N}\hat{r}_{xx}(m) = \left(1-\frac{|m|}{N}\right)\hat{r}_{xx}(m) \end{aligned} \tag{13.17}$$

where $\frac{|m|}{N}\hat{r}_{xx}(m)$ is the bias for the estimate $\hat{r}_{xx}(m)$. The variance of the estimate is

$$\text{var}\left[\hat{r}_{xx}(m)\right] \approx \frac{1}{N}\sum_{n=-\infty}^{\infty}\left[\left|\hat{r}_{xx}(m)\right|^2 + \hat{r}_{xx}(n-m)\hat{r}_{xx}(n+m)\right] \tag{13.18}$$

As N becomes infinity, the variance value becomes zero. Hence $\hat{r}_{xx}(m)$ is an asymptotically unbiased estimate.

Estimate 2 If the value of m is large, then only fewer points are considered for the estimate. Hence, considering a different estimate for the autocorrelation function, the sample autocorrelation for the sequence $x(n)$ is

$$\hat{r}_{xx}(m) = \begin{cases} \dfrac{1}{N-|m|}\displaystyle\sum_{n=0}^{N-|m|-1} x^*(n)x(n+m) & |m| = 0,1,\dots,N-1 \\ \dfrac{1}{N-|m|}\displaystyle\sum_{n=|m|}^{N-1} x^*(n)x(n+m) & m = -(N-1)\dots-1 \end{cases} \tag{13.19}$$

The Fourier transform of the sample autocorrelation sequence is the estimate of the power spectrum or periodogram

$$\hat{P}_{xx}(f) = \sum_{m=-(N-1)}^{N-1} \hat{r}_{xx}(m)e^{-j2\pi fm} \tag{13.20}$$

Mean and Variance of the Autocorrelation Estimate 2 The mean value of the estimate $\hat{r}_{xx}(m)$ is obtained by taking mathematical expectation of $\hat{r}_{xx}(m)$.

$$E\left[\hat{r}_{xx}(m)\right] = \frac{1}{N-|m|}\sum_{n=0}^{N-|m|-1} E\left[x^*(n)x(n+m)\right]$$
$$= \hat{r}_{xx}(m), \qquad \text{when} \quad N \to \infty \tag{13.21}$$

Therefore $\hat{r}_{xx}(m)$ is an asymptotically unbiased estimate of the autocorrelation of the sequence $x(n)$. The approximate value of variance of the estimate $\hat{r}_{xx}(m)$ is given by Jenkins and Watts as

$$\text{var}\left[\hat{r}_{xx}(m)\right] = \frac{N}{\left(N-|m|\right)^2}\sum_{n=-\infty}^{\infty}\left[\left|\hat{r}_{xx}(m)\right|^2 + \hat{r}_{xx}(m)(n-m)\hat{r}_{xx}(m)(n+m)\right] \tag{13.22}$$

As N becomes infinity, the variance becomes zero, and this estimate $\hat{r}_{xx}(m)$ is consistent.

Estimate for Power Density Spectrum The estimate for the power density spectrum is

$$\hat{P}_x(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)e^{-j2\pi fn}\right|^2 \tag{13.23}$$

By substituting the value for the estimate for autocorrelation function $\hat{r}_{xx}(m)$, the power density spectrum is computed by,

$$\hat{P}_x(f) = \sum_{m=-(N-1)}^{N-1} \hat{r}_{xx}(m)e^{-j2\pi fm} \tag{13.24}$$

Equation (13.24) gives the estimate for power density spectrum is called the periodogram. Due to the finite length and randomness associated with signals, the power spectral density obtained from different records of a signal varies randomly about an average value of the power spectral density.

Mean and Variance of Periodogram Estimate The mean value of the periodogram estimate is calculated as

$$E\left[\hat{P}_x(f)\right] = \frac{1}{N}E\left[\left(\left|X(f)\right|^2\right)\right] \tag{13.25}$$

$$= \frac{1}{N}E\left[\sum_{m=0}^{N-1}\left\{x(m)e^{-j2\pi fm}\right\}\sum_{n=0}^{N-1}\left\{x(n)e^{-j2\pi fn}\right\}\right]$$

$$= \sum_{m=-(N-1)}^{N-1}\left(1-\frac{|m|}{N}\right)r_{xx}(m)e^{-j2\pi fm} \tag{13.26}$$

As $N \to \infty$, the mean value becomes

$$E\left[\hat{P}_x(f)\right] = \sum_{m=-\infty}^{\infty} r_{xx}(m)e^{-j2\pi fm} = P_x(f) \tag{13.27}$$

Therefore, the periodogram is said to be asymptotically unbiased.

In general, the estimation of a parameter (θ), bias and variance of this estimator is defined as

Bias: $B = E(\hat{\theta}) - \theta$

Variane: $\text{var} = E\{(\hat{\theta} - E(\hat{\theta})\}$

The variance of the periodogram can be evaluated using the above definition as

$$\text{var}\left[\hat{P}_x(f)\right] = P_x^2(f)\left[1+\left(\frac{\sin 2\pi f N}{N\sin 2\pi f}\right)^2\right] \tag{13.28}$$

As N increases, the estimated power spectrum converges to the true spectrum. Hence the estimate is an asymptotically unbiased estimate. But the variance does not become zero as N becomes infinity. Hence periodogram is not a consistent estimate of the power density spectrum.

The covariance of the periodogram estimate can be determined using the formula $\text{cov}\left[\hat{P}_x(f_1)\hat{P}_x(f_2)\right]=E\left[\hat{P}_x(f_1)\hat{P}_x(f_2)\right]-E\left[\hat{P}_x(f_1)\right]E\left[\hat{P}_x(f_2)\right]$. When $f_1 = f_2$, we get the variance of the periodogram.

From Eq.(13.26),

$$\hat{P}_x(f)=\frac{1}{N}\left|\sum_{n=0}^{N-1}x(n)e^{-j2\pi fn}\right|^2$$

$$=\frac{1}{N}\left[\sum_{n=0}^{N-1}x(n)e^{-j2\pi fn}\right]\left[\sum_{n=0}^{N-1}x(k)e^{-j2\pi fk}\right]^*$$

$$=\frac{1}{N}\left[\sum_{n=0}^{N-1}x(n)e^{-j2\pi fn}\right]\left[\sum_{n=0}^{N-1}x^*(k)e^{-j2\pi fk}\right]$$

$$=\frac{1}{N}\sum_{n=0}^{N-1}\sum_{k=0}^{N-1}x(n)x^*(k)e^{-j2\pi(n-k)f} \tag{13.29}$$

Therefore, the second-order moment of the periodogram is defined as

$$E\left[\hat{P}_x(f_1)\hat{P}_x(f_2)\right]=\frac{1}{N^2}\sum_{n=0}^{N-1}\sum_{k=0}^{N-1}\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}E\left[x(n)x^*(k)x(p)x^*(q)\right]e^{-j2\pi(n-k)f_1}.e^{-j2\pi(p-q)f_2} \tag{13.30}$$

If $x(n)$ is Gaussian, we may use the moment factoring theorem which can be stated as

$$E\left[x(n)x^*(k)x(p)x^*(q)\right]=E\left[x(n)x^*(k)\right]E\left[x(p)x^*(q)\right]+E\left[x(n)x^*(q)\right]E\left[x(p)x^*(k)\right] \tag{13.31}$$

The terms $E\left[x(n)x^*(k)\right]$ and $E\left[x(p)x^*(q)\right]$ are equal to σ_x^4 when $n = k$ and $p = q$ for white noise and zero otherwise. Substituting Eq.(13.31) into Eq.(13.30), we get two terms for the second-order moment of the periodogram.

The first term becomes

$$\frac{1}{N^2}\sum_{n=0}^{N-1}\sum_{p=0}^{N-1}\sigma_x^4=\sigma_x^4$$

The second term may further be simplified as

$$\frac{1}{N^2}\sum_{n=0}^{N-1}\sum_{k=0}^{N-1}\sigma_x^4 e^{-j2\pi(n-k)f_1}e^{-j2\pi(n-k)f_2}$$

$$=\frac{\sigma_x^4}{N^2}\sum_{n=0}^{N-1}e^{j2\pi(f_1-f_2)n}\sum_{k=0}^{N-1}e^{j2\pi(f_1-f_2)k}$$

$$=\frac{\sigma_x^4}{N^2}\left[\frac{1-e^{-j2\pi N(f_1-f_2)}}{1-e^{-j2\pi(f_1-f_2)}}\right]\left[\frac{1-e^{j2\pi N(f_1-f_2)}}{1-e^{j2\pi(f_1-f_2)}}\right]$$

$$=\frac{\sigma_x^4}{N^2}\left[\frac{1-e^{-j2\pi N(f_1-f_2)}-e^{j2\pi N(f_1-f_2)}+1}{1-e^{-j2\pi(f_1-f_2)}-e^{j2\pi(f_1-f_2)}+1}\right]=\frac{\sigma_x^4}{N^2}\left[\frac{2-2\cos 2\pi N\left(f_1-f_2\right)}{2-2\cos 2\pi\left(f_1-f_2\right)}\right]$$

$$= \frac{\sigma_x^4}{N^2}\left[\frac{1-\cos 2\pi N(f_1-f_2)}{1-\cos 2\pi(f_1-f_2)}\right]$$

$$= \frac{\sigma_x^4}{N^2}\left[\frac{2\sin^2 \pi N(f_1-f_2)}{2\sin^2 \pi(f_1-f_2)}\right] = \sigma_x^4\left[\frac{\sin \pi N(f_1-f_2)}{N\sin \pi(f_1-f_2)}\right]^2$$

Combining the two terms, we get

$$E\left[\hat{P}_x(f_1)\hat{P}_x(f_2)\right] = \sigma_x^4\left[1+\left(\frac{\sin^2 \pi N(f_1-f_2)}{N\sin^2 \pi(f_1-f_2)}\right)\right]$$

We know that,

$$\text{cov}\left[\hat{P}_x(f_1)\hat{P}_x(f_2)\right] = E\left[\hat{P}_x(f_1)\hat{P}_x(f_2)\right] - E\left[\hat{P}_x(f_1)\right]E\left[\hat{P}_x(f_2)\right]$$

Therefore, $\text{cov}\left[\hat{P}_x(f_1)\hat{P}_x(f_2)\right] = \sigma_x^4\left[\frac{\sin N\pi(f_1-f_2)}{N\sin\pi(f_1-f_2)}\right]$

When $f_1 = f_2$, we get the variance as

$$\text{var}\left[\hat{P}_x(f)\right] = P_x^2(f)$$

The variance of the periodogram $\hat{P}_x(f)$ is not equal to zero as $N \to \infty$. Hence, the periodogram is not a consistent estimate of the power spectrum. Therefore, methods to reduce the variance of the periodogram are discussed below.

The Modified Periodogram

From Eq.(13.23), we know that the periodogram is proportional to the squared magnitude of the Fourier transform of a finite length sequence or finite duration signal. Using rectangular window, $w_R(n)$, periodogram can be written as

$$\hat{P}_x(f) = \frac{1}{N}\left|\sum_{n=-\infty}^{\infty} x(n)w_R(n)e^{-j2\pi fn}\right|^2 \tag{13.32}$$

where $$w_R(n) = \begin{cases} 1 & 0 \le n \le N-1 \\ 0 & \text{otherwise} \end{cases}$$

Use of rectangular window produces the least amount of spectral smoothing. Instead of using a rectangular window to $x(n)$, other windowing techniques can be applied. Thus the periodogram of a process that is windowed with any general window $w(n)$ is called modified periodogram and is given by

$$\hat{P}_x(f) = \frac{1}{NU}\left|\sum_{n=-\infty}^{\infty} x(n)w(n)e^{-j2\pi fn}\right|^2 \tag{13.33}$$

where N is the length of the window. The normalisation factor is defined as

$U = \frac{1}{N}\sum_{n=0}^{N-1}|w(n)|^2$, which is a constant.

Example 13.1 *Compute the autocorrelation and power spectral density for the signal*

$$x(t) = K\cos(2\pi f_c t + \Phi)$$

where K and f_c are constants. Φ is a random variable which is uniformly distributed over the interval $(-\pi, \pi)$.

Solution Since Φ is a random variable that is uniformly distributed, the probability density function of the random variable Φ is

$$f(\Phi)=\begin{cases}\dfrac{1}{2\pi}, & -\pi\le\Phi\le\pi\\ 0, & \text{otherwise}\end{cases}$$

The autocorrelation function for the signal $x(t)$ is

$$\begin{aligned}r_{xx}(\tau)&=E\left[x(t+\tau)x(t)\right]\\&=E\left[K^2\cos\left(2\pi f_c t+2\pi f_c\tau+\Phi\right)\cos\left(2\pi f_c t+\Phi\right)\right]\\&=\frac{K^2}{2}E\left[\cos\left(4\pi f_c t+2\pi f_c\tau+2\Phi\right)\right]+\frac{K^2}{2}E\left[\cos\left(2\pi f_c\tau\right)\right]\\&=\frac{K^2}{2}\int_{-\pi}^{\pi}\frac{1}{2\pi}\cos\left(4\pi f_c t+2\pi f_c\tau+2\Phi\right)d\Phi+\frac{K^2}{2}\cos\left(2\pi f_c\tau\right)\end{aligned}$$

$$\left[\text{since } E(x)=\int_{-\infty}^{\infty}xf(x)dx \text{ and } E(\text{constant})=\text{constant}\right]$$

$$=\frac{K^2}{2}\cos\left(2\pi f_c\tau\right)$$

The autocorrelation function plot is shown in Fig. E13.1(a).

The power spectral density function is obtained by taking the Fourier transform of the autocorrelation function. Therefore,

$$\begin{aligned}\hat{P}_x(f)&=\mathcal{F}\left(r_{xx}(\tau)\right)\\&=\mathcal{F}\left[\frac{K^2}{2}\cos\left(2\pi f_c\tau\right)\right]\\&=\frac{K^2}{4}\left[\delta\left(f-f_c\right)+\delta\left(f+f_c\right)\right]\end{aligned}$$

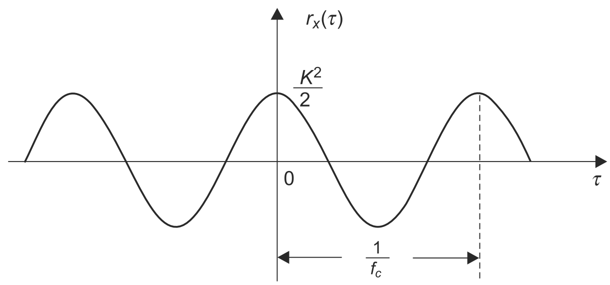


Fig. E13.1(a) *Autocorrelation of a sine wave with random phase*

The plot of power spectral density function is shown in Fig.E13.1(b).

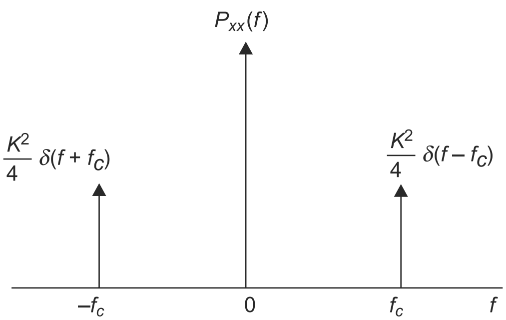


Fig. E.13.2(b) *Power spectral density of sine wave with random phase*

DFT IN SPECTRAL ESTIMATION 13.4

The periodogram is given by

$$\hat{P}_x(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)\, e^{-j2\pi fn}\right|^2 \tag{13.34}$$

The samples of the periodogram can be obtained by using *DFT* algorithm. Let $f = k/N$, where $k = 0, 1, 2, ...N - 1$. The periodogram becomes

$$\hat{P}_x\left(\frac{k}{N}\right) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)\, e^{-j2\pi nk/N}\right|^2 \tag{13.35}$$

where $k = 0, 1, ..., N - 1$.

If more samples are required in the frequency-domain, the length of the sequence $x(n)$ can be increased by zero padding. Letting the new length be L, the power spectral density is

$$\hat{P}_x\left(\frac{k}{L}\right) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)\, e^{-j2\pi nk/L}\right|^2$$

where $k = 0, 1, ..., L - 1$.

The periodogram does not increase the resolution but provides the interpolated values.

Example 13.2 *Consider the discrete time signal*

$$x(n) = \cos(2\pi f_1 n) + \cos(2\pi f_2 n), \quad n = 0.1,...,7$$

Obtain the power spectrum for the data sequence length L = 8, 16, 32 by considering the different values of f_1 and f_2, where $f_2 = f_1 + \Delta f$ and Δf is a small deviation from f_1 or simply frequency separation.

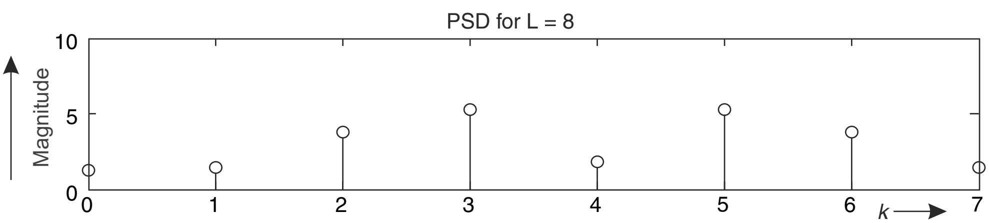
Solution The power spectrum for the discrete time signal $x(n)$ is

$$\hat{P}_x\left(\frac{k}{L}\right) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)\, e^{-j2\pi nk/L}\right|^2$$

Let $f_1 = 0.3$ and $\Delta f = 0.05$ which results $f_2 = 0.35$. Then

$$x(n) = \cos(2\pi(0.3)n) + \cos(2\pi(0.35)n)$$

The power spectrum is calculated for different values of L, by appending zero to the original sequence is shown in Fig. E.13.2(a).



(Contd.)

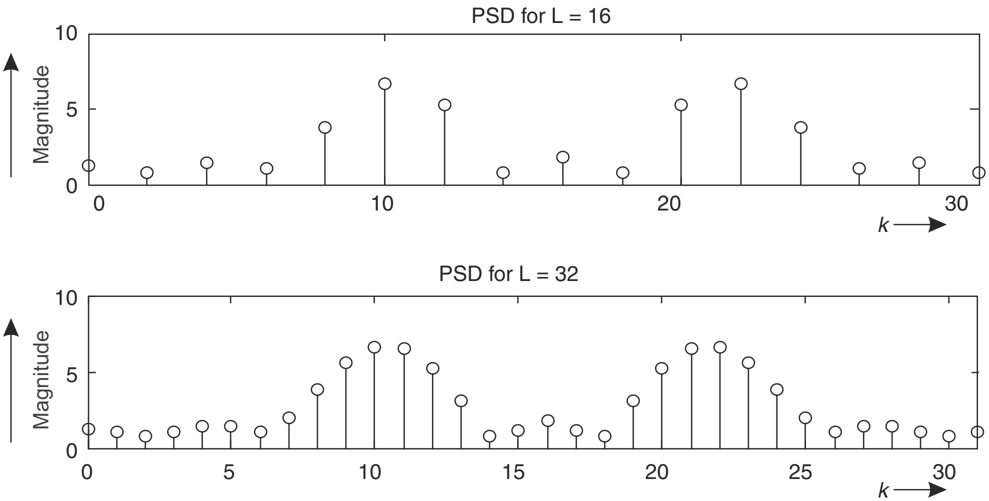


Fig. E.13.2(a) *Power Spectrum* $\frac{1}{N}\left[\left|X(k)\right|^2\right]$ *versus k for* $\omega_1 = 2\pi(0.3)$ *and* $\omega_2 = 2\pi(0.35)$

Let $\Delta f = 0.02$ and $f_1 = 0.3$; hence $f_2 = 0.32$. Then

$$x(n) = \cos(2\pi(0.3)n) + \cos(2\pi(0.32)n)$$

The power spectrum of $x(n)$ for various values of L is shown in Fig. E13.2(b).

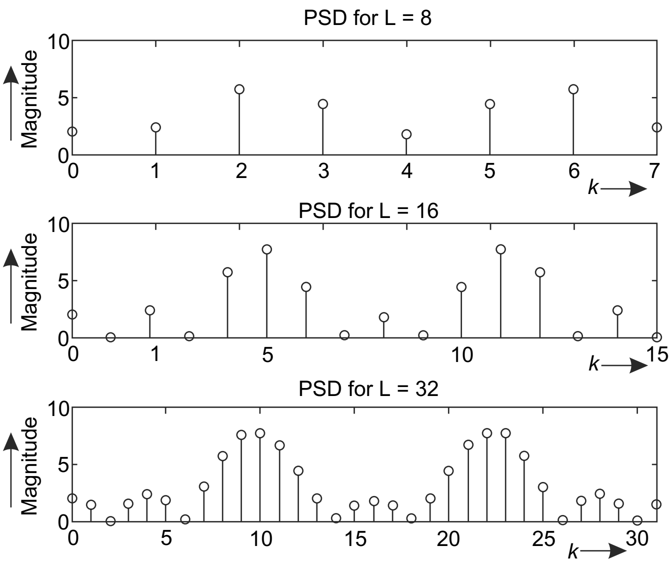


Fig. E13.2(b) *Power Spectrum* $1/N\,|X(k)|^2$ *Versus k for* $\omega_1 = 2\pi(0.3)$ *and* $\omega_2 = 2\pi(0.32)$

It is clear from Fig. 13.2(a) and (b) that when Δf is very small, the spectral components are not resolvable. The effect of zero padding is to provide more interpolation and not to improve the frequency resolution.

Example 13.3 *Estimate the power spectrum of the sampled data sequence {2,3,1,1,2,1,0,1}. Assume the Sampling rate as 1000 Hz.*

Solution Let $x(n) = \{2, 3, 1, 1, 2, 1, 0, 1\}$

Power Spectrum = $\frac{1}{N}|X(k)|^2$

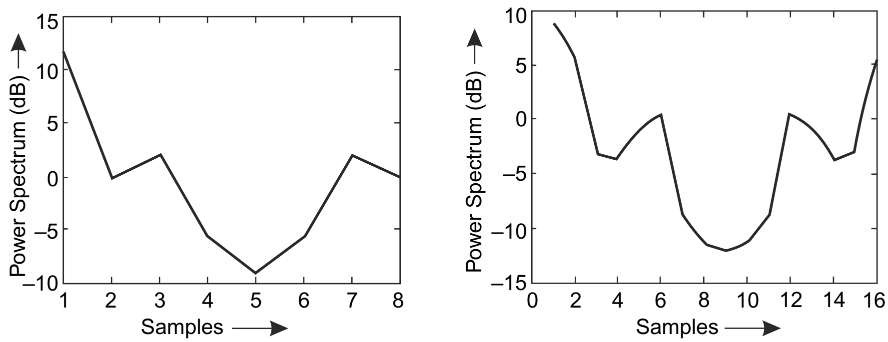


Fig. E13.3 *Power Spectrum Magnitude of x(n)*

POWER SPECTRUM ESTIMATION: NON-PARAMETRIC METHODS 13.5

The estimation techniques based on non-parametric methods decrease the frequency resolution in reducing the variance of the spectral estimate. These methods estimate the spectrum from finite number of noisy measurements and make no assumption about how the data were generated and hence are called non-parametric.

13.5.1 Averaging Periodogram—The Bartlett Method

Consider the N-point sequence $x(n)$ and divide it into K non-overlapping sequences each of length $M = N / K$. The i^{th} subsequence is given by

$$x_i(n) = x(n + iM), \quad i = 0,1,..., \quad K-1, \quad n = 0,1,... \ M-1 \tag{13.36}$$

The periodogram for each subsequence can be computed as

$$\hat{P}_x^{(i)}(f) = \frac{1}{M}\left|\sum_{n=0}^{M-1} x_i(n) e^{-j2\pi fn}\right|^2, \qquad i = 0,1,..., \ K-1 \tag{13.37}$$

The Bartlett power spectral estimate can be obtained by averaging the periodogram of the K subsequences, which is computed as

$$\hat{P}_x^B(f) = \frac{1}{K}\sum_{i=0}^{k-1} \hat{P}_x^{(i)}(f) = \frac{1}{K}\sum_{i=0}^{k-1} \frac{1}{M}\left|\sum_{n=0}^{M-1} x_i(n) e^{-j2\pi fn}\right|^2 \tag{13.38}$$

$$= \frac{1}{N}\sum_{n=0}^{M-1} \left|\sum_{n=0}^{M-1} x_i(n) e^{-j2\pi fn}\right|^2$$

Mean The mean value of the Bartlett estimate is computed as

$$E\left[\hat{P}_x^{(B)}(f)\right] = \frac{1}{K}\sum_{i=0}^{K-1} E\left[\hat{P}_x^{(i)}(f)\right]$$

$$= E\left[\hat{P}_x^{(i)}(f)\right] \qquad (13.39)$$

Expected value for the single periodogram is computed as

$$E\left[\hat{P}_x^{(i)}(f)\right] = \sum_{m=-(M-1)}^{M-1}\left(1-\frac{m}{|M|}\right) r_x(m) e^{-j2\pi fm} \qquad (13.40)$$

The above expression can be viewed as a windowing process as

$$E\left[\hat{P}_x^{(i)}(f)\right] = \sum_{m=-(M-1)}^{M-1} W_{Bart}(m) r_x(m) e^{-j2\pi fm}$$

where,
$$W_{Bart}(m) = \begin{cases} 1-\dfrac{|m|}{M} & |m| \le M-1 \\ 0, & \text{otherwise} \end{cases} \qquad (13.41)$$

is known as Bartlett window.The frequency-domain representation for Bartlett window is

$$W_{Bart}(f) = \frac{1}{M}\left(\frac{\sin \pi f M}{\sin \pi f}\right)^2 \qquad (13.42)$$

If the data sequence length is reduced from N to M, the spectral width is increased by a factor $K = N / M$ and the frequency resolution is reduced by a factor K.

Variance By reducing the resolution, the variance of power spectral estimate is also reduced by a factor K.

$$\text{var}\left[\hat{P}_x^{B}(f)\right] = \frac{1}{K^2}\sum_{i=0}^{K-1}\text{var}\left[\hat{P}_x^{(i)}(f)\right]$$

$$= \frac{1}{K}\text{var}\left[\hat{P}_x^{(i)}(f)\right]$$

$$= \frac{1}{K}\hat{P}_x^2(f)\left[1+\left(\frac{\sin 2\pi f M}{M \sin 2\pi f}\right)^2\right] \qquad (13.43)$$

Thus, if N tends to infinity, both K and M are allowed to go to infinity and the variance goes to zero. Hence, the Bartlett estimate is a consistent estimate of the power spectrum.

A simple and straight forward approach to reduce the variance is as follows:

- Divide the N-point sequence $x(n)$ into K non-overlap subsequences of length M.
- Find the periodogram for each subsequence.
- Calculate the average of K subsequence periodograms.

13.5.2 Averaging Modified Periodogram—Welch Method

The following two modifications were made by Welch in 1967 in the averaging periodogram or Bartlett method.

(i) The subsequences of $x(n)$ are allowed to overlap.

(ii) A data window $w(n)$ other than rectangular window is applied to each subsequence in computing the periodogram.

In the Welch method, the overlapping subsequence are represented by

$$x_i(n) = x(n+iD), \quad n = 0,1,..., M-1, \quad i = 0,1,..., K-1 \tag{13.44}$$

where successive sequences are offset by 'D' points and "iD" is the starting index of the subsequence. If $D = M$, then this is same as the Bartlett method. The periodogram for each subsequence is computed as

$$\hat{P}_x^{(i)}(f) = \frac{1}{MU}\left|\sum_{n=0}^{M-1} x_i(n)w(n)e^{-j2\pi fn}\right|^2, \quad i = 0,1,..,K-1 \tag{13.45}$$

where U is the normalisation factor and is computed as

$$U = \frac{1}{M}\sum_{n=0}^{M-1}|w(n)|^2$$

The Welch power spectrum estimate is

$$\hat{P}_x^{w}(f) = \frac{1}{K}\sum_{i=0}^{K-1}\hat{P}_x^{(i)}(f) \tag{13.46}$$

$$\hat{P}_x^{w}(f) = \frac{1}{K.M.U}\left|\sum_{n=0}^{M-1} x_i(n)w(n)e^{-j2\pi fn}\right|^2, \quad i = 0,1,..,K-1$$

Mean The expected value of the Welch estimate is given by

$$E\left[\hat{P}_x^{w}(f)\right] = E\left(\frac{1}{K}\sum_{i=0}^{K-1}\left[\hat{P}_x^{(i)}(f)\right]\right) = \frac{1}{K}\sum_{i=0}^{K-1}E\left[\hat{P}_x^{(i)}(f)\right]$$

$$= E\left[\hat{P}_x^{(i)}(f)\right] \tag{13.47}$$

Expected value of the modified periodogram is given by

$$E\left[\hat{P}_x^{(i)}(f)\right] = \frac{1}{MU}\sum_{n=0}^{M-1}\sum_{m=0}^{M-1} w(n)w(m)E\left[x_i(n)x_i(m)\right]e^{-j2\pi f(n-m)}$$

$$= \frac{1}{MU}\sum_{n=0}^{M-1}\sum_{m=0}^{M-1} w(n)w(m)r_x(n-m)\,e^{-j2\pi f(n-m)}$$

$$= \frac{1}{MU}\int_{-1/2}^{1/2} P_x(\alpha)\left[\sum_{n=0}^{M-1}\sum_{m=0}^{M-1} w(n)w(m)e^{-j2\pi(f-\alpha)(n-m)}\right]d\alpha$$

$$= \int_{-1/2}^{1/2} P_x(\alpha)W(f-\alpha)\,d\alpha \tag{13.48}$$

where $$W(f) = \frac{1}{MU}\left|\sum_{n=0}^{M-1} w(n)e^{-j2\pi fn}\right|^2$$

and the normalisation factor U ensures that $\int_{-1/2}^{1/2} W(f)\,\mathrm{d}f = 1$

Variance The variance of the Welch power spectrum estimate is more difficult to compute since it is dependent on the overlap, window type and number of data segments. The variance of the PSD estimate can be defined as

$$\text{var}\left[\hat{P}_x^w(f)\right] = \frac{1}{L^2}\sum_{i=0}^{L-1}\sum_{j=0}^{L-1} E\left[\hat{P}_x^{(i)}(f)\hat{P}_x^{(i)}(f)\right] - \left\{E\left[\hat{P}_x^w(f)\right]\right\}^2 \tag{13.49}$$

For 50% overlap and Bartlett window,

$$\text{var}\{P_x^w(f)\} = \frac{9}{8K}P_x^2(e^{j\omega})$$

Expressing the variance in terms of M and N, we get

$$\text{var}\left[\hat{P}_x^w(f)\right] = \frac{9}{16}\frac{M}{N}P_x^2 \tag{13.50}$$

Thus, the variance can be decreased by decreasing M where $M = \frac{N}{K}$ or by increasing K. However, this increases the correlation between successive sequences resulting in poor resolution.

13.5.3 Smoothing the Periodogram-Blackman and Tukey Method

Smoothing the periodogram method was proposed by Blackman and Tukey. The autocorrelation sequence is windowed before calculating the power spectral density. Windowing is used because it decreases the contribution of unreliable estimate to the periodogram. The estimated autocorrelation is given by

$$\hat{r}_{xx}(m) = \frac{1}{N}\sum_{n=0}^{N-|m|-1} x(n)x(n+m)$$

The estimate is worst when m is largest because the average is computed for the fewer number of terms. For example, consider the best and worst case when $m = 0$ and $m = N - 1$,

$$\hat{r}_{xx}(0) = \frac{1}{N}\sum_{n=0}^{N-|m|-1} x(n)x(n+m) = \frac{1}{N}x^2(0) + \frac{1}{N}x^2(1) + \cdots + \frac{1}{N}x^2(N-1)$$

$$\hat{r}_{xx}(N-1) = \frac{1}{N}\sum_{n=0}^{0} x(n)x(n+N-1) = \frac{1}{N}x(0)x(N-1)$$

The $\hat{r}_{xx}(0)$ is based on the average of N terms. In the worst case $\hat{r}_{xx}(N-1)$ is based on only one term. Therefore irrespective of number of terms, the variance of $\hat{r}_{xx}(m)$ exists. The Blackman–Tukey estimate is used to improve the estimated power spectrum by applying a window $W_{BT}(m)$ to $\hat{r}_{xx}(m)$ before applying *DTFT*. Mathematically,

$$\hat{P}_x^{BT}(f) = \sum_{m=-(M-1)}^{M-1} \hat{r}_{xx}(m)w_{BT}(m)e^{-j2\pi fm} \tag{13.51}$$

where $w(m)$ is the window function with length $2M - 1$, which is also known as *lag window*. The lag window tapers away from the centre. The effect of multiplication by lag window is convolution of signal's *PSD* with the windows *PSD*. Therefore,

$$\hat{P}_x^{BT}(f) = \frac{1}{2\pi}\hat{P}_x(f) * W_{BT}(f) \tag{13.52}$$

where $\hat{P}_x(f)$ is the periodogram. The lag window should be symmetric about $m = 0$ and the window spectrum $W_{BT}(f)$ should be non-negative.

Mean Frequency-domain expected value of the Blackman–Tukey estimate is given by

$$E\left[\hat{P}_x^{BT}(f)\right] = E\left[\frac{1}{2\pi}\hat{P}_x(f) * W_{BT}(f)\right]$$

$$= E\left[\frac{1}{2\pi}W_{BT}(f) * W_{\Delta}(f) * P_x(f)\right] \tag{13.53}$$

$$= E\left[\frac{1}{2\pi} W_{BT\Delta}(f) * P_x(f)\right]$$

where $W_{BT\Delta}(f) = W_{BT}(f) * W_{\Delta}(f)$

As $N \to \infty, \frac{1}{2\pi} W_{BT\Delta}(f)$ converges to an impulse. Therefore, $E\left[\hat{P}_x^{BT}(f)\right] = P_x(f)$. Thus the Blackman–Tukey estimate is asymptotically unbiased.

Variance The variance of the Blackman–Tukey estimate is given by

$$\text{var}\left[\hat{P}_x^{BT}(f)\right] = E\left[\hat{P}_x^{BT}(f)^2\right] - \left\{E\left[\hat{P}_x^{BT}(f)\right]\right\}^2$$

$$E\left[\left(\hat{P}_x^{BT}(f)\right)^2\right] = \int_{-1/2}^{1/2}\int_{-1/2}^{1/2} E\left[\hat{P}_x(\alpha)\hat{P}_x(\Phi)\right] W(f-\alpha) W(f-\Phi)\,\mathrm{d}\alpha\mathrm{d}\Phi$$

$$E\left[\hat{P}_x(\alpha)\hat{P}_x(\Phi)\right] = P_x(\alpha)P_x(\Phi)\left\{1 + \left[\frac{\sin\pi(\Phi+\alpha)N}{N\sin\pi(\Phi+\alpha)}\right]^2 + \left[\frac{\sin\pi(\Phi-\alpha)N}{N\sin\pi(\Phi-\alpha)}\right]^2\right\}$$

$$E\left\{\left[\hat{P}_x^{BT}(f)\right]\right\} = \int_{-1/2}^{1/2}\int_{-1/2}^{1/2} P_x(\alpha)P_x(\Phi)$$

$$\left\{1 + \left[\frac{\sin\pi(\Phi+\alpha)N}{N\sin\pi(\Phi+\alpha)}\right] + \left[\frac{\sin\pi(\Phi+\alpha)N}{N\sin\pi(\Phi+\alpha)}\right]^2\right\} W(f-\alpha)W(f-\Phi)\,\mathrm{d}\alpha\mathrm{d}\Phi$$

$$= \left[\int_{-1/2}^{1/2} P_x(\Phi)W(f-\Phi)\,\mathrm{d}\Phi\right]^2 + \int_{-1/2}^{1/2}\int_{-1/2}^{1/2} P_x(\alpha)P_x(\Phi)W(f-\alpha)W(f-\Phi)$$

$$\left\{\left[\frac{\sin\pi(\Phi+\alpha)N}{N\sin\pi(\Phi+\alpha)}\right]^2 + \left[\frac{\sin\pi(\Phi-\alpha)N}{N\sin\pi(\Phi-\alpha)}\right]^2\right\}\mathrm{d}\alpha\,\mathrm{d}\Phi$$

$$\approx \left\{E\left[P_x^{BT}(f)\right]\right\}^2 + \frac{1}{N}\int_{-1/2}^{1/2} P_x(\alpha)W(f-\alpha)\left[P_x(-\alpha)W(f+\alpha) + P_x(\alpha)W(f-\alpha)\right]\mathrm{d}\alpha$$

$$\text{var}\left[P_x^{\text{BT}}(f)\right] \approx \frac{1}{N}\int_{-1/2}^{1/2} P_x(\alpha)W(f-\alpha)\left[P_x(-\alpha)W(f+\alpha) + P_x(\alpha)W(f-\alpha)\right]\mathrm{d}\alpha$$

Since for $N >> M$,

$$\int_{-1/2}^{1/2} P_x(\Phi)W(f-\Phi)\left\{\left[\frac{\sin\pi(\Phi+\alpha)N}{N\sin\pi(\Phi+\alpha)}\right]^2 + \left[\frac{\sin\pi(\Phi-\alpha)N}{N\sin\pi(\Phi-\alpha)}\right]^2\right\}\mathrm{d}\Phi$$

$$\approx \frac{P_x(-\alpha)W(f+\alpha) + P_x(\alpha)W(f-\alpha)}{N}$$

As $\left[\frac{\sin\pi(\Phi+\alpha)N}{N\sin\pi(\Phi+\alpha)}\right]^2$ and $\left[\frac{\sin\pi(\Phi-\alpha)N}{N\sin\pi(\Phi-\alpha)}\right]$ are narrow compared to $W(f)$ around $\Phi = -\alpha$ and $\Phi = \alpha$,

$$\int_{-1/2}^{1/2} P_x(\alpha)P_x(-\alpha)W(f-\alpha)W(f+\alpha)\,\mathrm{d}\alpha \approx 0$$

Therefore, $$\mathrm{var}\left[P_x^{BT}(f)\right] \approx \frac{1}{N}\int_{-1/2}^{1/2} P_x^2(\alpha)W^2(f-\alpha)\,\mathrm{d}\alpha$$

If $W(f)$ is narrow when compared with $P_x(f)$, then

$$\mathrm{var}\left[P_x^{\mathrm{BT}}(f)\right] \approx P_x^2(f)\left[\frac{1}{N}\int_{-1/2}^{1/2} W^2(\Phi)\,\mathrm{d}\Phi\right]$$

$$\approx P_x^2(f)\left[\frac{1}{N}\sum_{m=-(M-1)}^{M-1} W^2(m)\right] \tag{13.54}$$

13.5.4 Quality of Power Spectrum Estimators

The quality of the estimator is given by the ratio of the square of the mean of the power spectrum estimate to its variance, i.e.

$$Q_A = \frac{\left\{E\left[P_{xx}^A(f)\right]\right\}^2}{\mathrm{var}\left[P_{xx}^A(f)\right]} \tag{13.55}$$

Variability which is another measure of performance is the reciprocal of quality. One more measure of quality is the overall Figure of Merit, which is defined as

Figure of Merit = *Variability* × *Resolution*

In this section, the quality of periodogram using Bartlett, Welch and Balackman–Tukey power spectrum estimation techniques is calculated.

Periodogram In Section 13.3, it was shown that the periodogram is asymptotically unbiased, and for large values of N, variance is approximately equal to true power spectral density. Therefore,

$$Q_A = \frac{\left\{E\left[\hat{P}_x(f)\right]\right\}^2}{\mathrm{var}\left[\hat{P}_x(f)\right]} = 1 \tag{13.56}$$

Hence, the variability is equal to 1. The resolution of the periodogram $2\pi\,\Delta f = 2\pi\frac{0.89}{N}$. Then the overall figure of merit is equal to $\frac{0.89}{N}$.

Bartlett Power Spectrum Estimate

In Bartlett method, the variance is reduced by averaging periodogram. As $N = KM$, for large values of N, the variance equal to $\frac{1}{K}\hat{P}_x^2(f)$. The quality factor of averaging periodogram is

$$Q_{\mathrm{Bart}} = K = \frac{N}{M}$$

With 3 dB main lobe width of a rectangular window, the frequency resolution is $\Delta f = \frac{0.9}{M}$.

Therefore, $$M = \frac{0.9}{\Delta f} \tag{13.57}$$

and $$Q_{\mathrm{Bart}} = \frac{N}{0.9/\Delta f} = 1.1N\Delta f$$

Then the overall figure of merit is equal to $2\pi\frac{0.89}{N}$.

Welch Power Spectrum Estimate

The Quality factor of Averaging Modified Periodogram is

$$Q_w = \frac{8L}{9} = \frac{16N}{9M} \text{ for 50\% overlap with triangular window.} \tag{13.58}$$

with spectral width of triangular window at 3 dB points,

$$\Delta f = \frac{1.28}{M}$$

$Q_w = 1.39\, N\, \Delta f$, for 50% overlap with triangular window

Then the overall figure of merit is equal to $2\pi \frac{0.72}{N}$.

Blackman–Tukey Power Spectrum Estimate

In this method, the variance and resolution are dependent on the window used. The variance for triangular window is

$$\operatorname{var}\left[\hat{P}_x^{BT}(f)\right] \approx P_x^2(f)\left[\frac{1}{N}\sum_{m=-(M-1)}^{M-1} W^2(m)\right] = P_x^2(f)\left[\frac{1}{N}\sum_{m=-(M-1)}^{M-1}\left(1-\frac{|m|}{M}\right)^2\right] = P_x^2(f)\frac{2M}{3N}$$

Quality factor is $Q_{BT} = 1.5\frac{N}{M}$

Window length = $2M - 1$, Frequency resolution at 3 dB points is

$$\Delta f = \frac{1.28}{2M} = \frac{0.64}{M} \tag{13.59}$$

Hence, $$Q_{BT} = \frac{1.5}{0.64} N\Delta f = 2.34 N\Delta f$$

Quality factor increases when N is increased. For a desired quality level, Δf is decreased by increasing N.

Then the overall figure of merit is equal to $2\pi \frac{0.43}{N}$.

Computational Requirements

The following assumptions are made in the computation of the power spectrum:

1. Fixed data length = N
2. Frequency resolution = Δf
3. Radix 2 FFT algorithm

Bartlett Power Spectrum Estimate

$$\text{FFT length } = M = \frac{0.9}{\Delta f}$$

$$\text{Number of FFTs } = \frac{N}{M} = 1.11 N\Delta f$$

$$\text{Number of computations } = \frac{N}{M}\left(\frac{M}{2}\log_2 M\right) = \frac{N}{2}\log_2 \frac{0.9}{\Delta f}$$

Welch Power Spectrum Estimate (50% overlap)

$$\text{FFT length } = M\frac{1.28}{\Delta f}$$

Number of FFTs $= \dfrac{2N}{M} = 1.56N\Delta f$

Number of computations $= \dfrac{2N}{M}\left(\dfrac{M}{2}\log_2 M\right) = N\log_2\left(\dfrac{1.28}{\Delta f}\right)$

For windowing, each data record requires M multiplication $\left(\dfrac{2N}{M}\times M = 2N\right)$

Hence, the total computations $2N + N\log_2\dfrac{1.28}{\Delta f} = N\log_2\dfrac{5.12}{\Delta f}$

Blackman–Tukey Power Spectrum Estimate

The autocorrelation sequence $r_{xx}(m)$ can be computed using FFT. For large data points, FFT can be done by segmenting the data into $K = \dfrac{N}{2M}$ segments [windowing to $(2M - 1)$ points].

Using the approach, FFT length $= 2M = \dfrac{1.28}{\Delta f}$

Number of FFTs $= 2K + 1 = 2\left(\dfrac{N}{M}\right) + 1$

Number of computations $= \dfrac{N}{M}\left(M\log_2 2M\right) = N\log_2\dfrac{1.28}{\Delta f}$

13.5.5 Limitations of Non-Parametric Methods for Power Spectrum Estimation

(i) The non-parametric methods require long data sequences to obtain the necessary frequency resolution.

(ii) The non-parametric methods introduce spectral leakage effects because of windowing.

(iii) The assumption of the autocorrelation estimate $\hat{r}_x(m)$ to be zero for $m \geq N$ limits the frequency resolution and quality of the power spectrum estimate.

(iv) Assumption that the sequence is periodic with period N, is not realistic.

Example 13.4 *Show that*

(a) $$E\left[P_x(f_1)P_x(f_2)\right] = \sigma_x^4\left\{1 + \left[\frac{\sin\pi(f_1+f_2)N}{N\sin\pi(f_1+f_2)}\right]^2 + \left[\frac{\sin\pi(f_1-f_2)N}{N\sin\pi(f_1-f_2)}\right]^2\right\}$$

and (b) $$\operatorname{var}\left[P_x(f)\right] = \sigma_x^4\left\{1 + \left(\frac{\sin 2\pi f N}{N\sin 2\pi f}\right)^2\right\}.$$

Use the expression for the fourth joint moment for Gaussian random variable.

$$E\left[X_1X_2X_3X_4\right] = E\left[X_1X_2\right]E\left[X_3X_4\right] + E\left[X_1X_3\right]E\left[X_2X_4\right] + E\left[X_1X_4\right]E\left[X_2X_3\right]$$

Solution

(a) $$E\left[P_x(f_1)P_x(f_2)\right] = E\left[\frac{1}{N^2}\left|\sum_{n_1=0}^{N-1} x(n_1)e^{-j2\pi f_1 n_1}\right|^2\left|\sum_{n_2=0}^{N-1} x(n_2)e^{-j2\pi f_2 n_2}\right|^2\right]$$

$$= \frac{1}{N^2}\sum_{n_1=0}^{N-1}\sum_{n_2=0}^{N-1}\sum_{n_3=0}^{N-1}\sum_{n_4=0}^{N-1} E\left[x(n_1)x(n_2)x(n_3)x(n_4)\right]\left(e^{-j2\pi f_1(n_1-n_3)}\right)$$

But, $E\left[x(n_1)x(n_2)x(n_3)x(n_4)\right] = E\left[x(n_1)x(n_2)\right]E\left[x(n_3)x(n_4)\right] + E\left[x(n_1)x(n_3)\right]$
$$E\left[x(n_2)x(n_4)\right] + E\left[x(n_1)x(n_4)\right]E\left[x(n_2)x(n_3)\right]$$
$$= \sigma_x^4$$

$n_1 = n_2$ and $n_3 = n_4; n_1 = n_3$ and $n_2 = n_4; n_1 = n_4$ and $n_2 = n_3$

For $n_1 = n_4$ and $n_2 = n_4$,

$$e^{-2\pi f_1(n_1-n_3)}e^{-j2\pi f_2(n_2-n_4)} = e^{-j2\pi f_1(n_1-n_3)}e^{-j2\pi f_2(n_3-n_1)}$$
$$= e^{-j2\pi f_1(n_1-n_3)}e^{+j2\pi f_2(n_1-n_3)} = e^{-j2\pi(f_1-f_2)(n_1-n_3)}$$

For $n_1 = n_2$ and $n_3 = n_4$,

$$e^{-j2\pi f_1(n_1-n_3)}e^{-j2\pi f_2(n_2-n_4)} = e^{-j2\pi f_1(n_1-n_3)}e^{-j2\pi f_2(n_1-n_3)}$$
$$= e^{-j2\pi(f_1+f_2)(n_1-n_3)}$$

$$= \frac{1}{N^2}\sum_{n_1=0}^{N-1}\sum_{n_2=0}^{N-1}\sum_{n_3=0}^{N-1}\sum_{n_4=0}^{N-1}\left\{\begin{matrix} E\left[x(n_1)x(n_2)\right]E\left[x(n_3)x(n_4)\right]+ \\ E\left[x(n_1)x(n_3)\right]E\left[x(n_2)x(n_4)\right]+ \\ E\left[x(n_1)x(n_4)\right]E\left[x(n_2)x(n_3)\right] \end{matrix}\right\}\left\{\begin{matrix} e^{-j2\pi f(n_1-n_3)x_1} \\ e^{-j2\pi f_2(n_2-n_4)} \end{matrix}\right\}$$

$$= \frac{1}{N^2}\left\{\begin{matrix} \sum_{n_1=0}^{N-1}\sum_{n_2=0}^{N-1}\sigma_x^4 + \sum_{n_1=0}^{n-1}\sum_{n_3=0}^{N-1}\sigma_x^4 e^{-j2\pi(f_1-f_2)(n_1-n_3)} \\ +\sum_{n_1=0}^{N-1}\sum_{n_3=0}^{N-1}\sigma_x^4 e^{-j2\pi(f_1+f_2)(n_1-n_3)} \end{matrix}\right\}$$

$$= \frac{\sigma_x^4}{N^2}\left\{\begin{matrix} N^2 + \sum_{n_1=0}^{N-1}\sigma_x^4 e^{-j2\pi(f_1-f_2)n_1}\sum_{n_3=0}^{N-1}e^{j2\pi(f_1-f_2)n_3} \\ +\sum_{n_1=0}^{N-1}e^{-j2\pi(f_1+f_2)n_1}\sum_{n_3=0}^{N-1}e^{j2\pi(f_1+f_2)n_3} \end{matrix}\right\}$$

$$= \sigma_x^4\left\{\begin{matrix} 1+\dfrac{1-e^{-j2\pi(f_1-f_2)N}}{1-e^{-j2\pi(f_1-f_2)}}\times\dfrac{1-e^{j2\pi(f_1-f_2)N}}{1-e^{j2\pi(f_1-f_2)}}\times\dfrac{1}{N^2} \\ +\dfrac{1-e^{-j2\pi(f_1+f_2)N}}{1-e^{-j2\pi(f_1+f_2)}}\times\dfrac{1-e^{j2\pi(f_1+f_2)N}}{1-e^{j2\pi(f_1+f_2)}}\times\dfrac{1}{N^2} \end{matrix}\right\}$$

$$= \sigma_x^4\left\{\begin{matrix} 1+\dfrac{2-e^{-j2\pi(f_1-f_2)N}-e^{j2\pi(f_1-f_2)N}}{2-e^{-j2\pi(f_1-f_2)}-e^{j2\pi(f_1-f_2)}}\times\dfrac{1}{N^2} \\ +\dfrac{2-e^{-j2\pi(f_1+f_2)N}-e^{j2\pi(f_1+f_2)N}}{2-e^{-j2\pi(f_1+f_2)}-e^{j2\pi(f_1+f_2)}}\times\dfrac{1}{N^2} \end{matrix}\right\}$$

$$= \sigma_x^4\left\{1+\frac{1}{N^2}\left(\frac{2-2\cos 2\pi(f_1-f_2)N}{2-2\cos 2\pi(f_1-f_2)}\right)+\frac{1}{N^2}\left(\frac{2-2\cos 2\pi(f_1+f_2)N}{2-2\cos 2\pi(f_1+f_2)}\right)\right\}$$

$$= \sigma_x^4 \left\{ 1 + \left(\frac{\sin \pi (f_1 - f_2) N}{N \sin \pi (f_1 - f_2)} \right)^2 + \left(\frac{\sin \pi (f_1 + f_2) N}{N \sin \pi (f_1 + f_2)} \right)^2 \right\}$$

(b) $\text{var}\left[P_x(f)\right] = E\left[P_x^2(f)\right] - \left\{E\left[P_x(f)\right]\right\}^2$

But, $E\left[P_x(f)\right] = \sigma_x^2$

$$E\left[P_x^2(f)\right] = \frac{1}{N^2} \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} \sum_{n_3=0}^{N-1} \sum_{n_4=0}^{N-1} E\left[x(n_1)x(n_2)x(n_3)x(n_4)\right] e^{-j2\pi f(n_1 - n_2 + n_3 - n_4)}$$

and

$$E\left[x(n_1)x(n_2)x(n_3)x(n_4)\right] = E\left[x(n_1)x(n_2)\right]E\left[x(n_3)x(n_4)\right] + E\left[x(n_1)x(n_3)\right] E\left[x(n_2)x(n_4)\right] + E\left[x(n_1)x(n_4)\right]E\left[x(n_2)x(n_3)\right]$$

$$= \sigma_x^4$$

$n_1 = n_2$ and $n_3 = n_4$; $n_1 = n_3$ and $n_2 = n_4$; $n_1 = n_4$ and $n_2 = n_3$

Similar to part (a), the summation can be expanded and the value is

$$E\left[P_x^2(f)\right] = \frac{\sigma_x^4}{N^2} \left\{ 2N^2 + \left(\frac{\sin 2\pi f N}{N \sin 2\pi f} \right)^2 \right\}$$

$$\text{var}\left[P_x(f)\right] = \frac{\sigma_x^4}{N^2} \left\{ 2N^2 + \left(\frac{\sin 2\pi f N}{N \sin 2\pi f} \right)^2 \right\} - \sigma_x^4 = \sigma_x^4 \left\{ 1 + \left(\frac{\sin 2\pi f N}{N \sin 2\pi f} \right)^2 \right\}$$

Example 13.5 *Bartlett Method is used to estimate the power spectral density of a process sequence of N = 1000. What is the minimum length M of the subsequence that may be used to have resolution of Δf = 0.005. Why is it not desired to increase the length beyond the minimum length found above?*

Solution

(a) Since $\Delta f = 0.9/M$ then $M = \dfrac{0.89}{\Delta f} = \dfrac{0.9}{0.005} = 180$

(b) Increasing M will increase the resolution but it will also result in a decrease in the number of segments that may be averaged. This, in turn, will increase the variance of the spectrum estimate.

Example 13.6 *Determine the frequency resolution of the Bartlett, Welch and Blackman–Tukey methods of power spectrum estimates for a quality factor Q = 10. Assume that overlap in Welch's method is 50% and the length of the sample sequence in 1000.*

Solution Given Quality factor $Q = 10$;
Length of the sample sequence (N) = 1000
Overlap in Welch's method = 50%

Bartlett Method

$Q_{\text{Bart}} = 1.11 N \Delta f \Rightarrow$ Frequency resolution $\Delta f = \dfrac{Q_{\text{Bart}}}{1.11(N)} = \dfrac{10}{1.11(1000)} = 0.009$

Welch Method

$Q_w = 1.39 N \Delta f \Rightarrow$ Frequency resolution $\Delta f = \dfrac{Q_w}{1.39(N)} = \dfrac{10}{1390} = 0.0072$

Blackman–Tukey Method

$$Q_w = 2.34N\Delta f \Rightarrow \text{Frequency resolution } \Delta f = \frac{Q_{BT}}{2.34(N)} = \frac{10}{2340} = 0.0042.$$

Example 13.7 *Apply the Welch modified periodogram method to obtain the power spectrum of the data {0, 1, 0, 1, 0, 1, 0, 1} by subdividing the data sequence into three equal-length segments with 50% overlap.*

Solution $x(n) = \{0, 1, 0, 1, 0, 1, 0, 1\}$

The 50% overlap segments are

$$\begin{cases} x_1(n) = (0,1,0,1) \\ x_2(n) = (0,1,0,1) \\ x_3(n) = (0,1,0,1) \end{cases}$$

Modified periodogram estimate is

$$\hat{P}_x^{(i)}(f) = \frac{1}{MU}\left|\sum_{n=0}^{M-1} x_i(n)w(m)e^{-j2\pi fn}\right|^2, \qquad i = 0,1,\ldots L-1$$

where the number of segments $L = 3$, segment length $M = 4$ and U is the normalisation factor. If a rectangular window function

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

is used, then $U = \frac{1}{M}\sum_{n=0}^{M-1} w^2(n) = 1$

The Welch power estimate is

$$P_x^w(f) = \frac{1}{L}\sum_{i=0}^{L-1} P_x^{(i)}(f)$$

The Welch power spectrum (power per frequency) for the sequence $x(n) = \{0, 1, 0, 1, 0, 1, 0, 1\}$ is shown in Fig. E13.7.

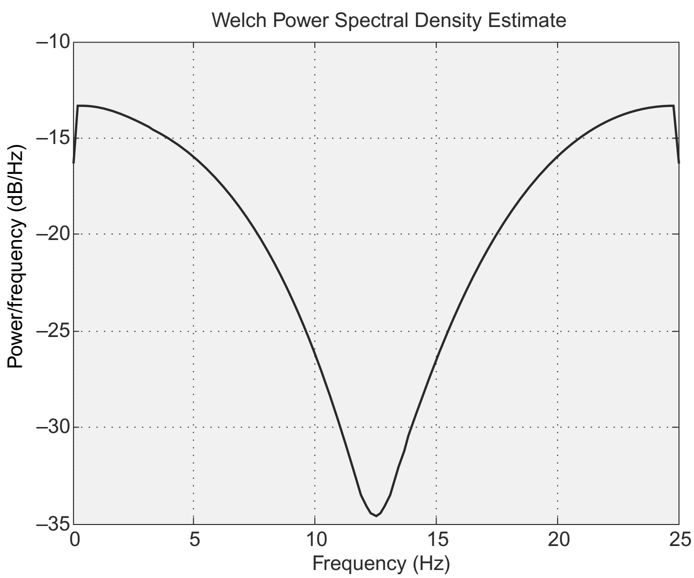


Fig. E13.7 *Welch PSD Estimate*

Example 13.8 *The data sequence {0, 1, 0, 1, 0, 1, 0, 1} is now assumed to contain a random noise component such that the new data sequence becomes {0.763, 1.656, 0.424, 1.939, 0.133, 1.881, 0.328, 1.348}. Calculate the power spectrum of this noisy data. Estimate the signal to noise ratio from the sampled data.*

Solution New data sequence, $x(n) = \{0.763, 1.656, 0.424, 1.939, 0.133, 1.881, 0.328, 1.348\}$

$$x_1(n) = \{0.763, 1.656, 0.424, 1.939\}$$
$$x_2(n) = \{0.424, 1.939, 0.133, 1.881\}$$
$$x_3(n) = \{0.133, 1.881, 0.328, 1.348\}$$

Modified periodogram estimate is

$$\hat{P}_x^{(i)}(f) = \frac{1}{MU}\left|\sum_{n=0}^{M-1} x_i(n)w(m)e^{-j2\pi fn}\right|^2, \qquad i = 0,1,...L-1$$

where U is the normalisation factor, which is given by

$$U = \frac{1}{M}\sum_{n=0}^{M-1} w^2(n)$$

The Welch power estimate is

$$P_x^w(f) = \frac{1}{L}\sum_{i=0}^{L-1} P_x^{(i)}(f)$$

Welch power spectrum (power per frequency) for the new sequence
$x(n) = \{0.763, 1.656, 0.424, 1.939, 0.133, 1.881, 0.328, 1.348\}$ is shown in Fig. E13.8.

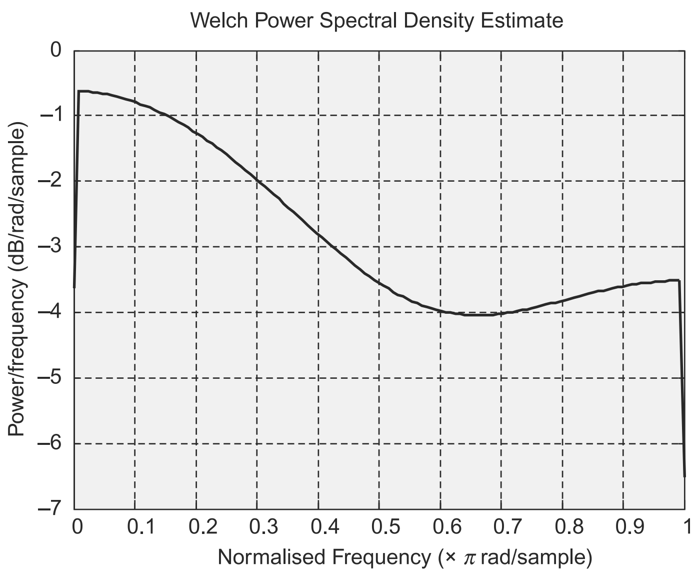


Fig. E13.8 *Welch PSD Estimate*

13.6 AR, MA AND ARMA MODELS

Signal modelling is an important problem that arises in many applications including signal compression and signal prediction. Let $x(n)$ be the observed signal which can be modelled as the output of a linear system represented by

$$H(z) = \frac{\sum_{k=0}^{q} b_k z^{-k}}{1 + \sum_{k=1}^{p} a_k z^{-k}} = \frac{X(z)}{W(z)} \tag{13.60}$$

where q refers to the number of zeros and p refers to number of poles.

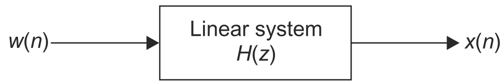


Fig. 13.2 *Linear system for analysing parametric models*

The difference equation of this system is

$$x(n) = -\sum_{k=1}^{p} a_k x(n-k) + \sum_{k=0}^{q} b_k w(n-k) \tag{13.61}$$

where $w(n)$ is the input sequence and the set of a_k and b_k are parameters of the filter coefficients that is included in the signal model shown in Fig. 13.2.

In the linear system model shown in Fig. 13.2, it is assumed that the observed data $x(n)$ is modelled as a stationary random process, and the system is excited by stationary random process with power spectral density σ_w^2. Let $P_w(f)$ be the power spectral density of the input sequence $w(n)$. Let $P_x(f)$ be the power spectral density of the output sequence $x(n)$ and $H(f)$ be the frequency response of the linear system, then

$$P_x(f) = |H(f)|^2 P_w(f) \tag{13.62}$$

If $w(n)$ is zero mean white noise sequence with variance σ_w^2, the autocorrelation sequence is

$$r_w(m) = \sigma_w^2 \, \delta(m) \tag{13.63}$$

The power spectral density of the input sequence $w(n)$ is

$$P_w(f) = \sigma_w^2$$

Hence, the power of spectral density of the output sequence $x(n)$ is

$$\begin{aligned} P_x(f) &= |H(f)|^2 P_w(f) \\ &= |H(f)|^2 \sigma_w^2 \end{aligned} \tag{13.64}$$

The process modelled by such a linear system defined by Eq.(13.60) is the ARMA process (Auto Regressive Moving Average) of order (p, q). It is represented as ARMA (p, q) which is also referred to as the pole-zero model.

If $b_0 = 1$ and $b_k = 0$ for $1 \le k \le q$, then

$$H(z) = \frac{1}{1 + \sum_{k=1}^{p} a_k z^{-k}} \tag{13.65}$$

The model for this system is called *AR* process of order p. It is an all pole model and is represented as *AR (p)*. If $a_k = 0$ for $1 \le k \le p$, then $H(z) = \sum_{k=0}^{q} b_k z^{-k}$ and then the model is moving average process of order q. It is an all zero model and is represented as *MA (q)*.

ARMA Process The output power spectral density for *ARMA* (*p, q*) process is given by

$$P_x(f) = |H(f)|^2 P_w(f)$$

$$= \sigma_w^2 \frac{\left|\sum_{k=0}^{q} b_k e^{-j2\pi fk}\right|^2}{\left|1+\sum_{k=1}^{p} a_k e^{-j2\pi fk}\right|^2} \tag{13.66}$$

The power spectral density depends only on the white noise variance and the filter coefficients. The filter coefficients are to be estimated. If the observed N data points are greater than $p + q + 1$ then a good estimate for the unknown parameters can be obtained.

Consider the general ARMA *(p,q)* model represented by the difference equation,

$$x(n) = -\sum_{k=1}^{p} a_k x(n-k) + \sum_{k=0}^{q} b_k w(n-k) \tag{13.67}$$

Let $b_0 = 1$ and $w(n)$ be a zero mean white noise process with variance σ_w^2.

Multiplying both sides by $x^*(n-m)$ and taking the expectation, we get

$$E\left[x(n)x^*(n-m)\right] = -\sum_{k=1}^{p} a_k E\left[x(n-k)x^*(n-m)\right] + \sum_{k=0}^{q} b_k E\left[w(n-k)x^*(n-m)\right]$$

$$r_{xx}(m) = -\sum_{k=1}^{p} a_k r_x(m-k) + \sum_{k=0}^{q} b_k r_{wx}(m-k) \tag{13.68}$$

The cross-correlation function $r_{wx}(m)$ is

$$r_{wx}(m) = E\left[x^*(n)w(n+m)\right]$$

From Fig. 13.6, $x(n)$ is written as

$$x(n) = \sum_{k=0}^{\infty} h(k)w(n-k)$$

Multiplying both sides by $w(n + m)$ and taking the expectation,

$$r_{wx}(m) = E\left[\sum_{k=0}^{\infty} h(k)w^*(n-k)w(n+m)\right]$$

$$= \sum_{k=0}^{\infty} h(k)E\left[w^*(n-k)w(n+m)\right]$$

$$= \sigma_w^2 h(-m) \tag{13.69}$$

Since $E\left[w^*(n)w(n+m)\right] = \sigma_w^2 \delta(m)$

$$r_{wx}(m) = \begin{cases} 0, & m > 0 \\ \sigma_w^2 h(-m), & m \le 0 \end{cases} \tag{13.70}$$

Therefore,

$$r_{xx}(m) = \begin{cases} \sum_{k=1}^{p} a_k r_x(m-k), & m > q \\ \sum_{k=1}^{p} a_k r_x(m-k) + \sigma_w^2 \sum_{k=0}^{q-m} h(k) b_{k+m}, & 0 \le m \le q \end{cases} \tag{13.71}$$

AR Process

A special type of *ARMA* process results when $q = 0$ in Eq.(13.60). In this case, $x(n)$ is modelled as the output of the linear system represented by

$$H(z) = \frac{b_0}{1 + \sum_{k=1}^{p} a_k z^{-k}}$$

If $q = 0$ and $b_0 = 1$ from Eq.(13.71), the $AR(p)$ model parameters can be obtained as

$$r_{xx}(m) = \begin{cases} \sum_{k=1}^{p} a_k r_{xx}(m-k), & m > 0 \\ \sum_{k=1}^{p} a_k r_{xx}(m-k) + \sigma_w^2, & m = 0 \end{cases} \tag{13.72}$$

When > 0, $r_{xx}(m)$, $m = 1, 2 \ldots P$, can be written in matrix form as

$$\begin{bmatrix} r_{xx}(0) & r_{xx}(-1) & \cdots & r_{xx}(-p+1) \\ r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(-p+2) \\ \cdots & \cdots & \cdots & \cdots \\ r_{xx}(p-1) & r_{xx}(p-2) & \cdots & r_{xx}(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} r_{xx}(1) \\ r_{xx}(2) \\ \vdots \\ r_{xx}(p) \end{bmatrix} \tag{13.73}$$

$$\sigma_\omega^2 = r_x(0) + \sum_{k=1}^{p} a_k r_x(-k) \tag{13.74}$$

Combining Eq. (13.73) and Eq.(13.74), we get

$$\begin{bmatrix} r_{xx}(0) & r_{xx}(-1) & \cdots & r_{xx}(-p) \\ r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(-p+1) \\ \vdots & \vdots & \vdots & \vdots \\ r_{xx}(p) & r_{xx}(p-1) & \cdots & r_{xx}(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \sigma_w^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

The correlation matrix is Toeplitz and non-singular and can be solved with the Levinson-Durbin algorithm for obtaining the inverse matrix.

MA Process

Another special case of *ARMA* process is the *MA* process which results when $p = 0$. If $p = 0$, $x(n)$ is generated by filtering white noise with an FIR filter that has a system function given by

$$H(z) = \sum_{k=0}^{q} b_k z^{-k}$$

If $a_k = 0$, for $k > 0$ from Eq.(13.67) the $MA(q)$ model is obtained as

$$r_{xx}(m) = \begin{cases} \sigma_w^2 \sum_{k=0}^{q} b_k b_{k+m}, & 0 \le m \le q \\ 0, & m > q \end{cases} \tag{13.75}$$

POWER SPECTRUM ESTIMATION: PARAMETRIC METHODS 13.7

Parametric methods provide better frequency resolution, since this modelling does not require window functions and the assumption that autocorrelation sequence to be zero for $|m| \geq N$ is not required. It extrapolates the values for $|m| \geq N$. But it requires prior information about the generation of data sequence. A model for the signal generation can be obtained from the observed data. These methods are useful for data sequences which are short.

The parametric spectral estimation is a three-step process:

1. Selection of the model
2. Estimate the model parameters from the observed/measured data or the correlation sequence estimated from the data.
3. Obtain the spectral estimate with the help of the estimated model parameters.

13.7.1 AR Model (Yule-Walker Method)

An estimate of the autocorrelation of the observed data is obtained first and then the *AR* parameters are calculated. Let the autocorrelation estimate (biased) be,

$$\hat{r}_{xx}(m) = \frac{1}{N}\sum_{n=0}^{N-|m|} x^*(n)x(n+m), \quad m \geq 0 \tag{13.76}$$

The autocorrelation matrix should be positive semi-definite to yield a stable *AR* model. The power spectrum estimate is

$$P_x^{yw}(f) = \frac{\sigma_{wp}^2}{\left|1+\sum_{k=1}^{p} a_p(k)e^{-j2\pi fk}\right|^2} \tag{13.77}$$

where $a_p(k)$ is the estimate of the *AR* parameters from Levinson Durbin algorithm and σ_{wp}^2 is the estimated minimum mean square value for the p^{th} order predictor which is given by

$$\sigma_{wp}{}^2 = \hat{r}_{xx}(0)\prod_{k=1}^{p}\left(1-|a_p(k)|^2\right)$$

MATLAB program for computing spectrum estimate of AR process is included in Appendix.

The Power spectrum estimated using autocorrelation method of an AR Model linear system of increasing order is shown in Fig. 13.3. From the figures, it can be observed that the spectral resolution increases with the increase in the number of poles. Usually the order of the model is increased until the modelling error is minimised.

MA Model

The MA(q) model for the observed data is represented by

$$\hat{r}_{xx}(m) = \begin{cases} \sigma_w^2 \sum_{k=0}^{q} b_k b_{k+m}, & 0 \leq m \leq q \\ 0, & m > q \end{cases} \tag{13.78}$$

The power spectral estimate is

$$\hat{P}_x^{MA}(f) = \sum_{m=-q}^{q} r_{xx}(m)e^{-j2\pi fm} = |B(z)|^2 \tag{13.79}$$

Hence, MA parameters $\{b_k\}$ need not be calculated, but an estimate of autocorrelation for $|m| \leq q$ will suffice.

ARMA Model

For certain types of processes, the ARMA model is used to estimate the spectrum with fewer parameters. This model is mostly used when the data is corrupted by noise.

Construct a set of linear equations for $m > q$ and use the method of least squares on the set of equations. Using the linear prediction,

$$r_{xx}(m) = -\sum_{k=1}^{p} a_k r_{xx}(m-k), m = q+1, q+2, \ldots, M < N \tag{13.80}$$

where $r_{xx}(m)$ is the estimated autocorrelation sequence. Model parameters $\{a_k\}$ are selected such that the squared error is minimised.

$$\varepsilon = \sum_{k=q+1}^{M} |e(n)|^2 = \sum_{k=q+1}^{M} \left| r_{xx}(m) + \sum_{k=1}^{p} a_k r_{xx}(m-k) \right|^2 \tag{13.81}$$

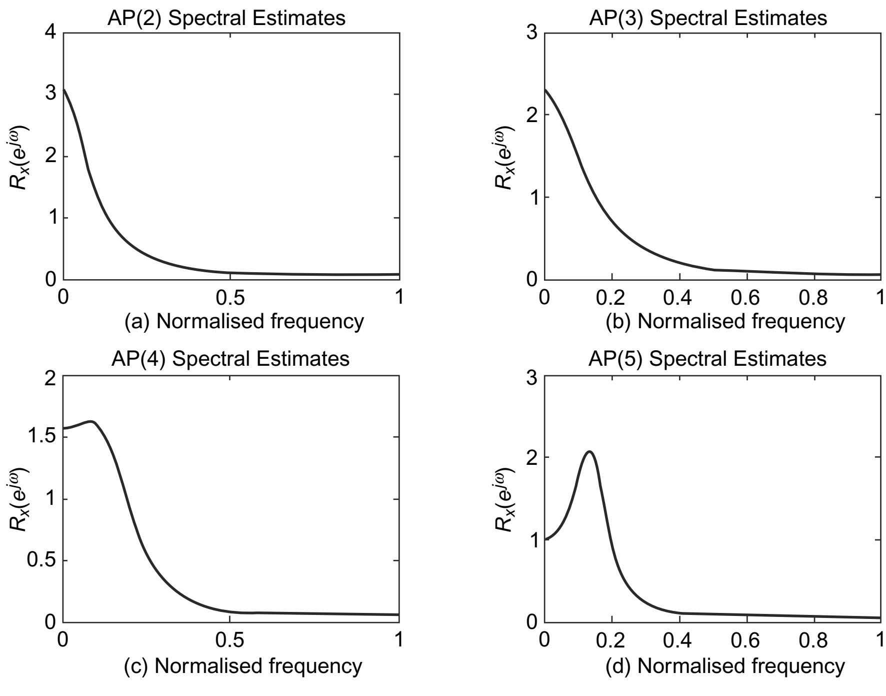


Fig. 13.3 *Spectrum Estimation of an AR Model Linear System of (a) Second Order (b) Third Order (c) Fourth Order, and (d) Fifth Order*

Thus, the AR parameters can be obtained and the procedure is known as the least squares modified Yule-Walker method. Suitable weightage can also be applied to the autocorrelation sequence.

After finding the AR model parameters, the system obtained is represented by

$$A(z) = \sum_{k=1}^{p} a_k z^{-k}$$

Consider the sequence $x(n)$ to be filtered by *FIR* filter, i.e. *AIR* model is $A'(z)$. Let the output be $y(n)$.

$$y(n) = x(n) + \sum_{k=1}^{p} a_k x(n-k), \qquad n = 0,1,\ldots,N-1$$

MA model $B(z)$ can be obtained by cascading *ARMA* (p, q) model with $A(z)$.

The sequence $y(n)$ for $p \le n \le N-1$ is used to obtain the correlation sequence $r_y(m)$. The MA power spectral estimate is

$$\hat{P}_y^{MA}(f) = \sum_{m=-q}^{p} r_y(m) e^{-j2\pi fm} \tag{13.82}$$

The ARMA power spectral estimate is

$$\hat{P}_x^{ARMA}(f) = \frac{\hat{P}_y^{MA}(f)}{\left|1 + \sum_{k=1}^{p} a'_k e^{-j2\pi fk}\right|^2} \tag{13.83}$$

Among the three models, AR model is widely used for power spectrum estimation due to

(i) *AR* model is suitable for representing power spectra with narrow peaks.

(ii) *AR* model is represented by simple linear equations.

The power spectrum in Fig. 13.4 has been estimated for models with higher orders. It can be observed from the above estimations that the spectrum is inherent with spurious peaks as displayed in (d) which ultimately results in spectral splitting which is undesirable.

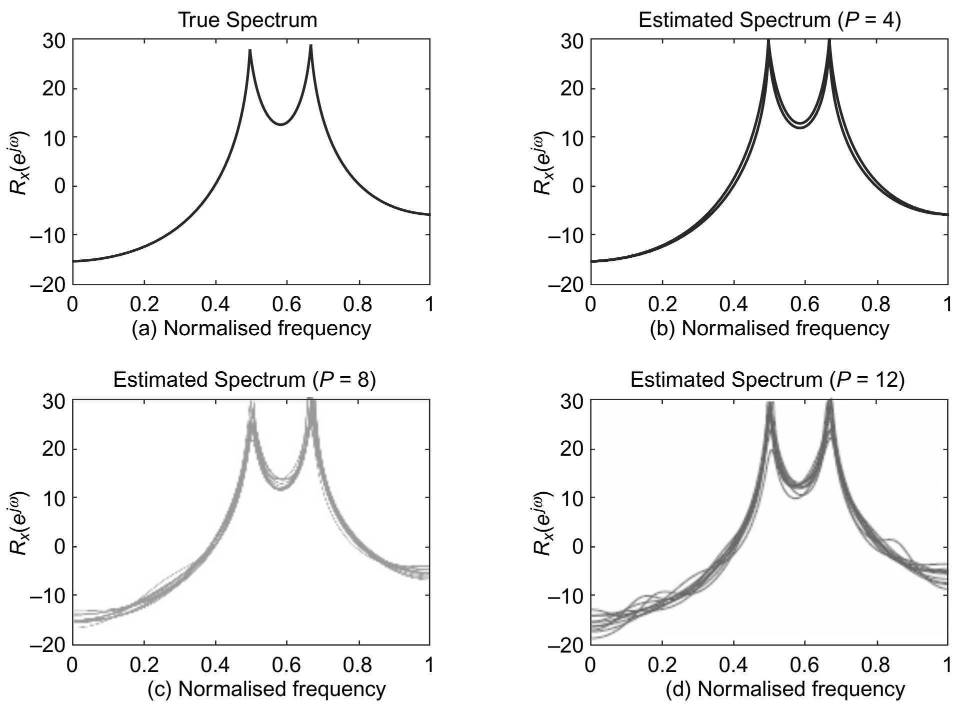


Fig. 13.4 *Power spectrum estimation of the ARMA model where (a) true spectrum (b) spectrum for order 4 (c) spectrum for order 8, and (d) spectrum for order 12*

THE LEVINSON-DURBIN RECURSION 13.8

A recursive algorithm to solve a general set of linear symmetric Toeplitz equations $Ra = b$ was developed by *N* Levinson. This recursion was useful to discover the lattice filter structure, which has found wide applications in speech processing, spectrum estimation, and digital filter implementations. Then Durbin improved the Levinson recursion for the special case in which the right-hand side of the Toeplitz equations is a unit vector. The *Levinson-Durbin recursion* is discussed in this section.

For computing the inverse of **R**, Levinson-Durbin algorithm is less complex when compared to the Gauss elimination method. In the Gauss elimination method, the computational complexity is proportional to N^3 whereas in Levinson Durbin algorithm, it is N^2. The set of equations are called normal equations or the Yule-Walker equations. These equations are solved recursively by using Levinson-Durbin Algorithm.

13.8.1 Development of the Recursion

All-pole model derived from Prony's method or the autocorrelation method requires that we solve the normal equations which, for a *p*th-order model, are

$$r_{xx}(k)+\sum_{l=1}^{p} a_p(l) r_{xx}(k-1)=0; \qquad k=1,2,\ldots p$$

where the modelling error is

$$\xi_p = r_{xx}(0)+\sum_{l=1}^{p} a_p(l) r_{xx}(l)$$

Combining Eq.(13.81) and Eq.(13.82) into matrix form, we have

$$\begin{bmatrix} r_{xx}(0) & r_{xx}^*(1) & r_{xx}^*(2) & \cdots & r_{xx}^*(p) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}^*(1) & \cdots & r_{xx}^*(p-1) \\ r_{xx}(2) & r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}^*(p-2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ r_{xx}(p) & r_{xx}(p-1) & r_{xx}(p-2) & \cdots & r_{xx}(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_p(1) \\ a_p(2) \\ \vdots \\ a_p(p) \end{bmatrix} = \xi_p \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{13.84}$$

which is a set of $(p + 1)$ linear equations in the $(p + 1)$ unknowns $a_p(1)$, $a_p(2)$, ..., $a_p(p)$ and ε_p. Equivalently, Eq.(13.83) may be written as

$$\mathbf{R}_p \mathbf{a}_p = \xi_p \mathbf{U}_1$$

where $\mathbf{R}_p$ is a $(p + 1) \times (p + 1)$ Hermitian Toeplitz matrix, $\mathbf{U}_1$ is a unit vector with 1 in the first position.

The Levinson-Durbin recursion for solving Eq.(13.84) is an algorithm that is recursive in the model order. In other words, the coefficients of the $(j + 1)$ st-order all-pole model, $\mathbf{a}_{j+1}$, are found from the coefficients of the *j*-pole model, $\mathbf{a}_j$. The solution to the *j*th-order normal equations may be used to derive the solution to the $(j + 1)$ st-order equations. Let $\mathbf{a}_j(i)$ be the solution to the *j*th-order normal equations.

$$\begin{bmatrix} r_{xx}(0) & r_{xx}^*(1) & r_{xx}^*(2) & \cdots & r_{xx}^*(j) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}^*(1) & \cdots & r_{xx}^*(j-1) \\ r_{xx}(2) & r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}^*(j-2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ r_{xx}(j) & r_{xx}(j-1) & r_{xx}(j-2) & \cdots & r_{xx}(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_j(1) \\ a_j(2) \\ \vdots \\ a_j(j) \end{bmatrix} = \begin{bmatrix} \xi_j \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{13.85}$$

In matrix notation,

$$\mathbf{R}_j\mathbf{a}_j = \xi_j\mathbf{U}_1 \tag{13.86}$$

Given $\mathbf{a}_j$, the solution to the $(j + 1)$ st-order normal equations is derived.

$$\mathbf{R}_{j+1}\mathbf{a}_{j+1} = \xi_{j+1}\mathbf{U}_1 \tag{13.87}$$

The procedure for doing this is as follows. Suppose that we append a zero to the vector $\mathbf{a}_j$ and multiply the resulting vector by $\mathbf{R}_{j+1}$. The result is

$$\begin{bmatrix} r_{xx}(0) & r_{xx}^*(1) & r_{xx}^*(1) & \cdots & r_{xx}^*(1) & r_{xx}^*(j+1) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}^*(1) & \cdots & r_{xx}(j-1) & r_{xx}^*(j) \\ r_{xx}(0) & r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(j-2) & r_{xx}^*(j-1) \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ r_{xx}(j) & r_{xx}(j-1) & r_{xx}(j-2) & \cdots & r_{xx}(0) & r_{xx}^*(1) \\ r_{xx}(j+1) & r_{xx}(j) & r_{xx}(j-1) & \cdots & r_{xx}(1) & r_{xx}(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_j(1) \\ a_j(2) \\ \vdots \\ a_j(j) \\ 0 \end{bmatrix} = \begin{bmatrix} \xi_j \\ 0 \\ 0 \\ \vdots \\ 0 \\ \gamma_j \end{bmatrix} \tag{13.88}$$

where the parameter γ_j is

$$\gamma_j = r_{xx}(j+1) + \sum_{i=1}^{j} a_j(i) r_{xx}(j+1-i) \tag{13.89}$$

Note that, if $\gamma_j = 0$, then the right side of Eq.(13.88) is a scaled unit vector and $\mathbf{a}_{j+1} = [1, a_j(1), ..., a_j(j), 0]^T$ is the solution to the $(j + 1)$ st-order normal Eq.(13.87). $\gamma_j \neq 0$ and $[1, a_j(1), ..., a_j(j), 0]^T$ is not the solution to Eq.(13.86).

The key step in the derivation of the Levinson-Durbin recursion is to note that the Hermitian Toeplitz property of $\mathbf{R}_{j+1}$ allows us to rewrite Eq.(13.88) in the equivalent form

$$\begin{bmatrix} r_{xx}(0) & r_{xx}(1) & r_{xx}(2) & \cdots & r_{xx}(j) & r_{xx}(j+1) \\ r_{xx}^*(1) & r_{xx}(0) & r_{xx}(1) & \cdots & r_{xx}(j-1) & r_{xx}(j) \\ r_{xx}^*(2) & r_{xx}^*(1) & r_{xx}(0) & \cdots & r_{xx}(j-2) & r_{xx}(j-1) \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ r_{xx}^*(j) & r_{xx}^*(j-1) & r_{xx}^*(j-2) & \cdots & r_{xx}(0) & r_{xx}(1) \\ r_{xx}^*(j+1) & r_{xx}^*(j) & r_{xx}^*(j-1) & \cdots & r_{xx}^*(1) & r_{xx}(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_j(j) \\ a_j(j-1) \\ \vdots \\ a_j(1) \\ 1 \end{bmatrix} = \begin{bmatrix} \gamma_j \\ 0 \\ 0 \\ \vdots \\ 0 \\ \xi_j \end{bmatrix} \tag{13.90}$$

Taking the complex conjugate of Eqs. (13.88) and combining the resulting equation with Eqs. (13.88), it follows that, for any (complex) constant P_{j+1},

$$R_{j+1}\left\{\begin{bmatrix} 1 \\ a_j(1) \\ a_j(2) \\ \vdots \\ a_j(j) \\ 0 \end{bmatrix} + \Gamma_{j+1}\begin{bmatrix} 0 \\ a_j^*(j) \\ a_j^*(j-1) \\ \vdots \\ a_j^*(1) \\ 1 \end{bmatrix}\right\} = \begin{bmatrix} \xi_j \\ 0 \\ 0 \\ \vdots \\ 0 \\ \gamma_j \end{bmatrix} + \Gamma_{j+1}\begin{bmatrix} \gamma_j^* \\ 0 \\ 0 \\ \vdots \\ 0 \\ \xi_j^* \end{bmatrix} \tag{13.91}$$

Since we want to find the vector a_{j+1}, which is to be multiplied by R_{j+1}, yields a scaled unit vector.

If we set $$P_{j+1} = -\frac{\gamma_j}{\xi_j^*} \tag{13.92}$$

then Eq.(13.91) becomes

$$\mathbf{R}_{j+1}\mathbf{a}_{j+1} = \xi_{j+1}\mathbf{U}_1$$

where $$\mathbf{a}_{j+1} = \begin{bmatrix} 1 \\ a_j(1) \\ a_j(2) \\ \vdots \\ a_j(j) \\ 0 \end{bmatrix} + \Gamma_{j+1} \begin{bmatrix} 0 \\ a_j^*(j) \\ a_j^*(j-1) \\ \vdots \\ a_j^*(1) \\ 1 \end{bmatrix} \tag{13.93}$$

which is the solution to the $(j + 1)$ st-order normal equations. Furthermore,

$$\xi_{j+1} = \xi_j + \Gamma_{j+1}\gamma_j^* = \xi_j\left[1 - \left|\Gamma_{j+1}\right|^2\right] \tag{13.94}$$

is the $(j + 1)$ st-order modelling error. If we define $a_j(0) = 1$ and $a_j(j + 1) = 0$, then Eq.(13.93), called the Levinson order-updata equation, may be written as

$$a_{j+1}(i) = a_j(j) + \Gamma_{j+1}a_j(j - i + 1); \quad i = 0, 1, \dots j + 1 \tag{13.95}$$

All that is required to complete the recursion is to define the conditions necessary to initialise the recursion. These conditions are given by the solution for the model of order $j = 0$,

$$a_0(0) = 1$$

$$\xi_0 = r_{xx}(0) \tag{13.96}$$

In summary, the steps of the Levinson-Durbin recursion are given in Table 13.1. The recursion is first initialised with the zeroth-order solution, as given in Eq.(13.95). Then, for $j = 0, 1, \dots, p-1$, the $(j + 1)$st-order model is found from the jth-order model in three steps.

Table 13.1 *The Levinson–Durbin Recursion*

Sl. No.	Steps of the Levinson–Durbin recursion
1.	Initialise the recursion (a) $a_0(0) = 1$ (b) $\xi_0 = r_{xx}(0)$ (c) $\gamma_0 = r_{xx}(1)$ (zeroth order solution)
2.	For $j = 0, 1, \dots, p-1$ (a) $\gamma_j = r_{xx}(j+1) + \sum_{i=1}^{j} a_j(i) r_{xx}(j-i+1)$ (b) $\Gamma_{j+1} = -\gamma_j / \xi_j$ (c) For $i = 1, 2 \dots, j$ $a_{j+1}(i) = a_j(i) + \Gamma_{j+1} a_j^*(j - i + 1)$ (order update equation) (d) $a_{j+1}(j + 1) = \Gamma_{j+1}$ (e) $\xi_{j+1} = \xi_j\left[1 - \left|\Gamma_{j+1}\right|^2\right]$
3.	$b(0) = \sqrt{\xi_p}$ (error update)

The first step is to use Eq.(13.89) and Eq.(13.92) to determine the value of P_{j+1}, which is called the $(j+1)^{th}$ reflection coefficient. The second step is to use the Levinson order-update equation to compute the coefficients $a_{j+1}(i)$ from $a_j(i)$. The third step is to update the error, ξ_{j+1}, using Eq.(13.94). This error may also be written in two equivalent forms. The first is

$$\xi_{j+1} = \xi_j\left[1-\left|\Gamma_{j+1}\right|^2\right] = r_{xx}(0)\prod_{i=1}^{j+1}\left[1-\left|\Gamma_i\right|^2\right] \tag{13.97}$$

and the second, which follows from calculation of modelling error is

$$\xi_{j+1} = r_x(0)+\sum_{i=1}^{j+1} a_{j+1}(i)r_{xx}(i) \tag{13.98}$$

13.8.2 AR Process and Linear Prediction

For an *AR* process, the parameters can be related to the p^{th} with order predictor.

Consider a sequence $x(n)$ stationary random process and a linear predictor of order m for $x(n)$.

$$\hat{x}(n) = -\sum_{k=1}^{m} a_m(k)x(n-k)$$

where $a_m(k)$, $k = 1, 2, ..., m$ are the m^{th} order predictor coefficients and $\hat{x}(n)$ is the forward predictor. The forward prediction error is

$$\begin{aligned} f_m(n) &= x(n)-\hat{x}(n) \\ &= x(n)+\sum_{k=1}^{m} a_m(k)x(n-k) \end{aligned}$$

The mean square value is

$$\begin{aligned} \xi_m^f &= E\left[\left|f_m(n)\right|^2\right] \\ &= E\left[\left|x(n)+\sum_{k=1}^{m} a_m(k)x(n-k)\right|^2\right] \\ &= r_{xx}(0)+2\,\mathrm{Re}\left[\sum_{i=i}^{m} a_m^*(l)r_{xx}(l)\right]+\sum_{k=1}^{m}\sum_{l=l}^{m} a_m^*(l)a_m(k)r_{xx}(l-k) \end{aligned}$$

Upon minimising the above quadratic equation, we get the set of linear equations as

$$r_{xx}(l) = -\sum_{k=1}^{m} a_m(k)r_{xx}(l-k)$$

The minimum mean square prediction error is

$$\min\left[\xi_m^f\right] = E_m^f = r_{xx}(0)+\sum_{k=1}^{m} a_m(k)r_{xx}(-k)$$

The prediction coefficients are identical to the parameters of an $AR(p)$ model. The minimum mean square prediction error is

$$\xi_p^f = \sigma_w^2$$

The set of linear equations can be solved by Levinson-Durbin algorithm yielding,

$$a_1(1) = \frac{-r_{xx}(1)}{r_{xx}(0)}$$

$$E_1^f = \left(1-\left|a_1(1)\right|^2\right)r_{xx}(0)$$

$$a_m(k) = a_{m-1}(k)+a_m(m)a_{m-1}^*(m-k)$$

where $$a_m(m) = -\frac{r_{xx}(m) + \sum_{k=1}^{m-1} a_{m-1}(k) r_{xx}(m-k)}{E_{m-1}^f}$$

$$E_m^f = \left(1 - |a_m(m)|^2\right) \; E_{m-1}^f = r_{xx}(0) \prod_{k=1}^{m} \left(1 - |a_k(k)|^2\right)$$

$a_m(m) = K_m$, m^{th} reflection coefficient in the equivalent lattice realisation of the predictor.

The forward prediction error using recursion is given by

$$f_m(n) = x(n) + \sum_{k=1}^{m-1} a_{m-1}(k) x(n-k) + a_m(m)\left[x(n-m) + \sum_{k=1}^{m-1} a_{m-1}^*(m-k) x(n-k)\right]$$

$$f_m(n) = f_{m-1}(n) + K_m g_{m-1}(n-1)$$

where $g_m(n)$ is the error in an m^{th} order backward predictor given by

$$g_m(n) = x(n-m) + \sum_{k=1}^{m} a_m^*(k) x(n-m+k)$$

The mean square value of the backward prediction error is

$$\xi_m^b = E\left\lfloor |g_m(n)|^2 \right\rfloor$$

When minimised,

$$\min\left\lfloor \xi_m^b \right\rfloor = E_m^b = E_m^f$$

For backward error, the order recursive equation is given by

$$g_m(n) = g_m(n-1) + K_m^* f_{m-1}(n)$$

The order recursive relations are given by

$$f_m(n) = f_{m-1}(n) + K_m g_{m-1}(n-1)$$

$$g_m(n) = g_{m-1}(n-1) + K_m^* f_{m-1}(n)$$

Minimisation of ξ_m^b yields,

$$\xi_m^f = E\left\lfloor |f_m(n)|^2 \right\rfloor$$

$$\xi_m^b = E\left\lfloor |g_m(n)|^2 \right\rfloor$$

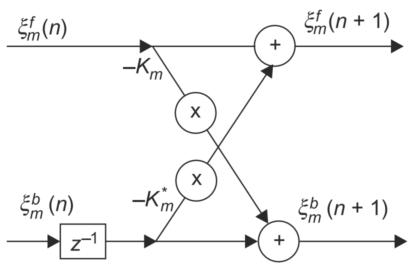


Fig. 13.5 *Lattice Filter Structure*

The value of K_m may be defined as

$$K_m = \frac{-E\left[f_{m-1}(n) g_{m-1}^*(n-1)\right]}{E\left[|g_{m-1}(n-1)|^2\right]}$$

and $$K_m^* = \frac{-E\left[f_{m-1}^*(n) g_{m-1}(n-1)\right]}{E\left[|g_{m-1}(n-1)|^2\right]}$$

Also, $$K_m = \frac{-E\left[f_{m-1}(n) g_{m-1}^*(n-1)\right]}{\sqrt{E\left[|g_{m-1}(n-1)|^2\right] E\left[|f_{m-1}(n)|^2\right]}}$$

It can be represented by the lattice filter structure as follows.

It is clear that in the lattice filter structure, the reflection coefficients $\{K_m\}$ is the negated value of the correlation coefficients between the forward and backward errors in the lattice. From the above equation, $|K_m| \le 1$. Then

$$E_m^f = \left(1-|K_m|^2\right) E_{m-1}^f$$

or
$$\sigma_{wm}^2 = \left(1-|K_m|^2\right)\left[\sigma_{wm-1}^2\right]$$

which is a monotonically decreasing sequence. For computing the order of *AR* model, this can be used. If there is no significant change in σ_{wm}^2 when the order is increased, then the algorithm can be stopped.

The reflection coefficients $\{K_m\}$ are equivalent to the coefficients of a linear *FIR* predictor. The coefficients satisfy $|K_m| \le 1$ which gives the *AR* model as stable as the roots are inside the unit circle.

$$A_p(z) = 1 + \sum_{k=1}^{p} a_p(k) z^{-k}$$

This filter is called forward prediction error filter. Similarly, the backward prediction error filter is defined by

$$B_p(z) = \sum_{k=0}^{p} b_p(k) z^{-k}$$

where $b_p(k) = a_p^*(p-k)$. Both these are noise whitening filters provided $x(n)$ is an *AR (p)* process.

13.8.3 Burg Method

For a given data sequence $x(n)$, $n = 0, 1, \ldots, N-1$, the m^{th} order forward and backward linear prediction estimates are

$$\hat{x}(n) = -\sum_{k=1}^{m} a_m(k) x(n-k)$$

$$\hat{x}(n-m) = -\sum_{k=1}^{m} a_m^*(k) x(n+k-m)$$

Let the forward and backward errors are represented by $f_m(n)$ and $g_m(n)$. Burg method minimises the total squared error given by

$$\xi_m = \sum_{n=M}^{N-1}\left[|f_m(n)|^2 + |g_m(n)|^2\right]$$

For minimising the error, with the constraint that satisfies the Levinson–Durbin recursion is

$$a_m(k) = a_{m-1}(k) + K_m a_{m-1}^*(m-k), \qquad 1 \le k \le m-1, \quad 1 \le m \le p$$

where $K_m = a_m(m)$. Using the expressions for $f_m(n)$ and $g_m(n)$ obtained in the previous section and minimising ξ_m with respect to K_m, we get

$$\hat{K}_m = \frac{-\sum_{n=M}^{N-1} f_{m-1}(n) g_{m-1}^*(n-1)}{\frac{1}{2}\sum_{n-M}^{N-1}\left[|f_{m-1}(n)|^2 + |g_{m-1}(n-1)|^2\right]}, m = 1, 2, \cdots, p$$

The denominator of $\hat{K}_m$ is the least squares estimate of the forward and backward errors. Hence,

$$\hat{K}_m = \frac{-\sum_{n=M}^{N-1} f_{m-1}(n)\, g_{m-1}^{*}(n-1)}{\frac{1}{2}\left[\hat{E}_{m-1}^{f} + \hat{E}_{m-1}^{b}\right]}, \quad m = 1, 2, \cdots, p$$

where the estimate of total squared error (ξ_m) is $\hat{E}_{m-1}^{f} + \hat{E}_{m-1}^{b} = \hat{E}_m$

$$\hat{E}_m = \left(1 - \left|\hat{K}_{m-1}\right|^2\right)\hat{E}_{m-1} - \left|\hat{f}_{m-1}(N)\right|^2 - \left|g_{m-1}(N-m)\right|^2$$

In Burg's algorithm, with the estimates of *AR* parameters, the power spectrum estimate becomes,

$$P_{xx}^{BU}(f) = \frac{\hat{E}_p}{\left|1 + \sum_{k=1}^{p} \hat{a}_p(k)\, e^{-j2\pi fk}\right|^2}$$

Advantages

Burg's method has the advantages of

(a) High frequency resolution
(b) Stable *AR* model
(c) Computationally efficient

Disadvantages

Burg's method has the following disadvantages:

(a) Exhibits spectral line splitting at high signal to noise ratios.
(b) Sensitive to the initial phase of a sinusoid.
(c) Produces spurious peaks in high order models.

To overcome the limitations, a weighting sequence is introduced in the calculation of total squared error. Therefore,

$$\xi_m^{wb} = \sum_{n=M}^{N-1} w_m(n)\left[\left|f_m(n)\right|^2 + \left|g_m(n)\right|^2\right]$$

Upon minimising the squared error, we get

$$\hat{K}_m = \frac{-\sum_{n=M}^{N-1} w_{m-1}(n)\, f_{m-1}(n)\, g_{m-1}^{*}(n-1)}{\frac{1}{2}\sum_{n=M}^{N-1} w_{m-1}(n)\left[\left|f_{m-1}(n)\right|^2 + \left|g_{m-1}(n-1)\right|^2\right]}$$

13.8.4 Unconstrained Least Squares Method

Minimising the sum of forward and backward errors, we get

$$\xi_p = \sum_{n=p}^{N-1}\left[\left|f_p(n)\right|^2 + \left|g_p(n)\right|^2\right]$$

$$= \sum_{n=p}^{N-1}\left[\left|x(n) + \sum_{k=1}^{p} a_p(k)\, x(n-k)\right|^2 + \left|a(n-p) + \sum_{k=1}^{p} a_p^{*}(k)\, x(n+k-p)\right|^2\right]$$

Without using the Levinson-Durbin constraint for *AR* parameters, the minimisation of ξ_p with respect to the prediction coefficients results in

$$\sum_{k=1}^{p} a_p(k) r_{xx}(l,k) = -r_{xx}(l,0), \quad l = 1,2,\cdots,p$$

where the autocorrelation function is

$$r_{xx}(l,k) = \sum_{n=0}^{N-1}\left[x(n-k)x^*(n-l) + x(n-p+l)x^*(n-p+k)\right]$$

The error is $\xi_p^{LS} = r_{xx}(0,0) + \sum_{k=1}^{p} \hat{a}_p(k)\, r_{xx}(0,k)$

The power spectral estimate is

$$P_{xx}^{LS}(f) = \frac{E_p^{LS}}{\left|1 + \sum_{k=1}^{p} \hat{a}_p(k) e^{-j2\pi fk}\right|^2}$$

Unconstrained least square method can also be called as unwindowed least squares method. The limitations in Burg's method are overcome by this method. Also computational efficiency is comparable with Levinson-Durbin algorithm. But stability is not guaranteed. In the above methods, *AR* parameter estimates are obtained from a block of data $x(n)$, $n = 0, 1, ..., N-1$. If continuous data is not available, then segments of data with N-point block are taken for estimation.

13.8.5 The Capon Method (Maximum Likelihood Method)

The Capon method was initially used for large seismic array of frequency wave number estimation. Later on, this was extended to single time series spectrum estimation by Lucas. Consider an *FIR* filter, where the filter coefficients are a_k, where $0 \le k \le p$. Let $x(n)$, $0 \le n \le N-1$ be the data sequence which is passed through the FIR filter. The filter response is

$$y(n) = \sum_{k=0}^{p} a_k x(n-k) = \mathbf{X}^T\mathbf{a}$$

where $\mathbf{X}^T = [x(n)x(n-1)...x(n-p)]$, the data vector, and **a** is the filter coefficient vector.

Let us assume that the mean of the data sequence be zero, i.e. $E[x(n)] = 0$. The variance of $y(n)$ becomes

$$\sigma_y^2 = E\left[\left|y(n)\right|^2\right]$$
$$= E[\mathbf{a}^{*T}\mathbf{X}^*\mathbf{X}\mathbf{a}] = \mathbf{a}^*\mathbf{R}_x\mathbf{a}$$

where $\mathbf{R}_x$ is autocorrelation matrix of $x(n)$ with elements $r_x(m)$.

The filter coefficients are chosen such that the frequency response is normalised to unity at a particular selected frequency, f_r. Therefore $\sum_{k=0}^{p} a_k e^{-j2\pi kft} = 1.$

In matrix form, $E^{*t}(f_1)\, \boldsymbol{a} = \mathbf{1}$, where $E^t(f_1) = [1\ e^{j2\pi ft} \ldots e^{j2\pi ft}]$. Minimising the variance σ_y^2 subjected to the constraint specified yields an FIR filter. This yields the filter coefficients

$$\hat{\boldsymbol{a}} = \frac{R_x^{-1}E^*(f_1)}{E^t(f_1)R_x^{-1}E^*(f_1)}$$

The variance becomes

$$\sigma_{\min}^2 = \frac{1}{E^t(f_l)\, R_x^{-1} E^*(f_l)}$$

The minimum variance power spectrum estimate at frequency f_l is represented in the above equation. By varying the frequency f_l from 0 to 0.5, the power spectrum estimate can be obtained. Even if f_l changes, R_{xx}^{-1} is computed only once. The denominator of $\sigma_{\min}^2$ can be computed using single DFT. The minimum variance power spectrum estimate of Capon's method is

$$P_x^{mu}(f) = \frac{1}{E^t(f) R_x^{-1} E^*(f)}$$

This estimate results in spectral peaks proportional to the power at that frequency.

13.8.6 The Pisarenko Harmonic Decomposition Method

The Pisarenko Harmonic Decomposition method provides an estimate for signal components which are sinusoids corrupted by additive white noise. A real sinusoid signal can be obtained from the difference equation

$$x(n) = -a_1 x(n-1) - a_2 x(n-2)$$

where, $a_1 = 2 \cos 2\pi f_k$ and $a_2 = 1$. Here the initial conditions are $x(-1) = -1$ and $x(-2) = 0$.

This system has complex-conjugate poles at $f = f_k$ and $f = -f_k$, which results in the sinusoid $x(n) = \cos 2\pi f_k$, $n \geq 0$. Considering p sinusoid components available in the signal,

$$x(n) = -\sum_{m=1}^{2p} a_m x(n-m)$$

The system function is given by

$$H(z) = \frac{1}{1 + \sum_{m=1}^{2p} a_m z^{-m}}$$

The denominator has *2p* roots on the unit circle. They correspond to the sinusoid frequencies. Assume that the sinusoids are corrupted by an additive white noise $w(n)$.

$$y(n) = x(n) + w(n) \quad \text{and} \quad E\left[\left|w(n)\right|^2\right] = \sigma_w^2$$

Substituting the value of $x(n)$ in the difference equation, we get

$$y(n) - w(n) = -\sum_{m=0}^{2p} a_m \left[y(n-m) - w(n-m)\right]$$

$$\sum_{m=0}^{2p} a_m y(n-m) = \sum_{m=0}^{2p} a_m w(n-m)$$

This leads to a difference equation for ARMA *(p,q)* process which has both AR and MA parameters identical. In matrix form,

$$\mathbf{Y}^T\mathbf{a} = \mathbf{W}^T\mathbf{a}$$

where $\mathbf{Y}^T = [y(n)y(n-1)...y(n-2p)]$ is the observed data vector

$\mathbf{W}^T = [w(n)w(n-1)...w(n-2p)]$ is the noise vector and

$\mathbf{a} = [1, a_1, ..., a_{2p}]$ is the coefficient vector

Premultiplying by **Y** and taking expected value,

$$E(\mathbf{YY}^{\mathrm{T}})\mathbf{a} = E(\mathbf{YW}^{T})\mathbf{a} = E(\mathbf{X}+\mathbf{W})\mathbf{W}^{T}]\mathbf{a}$$

$$\mathbf{R}_y\mathbf{a} = \sigma^2_w\mathbf{a}$$

The above equation is obtained with the following assumptions,

(i) $w(n)$ is zero mean and white, and

(ii) X is a deterministic signal

In Eigen equation form, $(\mathbf{R}_y - \sigma^2_w\mathbf{I})\,\mathbf{a} = \mathbf{0}$ where σ^2_w is the eigen value of the autocorrelation matrix $\mathbf{R}_y$ and **I** is an identity matrix. This forms the basis for the decomposition method. The autocorrelation values are given by

$$r_y(0) = \sigma_w^2 + \sum_{i=1}^{p} p_i$$

$$r_y(k) = \sum_{i=1}^{p} p_i \cos 2\pi f_i k, \quad k \neq 0$$

where $p_i = \dfrac{A_i^2}{2}$ is the average power in i^{th} sinusoid. Hence,

$$\begin{bmatrix} \cos 2\pi f_1 & \cos 2\pi f_2 & \cdots & \cos 2\pi f_p \\ \cos 4\pi f_1 & \cos 4\pi f_2 & \cdots & \cos 4\pi f_4 \\ \vdots & \vdots & & \vdots \\ \cos 2\pi f_1 & \cos 2\pi f_2 & \cdots & \cos 2\pi f_p \end{bmatrix} \begin{bmatrix} p_1 \\ P_2 \\ \vdots \\ P_p \end{bmatrix} = \begin{bmatrix} r_y(1) \\ r_y(2) \\ \vdots \\ r_y(p) \end{bmatrix}$$

The powers of sinusoids can be obtained if the frequencies are known. $r_{xx}(m)$ can be replaced by its estimate $\mathbf{R}_x(m)$.

The noise variance is obtained from,

$$\sigma_w^2 = r_y(0) - \sum_{t=1}^{p} p_i$$

Example 13.9 *Consider an AR process of order 1, described by $x(n) = ax(n-1) + w(n)$ where 'a' is a constant and $\{w(n)\}$ is a white noise process of zero mean and variance σ^2. Find the mean and autocorrelation function of the process $x(n)$.*

Solution $x(n) - ax(n-1) = w(n)$

Upon solving the linear difference equation with constant coefficients, the complementary solution and particular solution can be obtained.

Complementary solution

Consider the homogeneous equation

$$(n) - ax\,(n-1) = 0$$

The solution for the above equation is of the form

$$x(n) = ca^n$$

where c is a constant.

Particular solution

$$x(n-1) = z^{-1}\,x(n)$$

$x(n) - ax(n-1) = w(n)$ can be written as

$$x(n) - az^{-1}x(n) = w(n)$$

$$(1-az^{-1})x(n) = w(n)$$

$$x(n)=\frac{1}{1-az^{-1}}w(n)$$

$$=\left(\sum_{k=0}^{\infty}a^k z^{-k}\right)w(n), \qquad \left(\text{since } \frac{1}{1-az^{-1}}=1+az^{-1}+a^2z^{-2}+\cdots=\sum_{k=0}^{\infty}a^k z^{-k}\right)$$

$$=\sum_{k=0}^{\infty}a^k z^{-k}w(n)=\sum_{k=0}^{\infty}a^k w(n-k)$$

Combining both the solutions,

$$x(n)=ca^n+\sum_{k=0}^{\infty}a^k w(n-k)$$

Assuming $x(0) = 0$,

$$c=-\sum_{k=0}^{\infty}a^k w(-k)$$

Hence, $$x(n)=-a^n\sum_{k=0}^{\infty}a^k w(-k)+\sum_{k=0}^{\infty}a^k w(n-k)$$

$$=-\sum_{k=0}^{\infty}a^k w(n-k)+\sum_{k=0}^{\infty}a^k w(n-k)=-\sum_{k=0}^{\infty}a^k w(n-k)$$

Mean

$$E\left[w(n)\right]=0$$

$$E\left[x(n)\right]=E\left[\sum_{k=0}^{n-1}a^k w(n-k)\right]=\sum_{k=0}^{n-1}a^k E\left[w(n-k)\right]=0 \text{ for all } n$$

Autocorrelation

$$r_{xx}(1)=E\left[x(n)x(n-1)\right]$$

$$=E\left[\sum_{k=0}^{n-1}a^k w(n-k)\sum_{k=0}^{n-1}a^i w(n-l-i)\right]$$

$$=E\left[\sum_{k=0}^{n-1}\sum_{i=0}^{n-1}a^{k+i}w(n-k)w(n-l-i)\right]$$

$$=\sum_{k=0}^{n-1}\sum_{i=0}^{n-1}a^{k+i}E\left[w(n-k)w(n-l-i)\right]$$

$$E\left[w(n-k)w(n-l-i)\right]=\begin{cases}\sigma^2, & k=l+i\\ 0, & k\neq l+i\end{cases}$$

Since $[w(n)]$ is a white noise process,

$$E\left[x(n)x(n-l)\right]=\sigma^2 a^{-l}\sum_{k=0}^{n-1}a^{2k}=\sigma^2 a^{-l}\left\{\frac{1-a^{2n}}{1-a^2}\right\}$$

Example 13.10 *The second-order AR process is given by*

$$x(n) + a_1x(n-1) + a_2x(n-2) = w(n)$$

where $\{w(n)\}$ is a white noise process of zero mean and variance σ^2. Determine the conditions required for this AR process to be asymptotically stationary up to order two.

Solution $x(n) + a_1x(n-1) + a_2x(n-2) = w(n)$

$$Z[x(n-1)] = z^{-1}X(z)$$

$$Z[x(n-2)] = z^{-2}X(z)$$

Therefore, $X(z) + a_1z^{-1}X(z) + a_2z^{-2}X(z) = W(z)$

$$(1 + a_1z^{-1} + a_2z^{-2})\,X(z) = W(z)$$

Let $A(z) = \dfrac{W(z)}{X(z)} = 1 + a_1z^{-1} + a_2z^{-2} = \left(1-\gamma_1z^{-1}\right)\left(1-\gamma_2z^{-1}\right)$

where γ_1 and γ_2 are roots of $A(z)$.

Complementary Solution

The solution of the homogeneous equation

$$x(n) + a_1x(n-1) + a_2x(n-2) = 0$$

is of the form

$$c_1\gamma_1^n + c_2\gamma_2^n$$

where c_1 and c_2 are constants.

Particular Solution

$$X(z) = \frac{1}{\left(1-\gamma_1z^{-1}\right)\left(1-\gamma_2z^{-1}\right)}W(z)$$

$$= \frac{1}{\gamma_1-\gamma_2}\left(\frac{\gamma_1}{1-\gamma_1z^{-1}} - \frac{\gamma_2}{1-\gamma_2z^{-1}}\right)W(z)$$

$$= \frac{1}{\gamma_1-\gamma_2}\left(\sum_{k=0}^{\infty}\gamma_1^{k+1}z^{-k} - \sum_{k=0}^{\infty}\gamma_2^{k+1}z^{-k}\right)W(z)$$

$$= \frac{1}{\gamma_1-\gamma_2}\left(\sum_{k=0}^{\infty}\gamma_1^{k+1}z^{-k}W(z) - \sum_{k=0}^{\infty}\gamma_2^{k+1}z^{-k}W(z)\right)$$

Taking inverse z-transform, we get

$$x(n) = \frac{1}{\gamma_1-\gamma_2}\left(\sum_{k=0}^{\infty}\gamma_1^{k+1}w(n-k) - \sum_{k=0}^{\infty}\gamma_2^{k+1}z^{-k}w(n-k)\right)$$

$$= \sum_{k=0}^{\infty}\left(\frac{\gamma_1^{k+1}-\gamma_2^{k+1}}{\gamma_1-\gamma_2}\right)w(n-k)$$

Combining the above particular solution and complementary solution, we get the general solution as $x(n) = c_1\gamma_1^n + c_2\gamma_2^n + \displaystyle\sum_{k=0}^{\infty}\left(\frac{\gamma_1^{k+1}-\gamma_2^{k+1}}{\gamma_1-\gamma_2}\right)w(n-k)$

For the asymptotically stationary process, $(c_1\gamma_1^n + c_2\gamma_2^n)$ must decay to zero as $n\to\infty$.

Hence, $|\gamma_1| < 1$ and $|\gamma_2| < 1$

OPTIMUM FILTERS 13.9

Signal estimation is one of the most important problems in signal processing. In signal estimation, the desired signal is recovered in the best possible way from its degraded replica. In many applications, the desired signal (e.g., speech, image, radar signal, EEG, etc.) is not observed directly. Thus the desired signal may be noisy and distorted for a variety of reasons. Generally, filters are used to recover the desired signal from the measured data. To produce the best estimate of the signal, these filters should be optimum. Therefore, the optimum filters namely the Wiener and Kalman filters are introduced in this section.

13.9.1 Wiener Filter

A Wiener filter aims at designing a filter that would produce optimum estimate of a signal from a noisy measurement. The Wiener filter model shown in Fig. 13.6 is used to recover a signal d(n) from noisy observation $x(n)$.

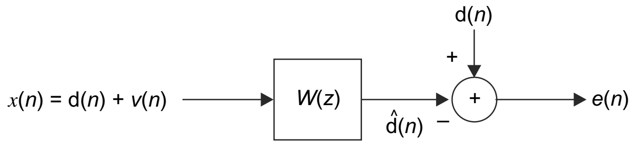


Fig. 13.6 *Wiener Filter Model*

Assuming d(n) and $v(n)$ are wide-sense stationary random processes, a Wiener filter aims to produce the minimum mean-square error ξ. Thus the estimation criterion to be minimized is

$$\xi = E\left[\left|e(n)^2\right|\right], \text{ where } e(n) = \mathrm{d}(n) - \hat{\mathrm{d}}(n) \text{ is the error}$$

Here, d(n) is the desired response and $\hat{d}(n)$ is the output response. Now, the problem is to find the filter that minimises ξ. Depending on how the signals $x(n)$ and d(n) are related to each other, a number of problems can be formulated.

(i) *Filtering* to estimate d(n) using a causal filter given $x(n) = \mathrm{d}(n) + v(n)$
(ii) *Smoothing* to estimate d(n) using all available data, i.e. non-causal
(iii) *Prediction* if $\mathrm{d}(n) = x(n+1)$, where $W(z)$ is a causal filter.
(iv) *Deconvolution* if $x(n) = \mathrm{d}(n)*g(n) + v(n)$, where $g(n)$ is the unit sample response of a linear shift invariant filter, then the Wiener filter becomes a deconvolution filter.

Let us now consider the design of FIR Wiener filter, IIR Wiener filter and linear predictor.

FIR Wiener Filter

An FIR Wiener filter is designed to produce the minimum mean square estimate of a given process $d(n)$ by filtering a set of observations of a statistically related process $x(n)$. Let us assume that $x(n)$ and $d(n)$ are jointly wide sense stationary random process with known autocorrelations $r_{xx}(k)$ and $r_{\mathrm{d}}(k)$ respectively, and known cross-correlation $r_{\mathrm{d}x}(k)$. The unit sample response of the Wiener filter is $w(n)$ and assuming $a(p-1)^{\mathrm{st}}$ order filter. The output of the Wiener filter is

$$\hat{\mathrm{d}}(n) = \sum_{t=0}^{p-1} w(l)\, x(n-l) \tag{13.99}$$

The Wiener filter design problem requires that the filter coefficients $w(k)$ are found such that it minimises the mean-square error.

$$\xi = E\left\{\left|e(n)\right|^2\right\} = E\left\{\left|\mathrm{d}(n) - \hat{\mathrm{d}}(n)\right|^2\right\} \tag{13.100}$$

In order to minimise the error, differentiate the above equation with respect to $w^*(k)$ and equate to zero for $k = 0, 1, ..., (p-1)$.

$$\frac{\partial \varepsilon}{\partial w^*(k)} = \frac{\partial}{\partial w^*(k)} E\left\{\left|e(n)e^*(n)\right|\right\} = E\left\{e(n)\frac{\partial e^*(n)}{\partial w^*(k)}\right\} = 0 \tag{13.101}$$

We know that, $e(n) = \mathrm{d}(n) - \hat{\mathrm{d}}(n)$

$$e(n) = \mathrm{d}(n) - \sum_{l=0}^{p-1} w(l)x(n-l)$$

$$e^*(n) = \mathrm{d}^*(n) - \sum_{l=0}^{p-1} w^*(l)x^*(n-l) \tag{13.102}$$

Therefore, $\dfrac{\partial e^*(n)}{\partial w^*(k)} = -x^*(n-k)$

From Eq.(13.101), we have

$$E\left[e(n)x^*(n-k)\right] = 0; k = 0,1,\cdots,(p-1) \tag{13.103}$$

which is called the orthogonality principle or projection theorem. In this,

(i) the error $\mathrm{d}(n) - \hat{\mathrm{d}}(n)$ is orthogonal to $\hat{\mathrm{d}}(n)$ and

(ii) the best estimator $\mathrm{d}(n)$ is the projection of $d(n)$ in $\hat{\mathrm{d}}(n)$.

Substituting the value of Eq.(13.102) in Eq.(13.103), we get

$$E\left\{\left[\mathrm{d}(n) - \sum_{l=0}^{p-1} w(l)x(n-l)\right]x^*(n-k)\right\} = 0$$

$$E\left\{\mathrm{d}(n)x^*(n-k)\right\} - \sum_{l=0}^{p-1} w(l)E\left(x(n-l)x^*(n-k)\right) = 0$$

$$\sum_{l=0}^{p-1} w(l)E\left\{x(n-l)x^*(n-k)\right\} = E\left\{\mathrm{d}(n)x^*(n-k)\right\}$$

$$\sum_{l=0}^{p-1} w(l)r_{xx}(k-l) = r_{\mathrm{dx}}(k); \qquad k = 0,1,\cdots,\ (p-1)$$

In matrix form,

$$\begin{bmatrix} r_{xx}(0) & r_{xx}^*(1) & \cdots & r_{xx}^*(p-1) \\ r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}^*(p-2) \\ r_{xx}(2) & r_{xx}(1) & \cdots & r_{xx}^*(p-3) \\ \vdots & \vdots & \cdots & \\ r_{xx}(p-1) & r_{xx}(p-2) & \cdots & r_{xx}(0) \end{bmatrix} \begin{bmatrix} w(0) \\ w(1) \\ w(2) \\ \vdots \\ w(p-1) \end{bmatrix} = \begin{bmatrix} r_{\mathrm{dx}}(0) \\ r_{\mathrm{dx}}(1) \\ r_{\mathrm{dx}}(2) \\ \vdots \\ r_{\mathrm{dx}}(p-1) \end{bmatrix} \tag{13.104}$$

Equation (13.104) is called Wiener-Hopf equation. In simplified form, $\mathbf{R}_x w = \mathbf{r}_{dx}$, where $\mathbf{R}_x$ is a $p \times p$ Hermitian Toeplitz matrix of autocorrelation, $\boldsymbol{W}$ is the vector of the filter coefficients and $\mathbf{r}_{\mathrm{d}x}$ is the vector of the cross-correlation between the desired signal $d(n)$ and the observed signal $x(n)$.

From Eq.(13.104), $\mathbf{w} = \mathbf{R}_x^{-1}\mathbf{r}_{\mathrm{d}x}$

The mean square error is derived as follows:

$$\xi = E\left\{\left|e(n)^2\right|\right\} = E\left(e(n)e^*(n)\right)$$

$$= E\left\{e(n)\left(\mathrm{d}(n) - \sum_{l=0}^{p-1} w(l)x(n-l)\right)^*\right\} = E\left\{e(n)\mathrm{d}^*(n) - \sum_{l=0}^{p-1} w^*(l)e(n)x^*(n-l)\right\}$$

$$= E\left\{e(n)\mathrm{d}^*(n)\right\} - \sum_{l=0}^{p-1} w^*(l)E\left(e(n)x^*(n-l)\right)$$

From Eq.(13.103), we have

$$E\left(e(n)x^*(n-k)\right) = 0$$

Therefore, $\xi_{\min} = E\left\{e(n)\mathrm{d}^*(n)\right\}$

Substituting for $e(n)$, we get

$$\xi_{\min} = E\left\{\left(\mathrm{d}(n) - \sum_{l=0}^{p-1} w(l)x(n-l)\right)\mathrm{d}^*(n)\right\}$$

$$= E\left\{\mathrm{d}(n)\mathrm{d}^*(n) - \sum_{l=0}^{p-1} w(l)x(n-l)\mathrm{d}^*(n)\right\}$$

$$= E\left\{\mathrm{d}(n)\mathrm{d}^*(n)\right\} - \sum_{l=0}^{p-1} w(l)E\left(x(n-l)\mathrm{d}^*(n)\right)$$

$$\xi_{\min} = r_{\mathrm{d}}(0) - \sum_{l=0}^{p-1} w(l)r_{\mathrm{d}x}^*(l)$$

$$\xi_{\min} = r_{\mathrm{d}}(0) - \mathbf{r}_{\mathrm{dx}}^{\mathrm{H}} w \tag{13.105}$$

Therefore, the minimum mean square error is expressed as

$$\xi_{\min} = r_{\mathrm{d}}(0) - \mathbf{r}_{\mathrm{dx}}^{\mathrm{H}}\mathbf{R}_{\mathrm{x}}^{-1}\mathbf{r}_{\mathrm{dx}} \tag{13.106}$$

Thus, the minimum error is expressed in terms of the autocorrelation matrix R_x and the cross correlation matrix $\mathbf{r}_{\mathrm{dx}}$.

Filtering

In the filtering problem, a signal d(n) is to be estimated from a noise corrupted observation

$$x(n) = \mathrm{d}(n) + v(n)$$

where $v(n)$ is the uncorrelated noise with d(n), i.e., $E\{\mathrm{d}(n)v^*(n-k)\} = 0$

Therefore, the cross-correlation between $x(n)$ and d(n) is

$$r_{\mathrm{d}x}(k) = E\left\{\mathrm{d}(n)x^*(n-k)\right\}$$

$$= E\left\{\mathrm{d}(n)\left(\mathrm{d}^*(n-k) + v^*(n-k)\right)\right\}$$

$$= E\left\{\mathrm{d}(n)\mathrm{d}^*(n-k)\right\} + E\left\{\mathrm{d}(n)v^*(n-k)\right\} = r_{\mathrm{dd}}(k)$$

We know that $r_{xx}(k) = E\{x(n+k)x^*(n)\}$

$$= E\{(\mathrm{d}(n+k)+v(n+k))(\mathrm{d}^*(n)+v^*(n))\}$$

Since the signal and noise are uncorrelated, we have

$$r_{xx}(k) = E\{\mathrm{d}(n+k)\mathrm{d}^*(n)\} + E(v(n+k)v^*(n)) = r_{\mathrm{dd}}(k) + r_{vv}(k)$$

Using above equations, the Wiener-Hopf equation becomes

$$\mathbf{R}_{\mathrm{d}} + \mathbf{R}_{\mathrm{v}}\mathbf{w} = \mathbf{r}_{\mathrm{d}x}$$

where $\mathbf{R}_{\mathrm{d}}$ and $\mathbf{R}_{\mathrm{v}}$ are the autocorrelation matrices of d(n) and v(n) respectively.

Linear Prediction

Linear prediction is also an important problem in digital signal processing. With noise-free observations, linear prediction is concerned with the estimation of $x(n + 1)$ in terms of a linear combination of the current and previous values of $x(n)$.

Thus, an *FIR* linear predictor of order $(p - 1)$ has the form

$$\hat{\mathrm{d}}(n) = \hat{x}(n+1) = \sum_{k=0}^{p-1} w(k)x(n-k),$$

where $w(k)$ are the coefficients of Wiener prediction filter. The cross-correlation is

$$r_{\mathrm{d}x}(k) = E\{\mathrm{d}(n)x^*(n-k)\}$$

$$= E\{x(n+1)x^*(n-k)\} = r_{xx}(k+1)$$

Then the Wiener-Hopf equations for the optimum linear predictor are

$$\begin{bmatrix} r_{xx}(0) & r_{xx}^*(1) & \cdots & r_{xx}^*(p-1) \\ r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}^*(p-2) \\ \vdots & \vdots & & \vdots \\ r_{xx}(p-1) & r_{xx}(p-2) & \cdots & r_{xx}(0) \end{bmatrix} \begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(p-1) \end{bmatrix} = \begin{bmatrix} r_{xx}(1) \\ r_{xx}(2) \\ \vdots \\ r_{xx}(p) \end{bmatrix}$$

The mean-squared error is

$$\xi_{\min} = r_x(0) - \sum_{k=0}^{p-1} w(k) r_{xx}^*(k+1)$$

In noise environment, the input to the Wiener filter is $y(n) = x(n) + v(n)$. The aim is to estimate $x(n + 1)$ in terms of a linear combination of p previous values of $y(n)$. The model for Wiener prediction in a noisy environment is shown in Fig. 13.7.

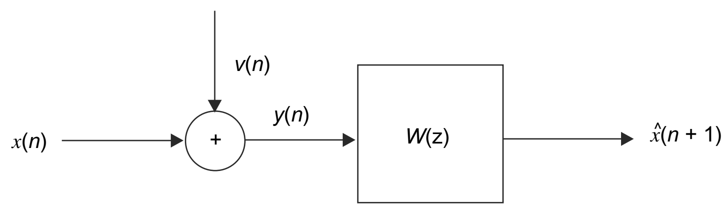


Fig. 13.7 *Wiener Prediction Filter in a Noisy Environment*

$$\hat{x}(n+1)=\sum_{k=0}^{p-1} w(k)\,y(n-k)=\sum_{k=0}^{p-1} w(k)\big[x(n-k)+v(n-k)\big]$$

Using, the Wiener-Hopf equations, $\mathbf{R}_y w = \mathbf{r}_{\mathrm{dy}}$ and if the noise is uncorrelated with the signal, then

$$r_{yy}(k)=E\big[y(n)\,y^*(n-k)\big]=r_{xx}(k)+r_{vv}(k)$$

$$\mathbf{R}_{\mathrm{y}}=\mathbf{R}_{\mathrm{x}}+\mathbf{R}_{\mathrm{v}}$$

The cross-correlation between d(n) and $y(n)$ is given by

$$r_{\mathrm{dy}}(k)=E\big(\mathrm{d}(n)\,y^*(n-k)\big)=E\big(x(n+1)+y^*(n-k)\big)=r_{xx}(k+1)$$

Therefore, the difference in linear prediction with and without noise is in the autocorrelation matrix for the input signal. The above problem is a one-step predictor and this can also be extended for multi-step prediction. In multi-step prediction, $x(n+\alpha)$ is predicted in terms of linear combination of the p previous values of $x(n)$.

$$\hat{x}(n+\alpha)=\sum_{k=0}^{p-1} w(k)\,x(n-k)$$

where α is a positive integer.

The Wiener-Hopf equations for a multi-step predictor is given by

$$\begin{bmatrix} r_{xx}(0) & r_{xx}^*(1) & \cdots & r_{xx}^*(p-1) \\ r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}^*(p-2) \\ \vdots & \vdots & & \vdots \\ r_{xx}(p-1) & r_{xx}(p-2) & \cdots & r_{xx}(0) \end{bmatrix}\begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(p-1) \end{bmatrix}=\begin{bmatrix} r_{xx}(\alpha) \\ r_{xx}(\alpha+1) \\ \vdots \\ r_{xx}(\alpha+p-1) \end{bmatrix}$$

In matrix form,

$$\mathbf{R}_{\mathrm{x}} w=\mathbf{r}_{\alpha}$$

Similarly, $$\xi_{\min}=r_{xx}(0)-\sum_{k=0}^{p-1} w(k)\,r_{xx}^*(\alpha+k)=\mathbf{r}_{xx}(0)-\mathbf{r}_{xx}^{\mathrm{H}} w$$

IIR Wiener Filter

An IIR Wiener filter also reduces the mean-squared error using infinite coefficients. There are two approaches.

(i) No constraint is placed on the solution; hence the optimum filter will be non-causal and unrealisable,

(ii) By putting a constraint $h(n) = 0$ for $n < 0$ the non-causal Wiener filter, a simple closed-form expression for the frequency response can be found. For the causal Wiener filter, the system function is specified in terms of spectral factorisation.

Non-causal IIR Wiener Filter

The system function for the non-causal filter is

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

which minimises the mean-square error

$$\xi=E\Big(\big|e(n)\big|^2\Big) \tag{13.107}$$

where $e(n)$ is the difference between the desired process $\mathrm{d}(n)$ and the output of the Wiener filter $\hat{\mathrm{d}}(n)$.

$$e(n) = \mathrm{d}(n) - \hat{\mathrm{d}}(n) = \mathrm{d}(n) - \sum_{l=-\infty}^{p} h(l)x(n-l)$$

The Eq.(13.107) is expressed as

$$\xi = E\left[e(n)e^*(n)\right] \tag{13.108}$$

Differentiating Eq.(13.108) partially w.r.t. $h^*(k)$ and equating to zero, we have

$$\frac{\partial \varepsilon}{\partial h^*(k)} = E\left(e(n)\frac{\partial e^*(n)}{\partial h^*(k)}\right) = 0$$

But, $$\frac{\partial e^*(n)}{\partial h^*(k)} = -x^*(n-k)$$

Therefore, $$\frac{\partial \varepsilon}{\partial h^*(k)} = -E\left(e(n)x^*(n-k)\right) = 0; \quad -\infty < k < \infty$$

i.e. $$E\left(e(n)x^*(n-k)\right) = 0; \qquad -\infty \le k \le \infty$$

This is called orthogonality principle and it is same as that of the orthogonality principle of FIR Wiener filters.

Now, $$E\left(\left(\mathrm{d}(n) - \sum_{l=-\infty}^{\infty} h(l)x(n-l)\right)x^*(n-k)\right) = 0$$

$$E\left(\mathrm{d}(n)x^*(n-k)\right) - \sum_{l=-\infty}^{\infty} h(l)E\left(x(n-l)x^*(n-k)\right) = 0$$

$$r_{\mathrm{d}x}(k) - \sum_{l=-\infty}^{\infty} h(l)r_{xx}(k-l) = 0$$

$$\sum_{l=-\infty}^{\infty} h(l)r_{xx}(k-l) = r_{\mathrm{d}x}(k); \quad -\infty < k < \infty \tag{13.109}$$

which are the Wiener-Hopf equations of non-causal IIR Wiener filter. Comparing the Wiener-Hopf equations for *FIR* Wiener filter and the non-causal IIR filter, we see that only difference is in the limits on the summation. Equation (13.109) corresponds to an infinite set of linear equations with infinite number of unknowns and it can be written as

$$h(k) * r_{xx}(k) = r_{\mathrm{d}x}(k)$$

In the frequency domain,

$$H\left(e^{j\omega}\right)P_x\left(e^{j\omega}\right) = P_{\mathrm{d}x}\left(e^{j\omega}\right)$$

The frequency response of the Wiener filter is

$$H\left(e^{j\omega}\right) = \frac{P_{\mathrm{d}x}\left(e^{j\omega}\right)}{P_x\left(e^{j\omega}\right)}$$

i.e. $$H(z) = \frac{P_{\mathrm{d}x}(z)}{P_x(z)}$$

The minimum value of the mean squared error is derived as follows:

$$\xi = E\left[\left|e(n)^2\right|\right] = E\left(e(n)e^*(n)\right)$$

$$= E\left(e(n)\left(\mathrm{d}^*(n) - \sum_{l=-\infty}^{\infty} h^*(l)x^*(n-l)\right)\right)$$

$$\xi_{\min} = r_{\mathrm{dd}}(0) - \sum_{l=-\infty}^{\infty} h^*(l) r_{\mathrm{d}x}(l)$$

Using Parseval's theorem, this can be expressed in the frequency domain as

$$\xi_{\min} = r_{\mathrm{dd}}(0) - \frac{1}{2\pi}\int_{-\pi}^{\pi} H\left(e^{j\omega}\right)P_{\mathrm{d}x}^*\left(e^{j\omega}\right)\mathrm{d}\omega$$

Also, $$r_{\mathrm{dd}}(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_{\mathrm{d}}\left(e^{j\omega}\right)\mathrm{d}\omega$$

Therefore, $$\xi_{\min} = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left(P_{\mathrm{d}}\left(e^{j\omega}\right) - P_{\mathrm{d}x}^*\left(e^{j\omega}\right)H\left(e^{j\omega}\right)\right)\mathrm{d}\omega \tag{13.110}$$

Noise Filtering

The input to the IIR Wiener filter in noisy environment is

$$x(n) = \mathrm{d}(n) + v(n)$$

Assuming d(n) and $v(n)$ are uncorrelated zero mean random process, the autocorrelation of $x(n)$ is

$$r_{xx}(k) = r_{\mathrm{dd}}(k) + r_{vv}(k)$$

Then the power spectral density is

$$P_x\left(e^{j\omega}\right) = P_{\mathrm{d}}\left(e^{j\omega}\right) + P_v\left(e^{j\omega}\right)$$

The cross correlation of the input and output is

$$r_{\mathrm{d}x}(k) = E\left(\mathrm{d}(n)x^*(n-k)\right)$$

$$= E\left(\mathrm{d}(n)\left(\mathrm{d}^*(n-k) + v^*(n-k)\right)\right) = E\left(\mathrm{d}(n)\mathrm{d}^*(n-k)\right) = r_{\mathrm{d}}(k)$$

Therefore, $$P_{\mathrm{d}x}\left(e^{j\omega}\right) = P_{\mathrm{d}}\left(e^{j\omega}\right) \tag{13.111}$$

Then, the filter transfer function is

$$H\left(e^{j\omega}\right) = \frac{P_{\mathrm{d}}\left(e^{j\omega}\right)}{P_{\mathrm{d}}\left(e^{j\omega}\right) + P_v\left(e^{j\omega}\right)}$$

The minimum mean squared error can be obtained from Eq.(13.110) and Eq.(13.111) as

$$\xi_{\min} = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left[P_{\mathrm{d}}\left(e^{j\omega}\right) - H\left(e^{j\omega}\right)P_{\mathrm{d}\left(e^{j\omega}\right)}\right]\mathrm{d}\omega$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} P_{\mathrm{d}}\left(e^{j\omega}\right)\left(1 - H\left(e^{j\omega}\right)\right)\mathrm{d}\omega$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} P_{\mathrm{d}}\left(e^{j\omega}\right)\left(1 - \frac{P_{\mathrm{d}}\left(e^{j\omega}\right)}{P_{\mathrm{d}}\left(e^{j\omega}\right) + P_v\left(e^{j\omega}\right)}\right)\mathrm{d}\omega$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} P_v\left(e^{j\omega}\right)\frac{P_{\mathrm{d}}\left(e^{j\omega}\right)}{P_{\mathrm{d}}\left(e^{j\omega}\right) + P_v\left(e^{j\omega}\right)}\mathrm{d}\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_v\left(e^{j\omega}\right)H\left(e^{j\omega}\right)\mathrm{d}\omega$$

In the z-domain,

$$\xi_{\min} = \frac{1}{2\pi}\oint_C P_v(z)H(z)z^{-1}\mathrm{d}z$$

If $v(n)$ is a white noise process, then $P_v\left(e^{j\omega}\right) = \sigma_v^2$

$$\xi_{\min} = \sigma_v^2\,\frac{1}{2\pi}\int_{-\pi}^{\pi} H\left(e^{j\omega}\right)\mathrm{d}\omega = \sigma_v^2 h(0)$$

Causal IIR Wiener Filter

The causal Wiener filter is the one whose impulse response is constrained to zero for negative time. The optimal causal impulse response has zero response for negative time and has zero derivatives of ε with respect to impulse response for all times equal to and greater than zero. The causal Wiener filter equation becomes,

$$\sum_{l=0}^{\infty} h(l)\, r_{xx}(k-l) = r_{\mathrm{d}x}(k); \qquad k > l$$

This is not a simple convolution like the unconstrained Wiener equation and special methods will be needed to find the solution. Let the filter input be white noise with zero-mean and unit variance, so that

$$r_{xx}(k) = \delta(k)$$

For this input, the causal Wiener equations become,

$$\sum_{l=0}^{\infty} h(l)\delta(k-l) = r_{\mathrm{d}x}(k); \quad 0 \le k < \infty$$

$$h(k) = \begin{cases} r_{\mathrm{d}x}(k) & k \ge \infty \\ 0 & k < 0 \end{cases}$$

With the same input, but without the causality constraint, the Wiener equation would be

$$\sum_{l=-\infty}^{\infty} h(l) r_{xx}(k-l) = h(k) * r_{\mathrm{d}x}(k)$$

When the input to the Wiener filter is white, the optimal solution with a causality constraint is the same as the optimal solution without constraint, except that with the causality constraint, the impulse response is set to zero for negative time. Hence, to obtain the causal solution, the unconstrained two-sided Wiener solution is found and its non-causal part is removed in the time domain.

Usually, the input to the Wiener filter is not a white noise. Hence, it has to be whitened. A whitening filter is designed for this using a priori knowledge of the input autocorrelation function.We know that by symmetry property,

$$r_{xx}(k) = r_{xx}(-k)$$

and

$$P_x(z) = P_x(z^{-1})$$

Accordingly, there must be symmetry in the numerator and denominator factors.

$$P_x(z) = A\frac{(1-az^{-1})(1-az)(1-bz^{-1})(1-bz)\cdots}{(1-\alpha z^{-1})(1-\alpha z)(1-\beta z^{-1})(1-\beta z)\cdots}$$

Assuming that all the parameters $a, b \ldots \alpha, \beta, \ldots$ have magnitude less than one, then $P_x(z)$ is factorised as

$$P_x(z) = Q^+(z)Q^-(z)$$

where
$$Q^+(z) = \sqrt{A}\frac{(1-az^{-1})(1-bz^{-1})\cdots}{(1-\alpha z^{-1})(1-\beta z^{-1})\cdots}$$

$$Q^-(z) = \sqrt{A}\frac{(1-az)(1-bz)\cdots}{(1-\alpha z)(1-\beta z)\cdots}$$

All poles and zeros of $Q^+(z)$ will be inside the unit circle in the z-plane.

All poles and zeros of $Q^-(z)$ will be outside the unit circle in the z-plane.

$$Q^+(z) = Q^-(z^{-1})$$

$$Q^+(z^{-1}) = Q^-(z)$$

The whitening filter is designed by assuming

$$H(z) = \frac{1}{Q^+(z)}$$

To verify its whitening properties, let the input $x(n)$ with autocorrelation function $R_x(k)$ and its output $w(n)$ with autocorrelation function $r_{vv}(n)$. The transform of the output autocorrelation function is

$$P_v(z) = H(z)H(z^{-1})P_x(z) = \frac{1}{Q^+(z)}\frac{1}{Q^-(z)}P_x(z) = 1$$

Therefore, the autocorrelation function of the output is $r_{vv}(k) = \delta(k)$ and the output is white noise.

The whitening filter is causal and stable, since the zeros of $Q^+(z)$ are all inside the unit circle in the z-plane. The inverse of the whitening filter has a transfer function equal to $Q^+(z)$, which is stable and causal because $Q^+(z)$ has all of its poles inside the unit circle. To find $Y_{\text{opt}}(z)$ disregarding causality in terms of the autocorrelation of $x(n)$ and cross-correlation with $d(n)$.

The cascading of $\frac{1}{Q^+(z)}$ and $Y_{\text{opt}}(z)$ should be equal to the unconstrained Wiener filter is given by

$$H(z) = \frac{1}{Q^+(z)}Y_{\text{opt}}(z)$$

But
$$H(z) = \frac{P_{\text{dx}}(z)}{P_x(z)}$$

$$Y_{\text{opt}}(z) = \frac{P_{\text{dx}}(z)}{P_x(z)}Q^+(z) = \frac{P_{\text{dx}}(z)}{Q^+(z)Q^-(z)}Q^+(z) = \frac{P_{\text{dx}}(z)}{Q^-(z)}$$

Therefore,
$$H(z) = \frac{1}{Q^+(z)}\frac{P_{\text{dx}}}{Q^-(z)} \tag{13.112}$$

Thus the non-causal IIR Wiener filter that minimises the minimum mean square estimate of d(n) from the whitened signal is given in Eq.(13.112), the causal part of the non-causal IIR Wiener filter produces the causal IIR Wiener filter.

13.9.2 Kalman Filtering

The Kalman filter gives the optimal solution to many tracking and data-prediction tasks. It is used in the analysis of visual motion. The filter is constructed as a mean-squared error minimiser and an alternative derivation of the filter which relates to maximum likelihood statistics.The main aim of filtering is to extract the required information from a signal. The performance of the filter can be measured using a cost function or loss function. In fact, the goal of the filter is to minimise this loss function.

Mean-Squared Error

Let $x(n)$ be the transmitted message signal and $y(n)$ denotes the received signal. The received signal can be expressed as

$$y(n) = a(n)\, x(n) + w(n)$$

where $a(n)$ is the gain term and $w(n)$ is the additive noise. The difference between the estimate of $\hat{x}(n)$ and $x(n)$ is called the error.

$$f(e(n)) = f(x(n) - \hat{x}(n))$$

The particular shape of $f(e(n))$ is dependent upon the application and the function should be both positive and increase monotonically. An error function which exhibits these characteristics is the squared error function.

$$f(e(n)) = (x(n) - \hat{x}(n))^2$$

Since it is necessary to consider the ability of the filter to predict many data over a period of time, a more meaningful metric is the expected value of the error function.

$$\text{loss function } = E\left[f(e(n))\right] \tag{13.113}$$

Maximum Likelihood

Though the above derivation of mean-squared error is intuitive, it is also somewhat heuristic. A more rigorous derivation can be developed using maximum likelihood statistics to find $\hat{x}(n)$ which maximises the probability or likelihood of $y(n)$. Assuming that the additive random noise is Gaussian distributed with a standard deviation of $\sigma(n)$ which gives

$$P\left(y(n)\middle|\hat{x}(n)\right) = K(n)\exp\left[-\frac{\left(y(n) - a(n)\hat{x}(n)\right)^2}{2\sigma^2(n)}\right]$$

where $K(n)$ is a normalisation constant. The maximum likelihood function of this is given by

$$P(y|\hat{x}) = \prod_n K(n)\exp\left(-\frac{\left(y(n) - a(n)\hat{x}(n)\right)^2}{2\sigma^2(n)}\right)$$

Hence, $$\log P(y|\hat{x}) = -\frac{1}{2}\sum_n\left(\frac{\left(y(n) - a(n)\hat{x}(n)\right)^2}{2\sigma^2(n)}\right) + C \tag{13.114}$$

where C is a constant.

The driving function of Eq.(13.114) is the *Mean Squared Error* (*MSE*), which may be maximised by the variation of $\hat{x}(n)$. Therefore, the mean-squared error function is applicable when the expected variation of $y(n)$ is best modelled as a Gaussian distribution. In such a case, the MSE provides the value of $\hat{x}(n)$ which maximises the likelihood of the signal $y(n)$.

Kalman Filter Derivation

Assume that we want to know the value of a variable within a process of the form:

$$x(n+1) = \Phi\, x(n) + w(n)$$

where $x(n)$ is the state vector of the process at time n; Φ is the time varying state transition matrix (size $(n \times m)$) of the process from the state at n, to the state $n + 1$, and is assumed stationary over time, $w(n)$ is the associated white noise process with known covariance of size $(n \times m)$.

Observations on this variable can be modelled in the form of

$$z(n) = H\, x(n) + v(n) \tag{13.115}$$

where $z(n)$ is the actual measurement of x (size $(m \times 1)$) at time n; H is the noiseless connection (size $(m \times n)$) between the state vector and the measurement vector, and is assumed stationary over time $(m \times n)$; $v(n)$ is the associated measurement error. This is again assumed to be a white noise process with known covariance and has zero cross-correlation with the process noise, (size $(n \times 1)$).

For the minimisation of the Mean Squared Error (MSE) to yield the optimal filter, it must be possible to correctly model the system errors using Gaussian distributions. The covariances of the two noise models are assumed stationary over time and are given by

$$Q = E\,[w(n)w^T(n)] \tag{13.116}$$

$$R = E\,[v(n)v^T(n)] \tag{13.117}$$

The mean squared error is given by Eq.(13.113), which is equivalent to

$$E\,[e(n)e^T(n)] = P(n) \tag{13.118}$$

where $P(n)$ is the error covariance matrix at time n, (size $(n \times n)$). Equation (13.118) may be expanded to give

$$P(n) = E\left[e(n)\,e^T(n)\right] = E\left[\left(x(n)-\hat{x}(n)\right)\left(x(n)-\hat{x}(n)\right)^T\right] \tag{13.119}$$

Assuming the prior estimate of $\hat{x}(n)$ is called $\hat{x}^T(n)$ and was gained by knowledge of the system. It is possible to write an update equation for the new estimate, combining the old estimate with measurement data.

Thus,
$$\hat{x}(n) = \hat{x}(n) + K(n)\left(z(n) - H\,\hat{x}(n)\right) \tag{13.120}$$

where $K(n)$ is the Kalman gain. The term $z(n) - H\hat{x}(n)$ in the above equation is called the innovation or measurement residual;

$$i(n) = z(n) - H\,\hat{x}(n) \tag{13.121}$$

Substituting Eq.(13.115) into Eq.(13.120), we get

$$\hat{x}(n) = \hat{x}(n) + K(n)\left(H\,x(n) + v(n) - H\,\hat{x}(n)\right) \tag{13.122}$$

Substituting Eq.(13.122) in Eq.(13.119), we get

$$P(n) = E[(K - K(n)H)\,(x(n) - \hat{x}(n)) - K(n)v(n)\,(I - K(n)H)\,(x(n) - \hat{x}(n)) - K(n)v(n)]^T \tag{13.123}$$

At this point, it is noted that $x(n) - \hat{x}(n)$ is the error of the prior estimate. It is clear that this is uncorrelated with the measurement noise and hence the expectation may be rewritten as

$$P(n) = (I - K(n)H)\,E[(x(n) - \hat{x}'(n))(x(n) - \hat{x}'(n)^T)]$$

$$(I - K(n)H)^T + K(n)E[(v(n)v^T(n)]K^T(n)$$

$$P(n) = (I - K(n)H)\, P'(n)(I - K(n)H)^T + K(n)\, RK^T\,(n) \tag{13.124}$$

where $P'(n)$ is the prior estimate of $P(n)$. Equation (13.124) is the error covariance update equation. The diagonal of the covariance matrix contains the mean squared errors as given below:

$$P(n) = \begin{bmatrix} E\left[e(n-1)e^T(n-1)\right] & E\left[e(n)e^T(n-1)\right] & E\left[e(n+1)e^T(n-1)\right] \\ E\left[e(n-1)e^T(n)\right] & E\left[e(n)e^T\left(n\right)\right] & E\left[e(n+1)e^T(n)\right] \\ E\left[e(n-1)e^T(n+1)\right] & E\left[e(n)e^T(n+1)\right] & E\left[e(n+1)e^T(n+1)\right] \end{bmatrix}$$

The sum of the diagonal elements of a matrix is the trace of a matrix. In the case of the error covariance matrix, the trace is the sum of the mean squared errors. Therefore, the mean squared error may be minimised by minimising the trace of $P(n)$ which in turn will minimise the trace of $P(n)$.

The trace of $P(n)$ is first differentiated with respect to $K(n)$ and the result is set to zero in order to find the conditions of this minimum.

Expanding Eq.(13.124), we get

$$P(n) = P'(n) - K(n)HP'(n) - P'(n)H^T K^T(n)HP'(n)H^T + R)K^T(n) \tag{13.125}$$

Since the trace of a matrix is equal to the trace of its transpose, it may be written as

$$T\left[P(n)\right] = T\left[P'(n)\right] - 2T\left[K(n)HP'(n)\right] + T\left[K(n)\left(HP'(n)H^T + R\right)K^T(n)\right]$$

where $T[P(n)]$ is the trace of the matrix $P(n)$. Differentiating with respect to $K(n)$, we get

$$\frac{dT\left[P(n)\right]}{dK(n)} = -2\left(HP'(n)\right)^T + 2K(n)\left(HP'(n)H^T + R\right)$$

Setting to zero and rearranging, we get

$$\left[HP'(n)\right]^T = K(n)\left(HP'(n)H^T + R\right)$$

Upon solving for $K(n)$, we get

$$K(n) = P'(n)H^T\left(HP'(n)H^T + R\right)^{-1} \tag{13.126}$$

Equation (13.126) is the Kalman gain equation. The innovation, $i(n)$ defined in Eq.(13.121) has an associated measurement prediction covariance. It is defined as

$$S(n) = HP'(n)\,H^T + R$$

Finally, substituting Eq.(13.126) into Eq.(13.125), we get

$$P(n) = P'(n) - P'(n)H^T\left(HP'(n)H^T + R\right)^{-1} HP'(n) \tag{13.127}$$

$$= P'(n) - K(n)HP'(n) + \left(I - K(n)H\right)P'(n)$$

Equation (13.127) is the update equation for the error covariance matrix with optimal gain. The three Eq.(13.120), Eq.(13.126) and Eq.(13.127) develop an estimate of the variable $x(n)$. State projection is achieved by using

$$\hat{x}(n+1) = \Phi\,\hat{x}(n)$$

To complete the recursion, it is necessary to find an equation which projects the error covariance matrix into the next time instant, $(n + 1)$. This is achieved by first forming an expression for the prior error.

$$e'(n+1) = x(n+1) = \hat{x}'(n+1)$$
$$= \left(\Phi x(n) + w(n)\right) - \Phi\hat{x}(n) = \Phi e(n) + w(n)$$

Extending Eq.(13.119) to the next time interval $(n + 1)$ we get

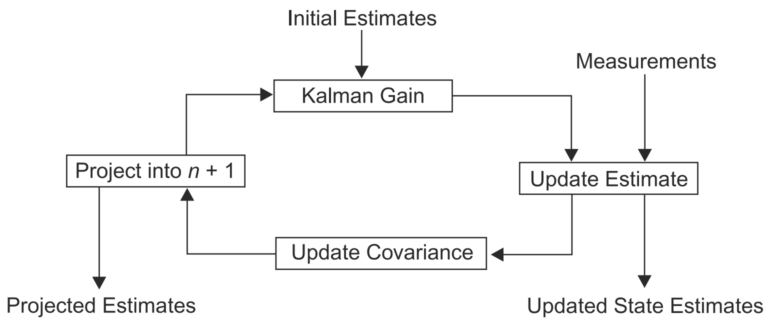
$$P'(n+1) = E\left|E'(n+1) + e^{T'}(n+1)\right|$$

Note that $e(n)$ and $w(n)$ have zero cross-correlation because the noise $w(n)$ actually accumulates between n and $(n + 1)$ whereas the error $e(n)$ is the error up to time, n.

Therefore,

$$P'(n+1) = E\left[e'(n+1)e^{T'}(n+1)\right]$$
$$= E\left[\Phi e(n)\left(\Phi e(n)^{T}\right)\right] + E\left[w(n)w^{T}(n)\right]$$
$$= \Phi P(n)\Phi^{T} + Q$$

This completes the recursive filter. The algorithmic loop is summaried in Fig. 13.8.



Description	Equation
Kalman Gain	$K(n) = P'H^{T}(HP'(n)H^{T} + R)^{-1}$
Update Estimate	$\hat{x}(n) = \hat{x}'(n) + K(n)(z(n) - H\hat{x}(n))$
Update Covariance	$P(n) = (I - K(n)H)\,P'(n)$
Project into $(n + 1)$	$P(n + 1) = \Phi P(n)\Phi^{T} + Q$

Fig. 13.8 *Kalman Filter Recursive Algorithm*

REVIEW QUESTIONS

13.1 What is known as a power spectrum?
13.2 What is the need for spectral estimation?
13.3 How can the energy density spectrum be determined?
13.4 What is autocorrelation function?
13.5 Differentiate between biased and unbiased estimators.
13.6 What is the relationship between autocorrelation and spectral density?
13.7 Give the estimate of autocorrelation function and power density for random signals.
13.8 Obtain the expression for mean and variance for the autocorrelation function of random signals.

13.9 Calculate the mean and variance for the autocorrelation function of random signals.
13.10 Give time and frequency domain representation for Bartlett window.
13.11 Define periodogram.
13.12 Explain how DFT and FFT are useful in power spectral estimation.
13.13 Explain power spectrum estimation using the Bartlett method.
13.14 What are the changes made in the Bartlett method to form the Welch method of power spectrum estimation?
13.15 Obtain the mean and variance of the averaging modified periodogram estimate.
13.16 Explain the periodogram method of power spectrum estimation.
13.17 Explain the Welch method of power spectrum estimation.
13.18 Justify how periodogram is an unbiased estimator.
13.19 How is the Blackman and Tukey method used in smoothing the periodogram?
13.20 Derive the mean and variance of the power spectral estimate of the Blackman and Tukey method.
13.21 How is power spectrum estimated in non-parametric methods?
13.22 How is power spectrum estimated in parametric methods?
13.23 List the various non-parametric methods of power spectrum estimation.
13.24 What are the limitations of non-parametric methods in spectral estimation?
13.25 How do the parametric methods overcome the limitations of the non-parametric methods?
13.26 Give the various steps involved in the parametric estimation process.
13.27 Explain the *AR*, *MA* and *ARMA* models.
13.28 Why is the AR model widely used?
13.29 What are the properties used to compare the power spectrum estimates of different method? Explain the properties.
13.30 Give the relationship between input and output power spectral density of a linear system.
13.31 Give the expression for cross-correlation.
13.32 Give the expression for power spectrum estimates of *AR, MA* and *ARMA* models.
13.33 Why are Wiener and Kalman filters called optimum filters?
13.34 What is linear prediction?
13.35 What is multistep linear prediction?
13.36 What is the difference between causal and non-causal IIR Wiener filter?
13.37 What is Wiener deconvolution?
13.38 The discrete time sequence is given by

$$x(n) = \sin 2\pi(0.12)n + \cos 2\pi(0.122)n,\ n = 0,1, ..., 15.$$

Evaluate the power spectrum $p(f) = \frac{1}{N}|X(f)|^2$ at the frequencies $f_k = k / L$, $k = 0,1,..., L-1$, for $L = 16, 32, 64$ and 128.
13.39 Using Welch method, calculate the variance of the Welch power spectrum estimate with the Bartlett window if there is 50% overlap between successive sequences.
13.40 Using the fourth joint moment for Gaussian random variables, show that

$$\text{cov}\left[P_{xx}(f_1)P_{xx}(f_2)\right] = \sigma_x^4\left\{\left[\frac{\sin\pi(f_1+f_2)N}{N\sin\pi(f_1+f_2)}\right]^2+\left[\frac{\sin\pi(f_1-f_2)N}{N\sin\pi(f_1-f_2)}\right]^2\right\}$$

under the condition that the sequence $x(n)$ is a zero mean white Gaussian noise sequence with variance σ_x^2.

13.41 For the AR process of order two, $x(n) = a_1 x(n-1) + a_2\, x(n-2) + w(n)$ where a_1 and a_2 are constants and $\{w(n)\}$ is a white noise process of zero mean and variance σ^2. Calculate the mean and autocorrelation of $\{x(n)\}$.

13.42 Determine the mean and the autocorrelation of the sequence $x(n)$ generated by the MA process described by the difference equation $x(n) = w(n) - aw(n-1) + bw(n-2)$ where $w(n)$ is a white noise process with variance σ^2.

13.43 Determine the power spectra for the random processes generated by $x(n) = w(n) - x(n-2)$ where $w(n)$ is a white noise process with variance σ^2.

13.44 Suppose we have $N = 500$ samples from a sample sequence of a random process.

(a) Determine the frequency resolution of the Bartlett, Welch (50% overlap), and Blackman–Tukey for a quality factor $Q = 12$.

(b) Determine the record length (M) for the Bartlett, Welch (50% overlap) and Blackman–Tukey methods.

13.45 The Bartlett is used to estimate the power spectrum of a signal from a sequence $x(n)$ consisting of $N = 3600$ samples.

(a) Determine the smallest length M of each segment in the Bartlett method that yields a frequency resolution of $\Delta f = 0.01$.

(b) Repeat part (a) for $\Delta f = 0.02$.

(c) Determine the Quality factor Q_B for parts (a) and (b).

13.46 The N-point DFT of a random sequence $x(n)$ is

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

Assume that $E[x(n)] = 0$ and $E(x(n)x(n+m)) = \sigma_x^2 \delta(m)$. Determine the variance and autocorrelation of $X(k)$.

13.47 Explain in detail the Levinson-Durbin algorithm.

13.48 Explain how *AR* parameters are obtained by using linear prediction.

13.49 Derive the power spectrum estimate using Burg method.

13.50 An *MA*(2) process is described by the difference equation $x(n) = w(n) + 0.81w(n-2)$ where $w(n)$ is a white noise process with variance $\sigma_w{}^2$.

(a) Determine the parameters of *AR*(2), *AR*(4) and *AR*(8) models that provide a minimum mean square error fit to the data $x(n)$.

(b) Plot the true spectra and those of the *AR*(p), $p = 2, 4, 8$ and compare the results. Comment on how well the *AR*(p) models approximate the *MA*(2) process.

13.51 Design a first order FIR Wiener filter to reduce the noise in $x(n) = \mathrm{d}(n) + v(n)$. Let $\mathrm{d}(n)$ be an *AR*(1) process with autocorrelation sequence. $r_{\mathrm{d}}(k) = a^{|k|}$, $0 < a < 1$ and $v(n)$ is the white noise with a variance of $\sigma_v{}^2$ which is uncorrelated with $\mathrm{d}(n)$.

13.52 Design a causal IIR Wiener filter to estimate a signal $\mathrm{d}(n)$ from the noisy observation $x(n) = v(n) + \mathrm{d}(n)$ where $v(n)$ is unit variance white noise process, $\mathrm{d}(n)$ is an *AR*(1) process given by $\mathrm{d}(n) = 0.9\mathrm{d}(n-1) + w(n)$ and $w(n)$ is a white noise with variance $\sigma_w^2 = 0.36$.

13.53 An autoregressive process of order 1 is given by the difference equation $x(n) = 0.9x(n-1) + w(n)$ where $w(n)$ is white noise with variance $\sigma_w^2 = 0.36$. Derive the Kalman filter state estimation equation, Kalman gain and error covariance.

Chapter 14 ADAPTIVE FILTERS

INTRODUCTION 14.1

In the previous chapter, an optimal linear filter was considered in a variety of applications. A filter is said to be optimal if it is designed with some knowledge about the input data. If this information is not known, then adaptive filters are used. The adaptive filter has the property of self-optimisation. It is a recursive time-varying filer characterised by a set of adjustable coefficients. An adaptive filter is useful whenever

(i) Signal properties are not known in advance or time varying,
(ii) System properties are not known in advance or time varying, and
(iii) Design requirements for fixed filters cannot easily be specified.

The adaptive filters find wide applications in system identification, speech and image encoding, channel equalisation, noise cancelling, line enhancing, frequency tracking, interference reduction and acoustic echo cancellation. Figure14.1 shows the basic schematic diagram of an adaptive filter, where $x(n)$, $\hat{d}(n)$, $d(n)$ and $e(n)$ are the input, output, desired output and error signal of the adaptive filter respectively at the time instant n. The adaptive filter estimates the output signal $\hat{d}(n)$ of the filter and compares it to a desired signal $d(n)$. By observing the error $e(n)$ between the output of the adaptive filter and the desired signal, i.e. $e(n) = d(n) - \hat{d}(n)$, an adaptive algorithm updates the filter coefficients with the aim of minimising the objective function. The adaptive filter shown in Fig.14.1 is a non-linear filter through its dependence on the input signal. But at a given instant of time, it will act as a linear filter. The adaptive filter is composed of three basic modules such as (i) filtering structure, (ii) adaptive algorithm, and (iii) performance criterion.

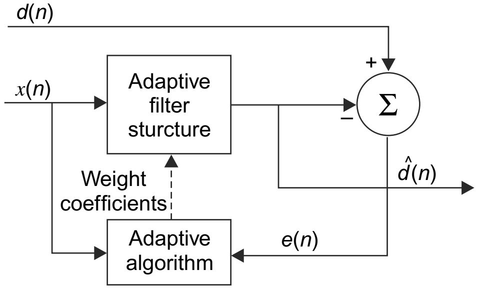


Fig. 14.1 *Schematic of an Adaptive Filter*

The characteristics of the adaptive filter are:

(i) It is described as a non-linear system with time-varying parameters.
(ii) The adjustable parameters are assigned with values depending on the estimated statistical nature of the signals. Hence, this filter is adaptable to the changing environment.
(iii) An algorithm is said to be stable if it converges to the minimum regardless of the starting point.
(iv) Since it is trained to perform specific filtering and decision-making tasks, the synthesis procedure is not required.

(v) The speed of convergence depends on the algorithm and the nature of the performance surface.
(vi) It is more complex and difficult to analyse.

The aim of an adaptive filter is to find and track the optimum filter quickly and accurately, corresponding to the same signal operating environment with the complete knowledge of the desired statistics. Its performance is calculated using the concepts of stability, quality and speed of adaptation, and tracking capabilities.

The majority of the adaptive algorithms originate from deterministic and gradient-based optimisation methods. The Steepest Descent Algorithm (SDA), Least Mean Square (LMS) Algorithm, Recursive Least Squares (RLS) algorithm and their properties and some practical applications are discussed in this chapter.

FILTERING STRUCTURE 14.2

The adaptive filter structure can be designed as non-recursive Finite Impulse Response (FIR) or recursive Infinite Impulse Response (IIR) filters where zeros and poles are on the unit circle. The output of an adaptive FIR filter is obtained as a linear combination of filter coefficients with the present and the $(N - 1)$ past input signal samples. The filter coefficients (or weights) are periodically updated by the adapting algorithm. Compared to IIR filters, the FIR filter is popularly used in the design of adaptive filters because it is simple, stable and has convergence characteristics and only adjustable zeros. Furthermore, many practical problems can be accurately modelled by an FIR filter to meet the design specifications. A direct form FIR filter with adjustable coefficients is shown in Fig. 14.2. The adaptive algorithm modifies the adaptive filter coefficients to improve the performance with the help of the performance criterion and the current signal. The performance criterion or objective function can be defined according to the given application. The value of the objective function at each iteration affects the update of the coefficients using the adaptive algorithm.

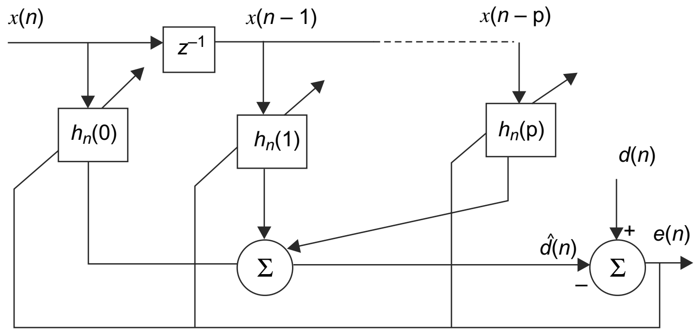


Fig. 14.2 *Direct Form Adaptive FIR Filter*

Let $\mathbf{x(n)} = [x(n), x(n-1), x(n-2), ..., x(n-p)]^T$

$\mathbf{H} = [h_n(0), h_n(1), ..., h_n(p)]^T$

The observed data $x(n)$ is passed through an FIR filter with weight vector $\mathbf{H}$ to obtain the estimate of the desired sequence, $\hat{d}(n)$. The filter output is an estimate of the desired signal $\hat{d}(n)$.

The adaptive filter starts with some random guess of the filter coefficients, then the coefficients are periodically updated on a sample-by-sample basis to minimise the difference between the filter output and a desired or target signal. The adaptation rule is given by

$$\hat{d}(n) = \sum_{k=0}^{p-1} h_n(k)x(n-k) = \mathbf{H^T x(n)} \tag{14.1}$$

$$h_k(n+1) = h_k(n) + \Delta h_k(n)$$

where $\Delta h_k(n)$ is the update on the filter coefficient h_k.

The estimation error is the difference between the estimated response $\hat{d}(n)$ and the desired response $d(n)$ as given by

$$e(n) = d(n) - \hat{d}(n) = d(n) - \sum_{k=0}^{p-1} h_n(k)x(n-k) \tag{14.2}$$

An objective function aimed to reduce the mean square error can be formulated as

$$\begin{aligned}\xi &= E\left[|e(n)|^2\right] = E\left[\left|d(n) - \sum_{k=0}^{p-1} h_n(k)x(n-k)\right|\right]^2 \\ &= E\left\lfloor d^2(n)\right\rfloor - 2E\left\lfloor d(n)x^T(n)\mathbf{H}\right\rfloor + E\left\lfloor \mathbf{H}^T x(n)x^T(n)\mathbf{H}\right\rfloor \\ &= \sigma_d^2 - 2\mathbf{P}^T\mathbf{H} + \mathbf{H}^T\mathbf{R}\mathbf{H}\end{aligned} \tag{14.3}$$

where $E[\cdot]$ denotes expectation, $\sigma_d^2 = E\left\lfloor d^2(n)\right\rfloor$ is the variance of $d(n)$, $\mathbf{P} = E[d(n)x(n)]$ is the cross-correlation vector between the desired signal and input signal, and $\mathbf{R} = E\left\lfloor x(n)x^T(n)\right\rfloor$ is the input signal autocorrelation matrix.

The objective function ξ is a quadratic function of the tap-weight coefficients which would allow a straightforward solution for $\mathbf{H}$ that minimise ξ if vector $\mathbf{P}$ and matrix $\mathbf{R}$ are known. A solution to this minimisation problem may be formed by setting differentiation of ξ with respect to $\mathbf{H}$.

$$\nabla = \frac{\partial \xi}{\partial \mathbf{H}} = -2\mathbf{P} + 2\mathbf{R}\mathbf{H}$$

For non-singular R, the optimal values for the tap-weight coefficients that minimise the objective function are computed as follows:

$$\mathbf{H}_{opt} = \mathbf{R}^{-1}\mathbf{P} \tag{14.4}$$

The above equation is called *Wiener–Hopf* equation. In practice, precise estimate of **R** and **P** are not available. When the input and the desired signals are ergodic, it is possible to use time averages to estimate **R** and **P**, that is implicitly performed by most adaptive algorithms.

If the optimal solution for **H** in the Mean Square Error (MSE) expression is replaced, the minimum *MSE* provided by the Wiener solution can be calculated. Therefore,

$$\begin{aligned}\xi_{\min}(n) &= E\left\lfloor d^2(n)\right\rfloor - 2\mathbf{H}_{opt}^T\mathbf{P} + \mathbf{H}_{opt}^T\ \mathbf{R}\mathbf{R}^{-1}\mathbf{P} \\ &= E\left\lfloor d^2(n)\right\rfloor - \mathbf{H}_{opt}^T\mathbf{P} = \sigma_d^2 - \mathbf{H}_{opt}^T\mathbf{P}\end{aligned} \tag{14.5}$$

Without computing **R** and **P** or performing a matrix inversion, adaptive algorithms can be used to compute $\mathbf{H}_{opt}$ on sample-by-sample basis for real-time applications.

THE STEEPEST DESCENT ALGORITHM (SDA) 14.3

The method of steepest descent is an iterative procedure that is used to find extrema of nonlinear functions. Let $h(n)$ be an estimate of the filter coefficient vector that minimises the mean-square error $\xi(n)$ at time n. At time $(n + 1)$, a new estimate is formed by adding a correction to $h(n)$ that is designed to bring $h(n)$ closer to the desired solution. The correction involves taking a step of size μ in the direction of maximum descent down the quadratic error surface, $\xi(n) = E\left[|e(n)|^2\right]$.

Taking gradient of the objective function, we get

$$\nabla\xi(n) = \begin{bmatrix} \dfrac{\partial\xi(n)}{\partial h(0)} \\ \dfrac{\partial\xi(n)}{\partial h(1)} \end{bmatrix}$$

The update equation for steepest descent algorithm is

$$h(n+1) = h(n) - \mu\nabla\xi(n)$$

$$= h(n) + \mu\left[-\frac{\partial E|e(n)|^2}{\partial h}\right] \tag{14.6}$$

where $\nabla e^*(n) = -x^*(n)$ and $\nabla\xi(n) = -E\lfloor e(n)x^*(n)\rfloor$. Therefore,

$$h(n + 1) = h(n) + \mu E\lfloor e(n)x^*(n)\rfloor \tag{14.7}$$

The step-size parameter 'μ', is a positive number. If it becomes excessively small, it represents slow movement of the coefficient vector towards the bottom of the quadratic error surface. An excessively large value of 'μ' may repeatedly overshoot the true bottom of the error surface, implying unstable filter. If $x(n)$ and $d(n)$ are jointly Wide Sense Stationary (WSS) processes, and the step size 'μ' with $0 < \mu < \dfrac{2}{\lambda_{max}}$ where λ_{max} represents the largest eigenvalue of the autocorrelation matrix $\mathbf{R_x}$, then the steepest-descent adaptation will converge to the optimum Wiener solution.

If $x(n)$ is a wide sense stationary (WSS) process,

$$E\lfloor e(n)x^*(n)\rfloor = E\lfloor d(n)x^*(n)\rfloor - E\lfloor h^T(n)x(n)x^*(n)\rfloor$$

$$= r_{dx} - \mathbf{R_x}h(n)$$

Therefore, the steepest descent algorithm can be written as

$$h(n+1) = h(n) + \mu\left(r_{dx} - \mathbf{R_x}h(n)\right) \tag{14.8}$$

The mean square value of the Wiener filter error signal defines the minimum mean square error of adaptive filters. However, not all adaptive filters converge to the solution of Wiener filter. The filter coefficients of adaptive filters might fluctuate around the Wiener filter coefficients and never equal to Wiener filter coefficients exactly. The mean square error surface with respect to the coefficients of an FIR filter is a quadratic bowl-shaped curve. It has a single global minimum that corresponds to the minimum value mean squared error. The values of the weight vector that corresponds to the bottom of the surface are the optimum Wiener coefficients. In most practical system where the second order moments r_d and r_h are unknown, the use of adaptive filters is the best choice.

The filter coefficients fluctuation introduces excess error to the error signal. Excess mean square error is the difference between the mean square error introduced by adaptive filters and the minimum mean square error produced by corresponding Wiener filters. Misadjustment is the ratio of the excess mean square error to the minimum mean square error.

Steps Involved in Steepest Descent Algorithm

(i) Initialise the steepest descent algorithm with an initial estimate $h(0)$ of the optimum weight vector **H**.

(ii) Evaluate the gradient of $\xi(n)$ at the current estimate $h(n)$ of the optimum weight vector.

(iii) Update the estimate at the time n by adding a correction that is formed by taking a step size μ in the negative gradient direction.

$$h(n+1) = h(n) - \mu \nabla \xi(n)$$

This is known as the adaptive filters weight update equation

(iv) Go back to (ii) and repeat the process.

Example 14.1 *The matrix* **R** *and the vector* **P** *are known for a given experimental environment:*

$$\mathbf{R} = \begin{bmatrix} 1 & 0.4045 \\ 0.4045 & 1 \end{bmatrix}, \qquad \mathbf{P} = [0 \quad 0.2939]^T \quad \text{and} \quad E\lfloor d^2(n) \rfloor = 0.5$$

Deduce the equation for the MSE.

Solution The MSE function is given by

$$\xi(n) = E\left[d^2(n)\right] - 2\mathbf{H}^T\mathbf{P} + \mathbf{H}^T\mathbf{RH}$$

$$= \sigma_d^2 - 2[h_1 \quad h_2]\begin{bmatrix} 0 \\ 0.2939 \end{bmatrix} + [h_1 \quad h_2]\begin{bmatrix} 1 & 0.4045 \\ 0.4045 & 1 \end{bmatrix}\begin{bmatrix} h_1 \\ h_2 \end{bmatrix}$$

Performing the algebraic calculations, we obtain the MSE equation as

$$\xi(n) = 0.5 + h_1^2 + h_2^2 + 0.8090 h_1 h_2 - 0.5878 h_2 \, .$$

Example 14.2 *The autocorrelation and the cross-correlation functions obtained in a particular situation are shown in Fig. E14.2. Calculate the minimum MSE for a filter with two coefficients. Assume one Eigen vector is* $\frac{1}{\sqrt{2}}[1 \quad 1]^T$.

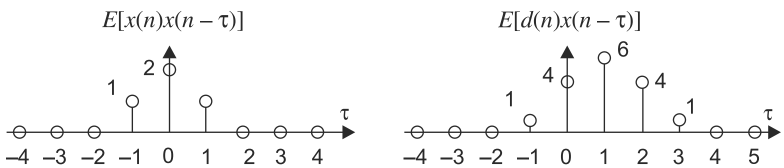


Fig. E14.2

Solution The autocorrelation matrix and cross-correlation matrix are

$$\mathbf{R} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad \text{and} \quad \mathbf{P} = \begin{bmatrix} 4 \\ 6 \end{bmatrix}$$

We know that, $\mathbf{H}_{opt} = \mathbf{R}^{-1}\mathbf{P}$

$$\mathbf{R}^{-1} = \frac{1}{3}\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

Finally, $$\mathbf{H}_{opt} = \frac{1}{3}\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}\begin{bmatrix} 4 \\ 6 \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 \\ 4 \end{bmatrix}$$

LEAST MEAN SQUARE (LMS) ALGORITHM 14.4

The most popular LMS algorithm, developed by Widrow and Hoff, is discussed in this section. It follows the stochastic gradient algorithms and possesses less computational complexity. The following are the most important properties of the LMS algorithm:

(i) LMS algorithm does not require the autocorrelation matrix of the filter input and the cross correlation between the filter input and its desired signal.
(ii) It does not use the *expectation* operation that is present in the steepest-descent method.
(iii) Implementation of the algorithm is simple and does not require matrix inversion.
(iv) Its iterative procedure involving:
   (a) Computation of the output of an FIR filter produced by a set of filter coefficients.
   (b) Generation of an estimated error by comparing the output of the filter to a desired response.
   (c) Adjustment of the filter coefficients based on the estimated error.
(v) It includes a step-size parameter, μ, which controls the stability and convergence of the algorithm.
(vi) It is a stable and robust algorithm.
(vii) The SDA contains deterministic quantities while the LMS operates on random quantities.

14.4.1 Derivation of LMS Algorithm

Using the steepest-descent method,

$$\begin{aligned} h(n+1) &= h(n) - \mu \nabla \xi(n) \\ &= h(n) + \mu \left[-\frac{\partial E\left[e^2(n)\right]}{\partial h} \right] \end{aligned} \tag{14.9}$$

where μ is the step size of the adaptation ($\mu < 1$). Here, lower the value of μ slower the convergence rate, higher the value of μ higher the chances of the adaptive system becoming unstable; $e(n)$ or $(d(n) - x(n))$ is the instantaneous error at time n and $x(n)$ is the sampled signal at time n.

Replacing the gradient of the mean square error function by a simple instantaneous squared error function, the LMS algorithm is defined for the coefficient update as

$$h(n+1) = h(n) + \mu \left[-\frac{\partial e^2(n)}{\partial h} \right] \tag{14.10}$$

$$\begin{aligned} \frac{\partial e^2(n)}{\partial h} &= \frac{\partial}{\partial h}\left(d(n) - h^T(n)x(n)\right)^2 \\ &= -2x(n)\left(d(n) - h^T(n)x(n)\right) = -2x(n)\,e(n) \end{aligned} \tag{14.11}$$

Substituting Eq. (14.11) in Eq. (14.10), we obtain

$$h(n+1) = h(n) + 2\mu x(n)\,e(n) \tag{14.12}$$

This is known as adaptive LMS algorithm. The algorithm requires $x(n)$, $d(n)$ and $h(n)$ to be known at each iteration. The summary of LMS adaptive algorithm is given in Table 14.1.

The LMS algorithm requires only $2N+1$ multiplications and $2N$ additions per iteration for an N tap-weight vector. Therefore, it has a relatively simple structure and the hardware required is directly proportional to the number of weights.

Table 14.1 *LMS Adaptive Algorithm*

Inputs :	M = filter length μ = step-size factor $x(n)$ = input data to the adaptive filter $x(0)$ = initialisation filter vector = 0
Outputs :	Adaptive filter output $y(n) = \mathbf{H}^T \boldsymbol{x}(\boldsymbol{n}) = \hat{d}(n)$ Error $e(n) = d(n) - y(n)$ $h(n+1) = h(n) + 2\,\mu e(n)\,x(n)$

14.4.2 Block-LMS Algorithm

The objective of this adaptive algorithm is to further reduce the computational burden in the computation of filter coefficients. The weight update equation is defined by

$$\mathbf{H}_{(j+1)L} = \mathbf{H}_{jL} + \frac{2\mu}{L}\sum_{l=0}^{L-1} e(jL+l)\,x(jL+l) \quad \text{for} \quad jL < n < (j+1)L: \quad \mathbf{H}_n = \mathbf{H}_{jL} \qquad (14.13)$$

where L is the blocksize in samples.

14.4.3 Sign-Error-LMS Algorithm

The weight-update equation for the sign-error-LMS algorithm is defined as

$$h(n+1) = h(n) + \mu \operatorname{sgn}\{e(n)\,x(n)\} \qquad (14.14)$$

where

$$\operatorname{sgn}\{e(n)\} = \begin{cases} 1, & \text{if } e(n) > 0 \\ 0, & \text{if } e(n) = 0 \\ -1, & \text{if } e(n) < 0 \end{cases}$$

Compared to the LMS algorithm, the sign-error LMS algorithm does not differ in the descent direction, but differs only in the effective step size.

14.4.4 Exponentially Weighted LMS Algorithm

The $\{x(n)\}$ and $\{d(n)\}$ have diminishing cross-correlation as the lag increases, and $e(n)$ will be of diminishing relevance for the increase in n. Hence, the estimates of those earlier errors $\{e(m), m \ll n\}$ should be weighted less than the estimates of the later errors in computing **H**. The weight-update equation is defined as

$$h(n+1) = h(n) - 2\mu \frac{\sum_{l=0}^{L} \lambda^l e(n-1)x(n-1)}{\sum_{l=0}^{L} \lambda^l} \qquad (14.15)$$

where λ^l is the eign value of the autocorrection matrix.

14.4.5 Normalised LMS Algorithm

The stability, convergence speed and fluctuation of the LMS algorithm depends on step size μ and the input signal power. The step size is inversely proportional to the filter length L and the signal power. One important technique to optimise the speed of convergence while maintaining the desired steady-state performance is the normalised LMS adaptive algorithm given by:

$$h(n+1) = h(n) + \mu(n)\,x(n)\,e(n) \qquad (14.16)$$

where $\mu(n)$ is a normalised step size and it is given by,

$$\mu(n) = \frac{\tilde{\mu}}{a + \|x(n)\|^2}$$

where $\|x(n)\|^2$ is the input signal energy and $\mu(n)$ controls the adaptation step size and 'a' is a constant

Characteristics of the Normalised LMS Algorithm

(i) Comparing to LMS above, the adaptation constant μ is dimensionless and the normalised LMS has the adaptation constant has the dimensioning of inverse power.

(ii) Normalised LMS algorithm can be viewed as an LMS algorithm with data-dependent adaptation step size.

(iii) The effect of large fluctuations in the power levels of the input signal is compensated at the adaptation level.

(iv) The normalised LMS algorithm is convergent in mean square sense, if $0 < \tilde{\mu} < 2$.

The normalised LMS (NLMS) introduces a variable that deals with the rate. It improves the convergence speed in a non-state environment. In another version, the Newton LMS, the weight-update equation includes whitening in order to achieve a single mode of convergence. For long adaptation processes the block LMS is used to make the LMS faster. In block LMS, the input signal is divided into blocks and weight are updated blockwise. A simple version of LMS is called the sign LMS. It uses the sign of the error to update the weights. Also LMS is not a blind algorithm, i.e. it requires a priori information for the reference signal.

14.4.6 Convergence and Stability of the LMS Algorithm

The LMS algorithm initialised with some arbitrary value for the weight vector converges and stays stable for $0 < \mu < 2/\lambda_{max}$, where λ_{max} is the largest eigenvalue of the correlation matrix R_x of the data. The convergence of the algorithm is inversely proportional to the eigenvalue spread of the correlation matrix R. When the eigenvalues of R are widespread, convergence may be slow. The eigenvalue spread of the correlation matrix is estimated by computing the ratio of the largest eigenvalue to the smallest eigenvalue of the matrix. In other words, if μ is chosen to be very small then the algorithm converges very slowly. However a large value of μ may lead to a faster convergence but may be less stable around the minimum value.

Example14.3 *Consider the single-weight adaptive filter shown in Fig. E14.3(a).*

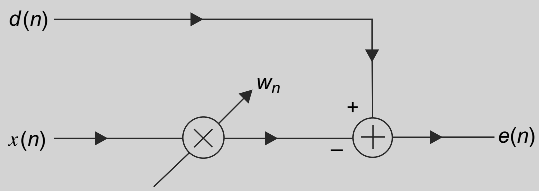


Fig. E14.3(a)

(a) Write down the LMS algorithm for updating the weight w.

(b) Suppose that x(n) is a constant:

$$x(n) = \begin{cases} k \;;\quad n \geq 0 \\ 0 \;;\quad Otherwise \end{cases}$$

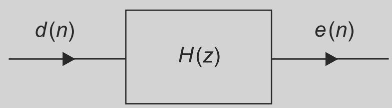


Fig. E14.3(b)

Find the system function H(z) shown in Fig. E14.3 (b), relating d(n) to e(n) using the LMS algorithm.

Solution The weight-update equation is given by

$$h(n+1) = h(n) + \mu e(n) z(n)$$

$$e(n) = d(n) - \hat{d}(n) = d(n) - k\, h(n)$$

$$h(n) = h(n-1) + \mu e(n-1)$$

$$e(n) - e(n-1) = d(n) - d(n-1) - k\left(h\left(n\right) - h\left(n-1\right)\right) = d(n) - d(n-1) - \mu k e(n-1)$$

Taking z-transform, we get

$$H(z)=\frac{E(z)}{D(z)}=\frac{1-z^{-1}}{1-\left(1-\mu k\right)z^{-1}}$$

RECURSIVE LEAST SQUARE (RLS) ALGORITHM 14.5

A limitation of the LMS algorithm is that it does not use all the information contained in the input data set for convergence. To overcome the problem of slow convergence of the LMS algorithm, the recursive least square (RLS) algorithm is implemented. The RLS algorithm uses the Newton's adaptation method to minimise the mean square surface. So, it computes recursively, the inverse of the mean square error function or inverse of the input correlation matrix. The RLS algorithm uses the matrix inversion lemma technique and it is a recursive implementation to minimise the LS objective function. The weighted least-squares (WLS) objective function is formulated as

$$\varepsilon(n)=\sum_{i=o}^{n}\lambda^{n-i}\left|e(i)\right|^2 \tag{14.17}$$

where $0<\lambda\le 1$ is an exponential scaling factor often referred to as forgetting factor which assigns greater importance to more recent data. With larger values of λ, the window length will be longer and will possess a longer memory. Bringing λ further away from 1, i.e. $\lambda \gg 1$ shortens the memory and enables the algorithm to track the statistical changes within the data.

Here, $$e(i)=y(i)-h_n^T x(i) \tag{14.18}$$

where $e(i)$ is the difference between desired signal $y(i)$ and filtered output at time i, using the latest set of filter co-efficient $h_n(k)$. Thus, minimizing $\varepsilon(n)$ by differentiating Eq. (14.17) with respect to $h_n^*(k)$, we have

$$\frac{\partial\varepsilon(n)}{\partial h_n^*(k)}=\sum_{i=0}^{n}\lambda^{n-i}e(i)\frac{\partial e^*(i)}{\partial h_n^*(k)}, \qquad \left[\text{since } \left|e(i)\right|^2=e(i)e^*(i)\right]$$

$$=-\sum_{i=0}^{n}\lambda^{n-i}e(i)x^*(i-k)=0 \quad \text{for } k=0,1,\cdots,p \text{ and } n>p \tag{14.19}$$

Substituting Eq. (14.18) in Eq. (14.19), we get

$$\sum_{i=0}^{n}\lambda^{n-i}\left\{y(i)-\sum_{l=0}^{p}h_n(l)x(i-1)\right\}x^*(i-k)=0$$

Expanding and rearranging, we get

$$\sum_{i=0}^{n}\lambda^{n-i}y(i)x^*(i-k)=\sum_{i=0}^{n}\lambda^{n-i}\sum_{l=0}^{p}h_n(l)x(i-l)x^*(i-k)=\sum_{l=0}^{p}h_n(l)\left[\sum_{i=0}^{n}\lambda^{n-i}x(i-l)x^*(i-k)\right]$$

or $$\sum_{l=0}^{p}h_n(l)\left[\sum_{i=0}^{n}\lambda^{n-i}x(i-l)x^*(i-k)\right]=\sum_{i=0}^{n}\lambda^{n-i}y(i)x^*(i-k)$$

In matrix form,

$$\mathbf{R}_{xx}(n)h(n)=r_{yx}(n)=r_{xy}(n)$$

Assuming that the autocorrelation matrix is invertible, we get

$$h(n)=\mathbf{R}_{xx}^{-1}(n)r_{xy}(n) \tag{14.20}$$

Now writing the recursion of the autocorrelation matrix over time, we get

$$r_{xx}(n) = \sum_{i=0}^{n} \lambda^{n-i} x(i)x^*(i)$$

$$r_{xx}(n+1) = \sum_{l=0}^{n+1} \lambda^{n+1-l} x(l)x^*(l)$$

$$= \sum_{l=0}^{n} \lambda^{n+1-l} x(l)x^*(l) + x(n+1)x^*(n+1) = \lambda \sum_{l=0}^{n} \lambda^{n-l} x(l)x^*(l) + x(n+1)x^*(n+1)$$

$$= \lambda r_{xx}(n) + x(n+1)x^*(n+1) \tag{14.21}$$

Similarly, the recursion of the cross-correlation vector over time, we get

$$\begin{aligned} r_{xy}(n+1) &= \sum_{l=0}^{n+1} \lambda^{n+1-l} x(l)y^*(l) \\ &= \lambda \sum_{l=0}^{n} \lambda^{n-l} x(l)y^*(l) + x(n+1)y^*(n+1) \\ &= \lambda r_{xy}(n) + x(n+1)y^*(n+1) \\ &= \lambda r_{yx}(n) + y(n+1)x^*(n+1) = r_{yx}(n+1) \end{aligned} \tag{14.22}$$

Matrix Inversion Lemma

The matrix inversion lemma or Woodbury's Identity is given by

$$\left[A + \boldsymbol{u}\boldsymbol{v}^H\right]^{-1} = A^{-1} - \frac{A^{-1}\boldsymbol{u}\boldsymbol{v}^H A^{-1}}{1 + \boldsymbol{v}^H A^{-1}\boldsymbol{u}}$$

where $\boldsymbol{u}$ and $\boldsymbol{v}$ are n-dimensional vectors.

Letting $A = \lambda \mathbf{R}_{xx}(n)$ and $\boldsymbol{u} = \boldsymbol{v} = \boldsymbol{x}(n+1)$, then we get the following recursion for the inverse of $\hat{\mathbf{R}}_{xx}(n+1)$

Substituting the lemma in the recursion in Eq. (14.21), we get

$$\hat{\mathbf{R}}_{xx}^{-1}(n+1) = \lambda^{-1}\hat{\mathbf{R}}_{xx}^{-1}(n) - \frac{\lambda^{-1}\hat{\mathbf{R}}_{xx}^{-1}(n)\boldsymbol{x}(n+1)\boldsymbol{x}^H(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\lambda^{-1}}{1 + \lambda^{-1}\boldsymbol{x}^H(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\boldsymbol{x}(n+1)}$$

Thus, $\mathbf{R}^{-1}(k)$ can be obtained recursively in terms of $\mathbf{R}^{-1}(k-1)$.

Recursion for the autocorrelation matrix is

$$\hat{\mathbf{R}}_{xx}^{-1}(n+1) = \lambda^{-1}\hat{\mathbf{R}}_{xx}^{-1}(n) - \frac{\lambda^{-2}\hat{\mathbf{R}}_{xx}^{-1}(n)\,\boldsymbol{x}(n+1)\,\boldsymbol{x}^H(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)}{1 + \lambda^{-1}\boldsymbol{x}^H(n+1)\,\hat{\mathbf{R}}_{xx}^{-1}(n)\,\boldsymbol{x}(n+1)} \tag{14.23}$$

The gain vector is defined by

$$\gamma(n+1) = \frac{\lambda^{-1}\hat{\mathbf{R}}_{xx}^{-1}(n)\,\boldsymbol{x}(n+1)}{1 + \lambda^{-1}\boldsymbol{x}^H(n+1)\,\hat{\mathbf{R}}_{xx}^{-1}(n)\,\boldsymbol{x}(n+1)} \tag{14.24}$$

Substituting Eq. (14.24) in Eq. (14.23), we get

$$\hat{\mathbf{R}}_{xx}^{-1}(n+1) = \lambda^{-1}\left[\hat{\mathbf{R}}_{xx}^{-1}(n) - \frac{\lambda^{-1}\hat{\mathbf{R}}_{xx}^{-1}(n)\boldsymbol{x}(n+1)\boldsymbol{x}^H(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)}{1 + \lambda^{-1}\boldsymbol{x}^H(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\boldsymbol{x}(n+1)}\right]$$

$$= \lambda^{-1}\left\lfloor \hat{\mathbf{R}}_{xx}^{-1}(n) - \gamma(n+1)\boldsymbol{x}^{H}(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\right\rfloor \tag{14.25}$$

Cross multiplying Eq. (14.24), we get

$$\gamma(n+1)\left\lfloor 1+\lambda^{-1}\boldsymbol{x}^{H}(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\boldsymbol{x}(n+1)\right\rfloor = \lambda^{-1}\hat{\mathbf{R}}_{xx}^{-1}(n)\boldsymbol{x}(n+1)$$

or, $$\gamma(n+1) = \lambda^{-1}\left[\mathbf{R}_{xx}^{-1}(n) - \gamma(n+1)\boldsymbol{x}^{H}(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\right]\boldsymbol{x}(n+1) \tag{14.26}$$

$$= \hat{\mathbf{R}}_{xx}^{-1}(n+1)\boldsymbol{x}(n+1)$$

where $\hat{\mathbf{R}}_{xx}^{-1}(n+1) = \lambda^{-1}\left[\mathbf{R}_{xx}^{-1}(n) - \gamma(n+1)\boldsymbol{x}^{H}(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\right]$

Thus the gain vector is the solution to linear equation. Therefore,

$$\hat{\mathbf{R}}_{\mathbf{xx}}(\mathbf{n+1})\gamma(n+1) = \boldsymbol{x}(\mathbf{n+1}) \tag{14.27}$$

The time update equation for the coefficient vector $\hat{h}(n)$ must be derived to complete the recursion. So, substituting $n = n + 1$ in Eq. (14.21), we get

$$\hat{h}(n+1) = \hat{\mathbf{R}}_{xx}^{-1}(n+1)\hat{r}_{xy}(n+1)$$

$$= \hat{\mathbf{R}}_{xx}^{-1}(n+1)\left[\lambda\hat{r}_{xy}(n) + \boldsymbol{x}(n+1)y^{*}(n+1)\right]$$

$$= \lambda\hat{\mathbf{R}}_{xx}^{-1}(n+1)\hat{r}_{xy}(n) + \boldsymbol{x}(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n+1)y^{*}(n+1)$$

Substituting Eq. (14.25) in the first term of the above equation and setting

$$\gamma(n+1) = \hat{\mathbf{R}}_{xx}^{-1}(n+1)x(n+1)$$

$$\hat{h}(n+1) = \left[\hat{\mathbf{R}}_{xx}^{-1}(n) - \lambda(n+1)\boldsymbol{x}^{H}(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\right]\hat{r}_{xy}(n) + \mathbf{x}(n+1)\gamma(n+1)$$

But $\hat{\mathbf{R}}_{xx}^{-1}(n)r_{xy}(n) = \hat{h}(n)$

Therefore, $$\hat{h}(n+1) = \hat{h}(n) + \lambda(n+1)\left\lfloor x(n+1) - \hat{h}^{H}(n)y(n+1)\right\rfloor$$

$$\hat{h}(n+1) = \hat{h}(n) + \alpha(n+1)\lambda(n+1) \tag{14.28}$$

where $\boldsymbol{x}(n+1) = \boldsymbol{x}(n+1) - \hat{h}^{H}(n)y(n+1)$ is the a priori error which is the difference between $\boldsymbol{x}(n+1)$ and estimate of $\boldsymbol{x}(n+1)$ that is formed by applying filter coefficients $\hat{h}(n)$ to data vector $y(n+1)$.

Parameters

P = filter order
λ = weighting factor
δ = value used to minimise $\hat{\mathbf{R}}_{xx}^{-1}(0)$

Initialisation

$h[0] = 0$

$\hat{\mathbf{R}}_{xx}^{-1}[0] = \delta^{-1}I$

Operation: For n = 1, 2, ...

1. Gain vector :

$$\gamma(n+1) = \frac{\lambda^{-1}\mathbf{R}_{xx}^{-1}(n)\boldsymbol{x}(n+1)}{1+\lambda^{-1}\boldsymbol{x}^{H}(n+1)\hat{\mathbf{R}}_{xx}^{-1}(n)\boldsymbol{x}(n+1)}$$

2. $\alpha(n+1) = \boldsymbol{x}(n+1) - \hat{h}^H(n)y(n+1)$
3. *Filter parameter:*
   $\hat{h}(n+1) = \hat{h}(n) + \alpha(n+1)\lambda(n+1)$
4. *Autocorrelation matrix:*
   $$\hat{\mathbf{R}}_{xx}^{-1}(n+1) = \frac{1}{\lambda}\left[\hat{\mathbf{R}}_{xx}^{-1}(n) - \gamma(n+1)\left(\hat{\mathbf{R}}_{xx}^{-1}(n)y(n+1)\right)^H\right]$$

14.5.1 Choice of a Particular Adaptive Algorithm

The choice of a particular adaptive algorithm depends on the following parameters.

1. Rate of convergence, i.e. the rate at which the algorithm reaches an optimum solution
2. Steady-state error
3. Ability to track statistical variations in the input data
4. Computational complexity
5. Ability to operate with ill-conditioned input data
6. Sensitivity to variations in the input

14.5.2 Comparison of LMS with RLS

1. The main drawback of the RLS algorithm is its computational complexity. But in general, the RLS algorithm is preferable over the LMS algorithm.
2. The convergence rate of the LMS algorithm is governed by the choice of the step size μ, where increasing μ increases the convergence.
3. The convergence rate of the RLS algorithm is independent of the spread of eigenvalues within the input correlation matrix. This is not the case for the LMS algorithm, as when the eigenvalue spread of the correlation matrix is greater, then the converges is rather slow.
4. The RLS algorithm incorporates a forgetting factor λ, governs the rate of convergence and steady-state error. In a stationary environment the best steady-state performance results from slow adaptation where λ is close to 1. Conversely, smaller values of λ result in faster convergence but greater steady-state error. This is similar to memory-convergence relationship existing within the LMS algorithm, which is determined by the step size μ.

THE MINIMUM MEAN SQUARE ERROR CRITERION 14.6

Consider the data sequence $\{x(n)\}$ which is a stationary random process. The autocorrelation sequence is

$$r_{xx}(m) = E\left[x(n)x^*(n-m)\right] \tag{14.29}$$

The observed data $x(n)$ is passed through an FIR filter to obtain the estimate of the desired sequence $\{d(n)\}$. The filter output which is the estimate of $d(n)$ is given by

$$\hat{d}(n) = \sum_{k=0}^{M-1} h(k)x(n-k) \tag{14.30}$$

The estimate error is

$$\begin{aligned} e(n) &= d(n) - \hat{d}(n) \\ &= d(n) - \sum_{k=0}^{M-1} h(k)x(n-k) \end{aligned} \tag{14.31}$$

The mean square error is given by

$$\xi(n) = E\left[\left|e(n)\right|^2\right]$$
$$= E\left[\left|d(n) - \sum_{k=-\infty}^{\infty} h(k)x(n-k)\right|^2\right]$$
$$= E\left\{\left|d(n)\right|^2 - 2\,\mathrm{Re}\left\{\sum_{l=0}^{M-1} h^*(l)d(n)x^*(n-1)\right\} + \sum_{k=0}^{M-1}\sum_{l=0}^{M-1} h^*(l)h(k)x^*(n-l)x(n-k)\right\}$$
$$= \sigma_d^2 - 2\,\mathrm{Re}\left[\sum_{l=0}^{M-1} h^*(l)\gamma_{dx}(l)\right] + \sum_{k=0}^{M-1}\sum_{l=0}^{M-1} h^*(l)h(k)\gamma_{xx}(l-k) \tag{14.32}$$

where $\sigma_d^2 = E\left[\left|d(n)\right|^2\right]$, $\gamma_{dx}(l) = E\left[d(n)x^*(n-1)\right]$, h_M = vector coefficients

Minimisation of ξ

Method 1 Minimise ξ with respect to the coefficient yields

$$\sum_{k=0}^{M-1} h(k)\gamma_{xx}(l-k) = \gamma_{xx}(l),\ \ l = 0, 1, \ldots, M-1.$$

This is equivalent to the Wiener–Hopf equation and the filter is called Wiener filter. The above equation is expressed in matrix form as,

$$\Gamma_M h_M = \gamma_d$$

where Γ_M is a $M \times M$ (Hermitian) Toeplitz matrix with elements $\Gamma_{lk} = \gamma_{xx}(l-k)$, γ_d is the $M \times 1$ cross-correlation vector with elements $\gamma_{xx}(l), l = 0,1,\ldots, M-1$, and h_M are the filter coefficients.

The optimum solution is

$$h_{opt} = \Gamma_M^{-1}\gamma_d$$

The minimum mean square error is

$$\xi_{\min} = \sigma_d^2 - \sum_{k=0}^{M-1} h_{opt}(k)\gamma_{dx}^*(k)$$
$$= \sigma_d^2 - \gamma_{dx}^{*t}\Gamma_M^{-1}\gamma_{dx} \tag{14.33}$$

Method 2 If the error $e(n)$ is orthogonal to the data in the estimate $d(n)$, then the orthogonality principle can be applied to minimise the mean square error.

$$E\left[e(n)x^*(n-1)\right] = 0;\ \ l = 0,1,\ldots M-1 \tag{14.34}$$

Substituting the value of $e(n)$, we get

$$E\left[\left\{d(n) - \sum_{k=0}^{M-1} h(k)x(n-k)\right\}x^*(n-l)\right] = 0$$

$$E\left[d(n)x^*(n-l) - \sum_{k=0}^{M-1} h(k)\,x(n-k)x^*(n-l)\right] = 0$$

$$E\left[\sum_{k=0}^{M-1} h(k)x(n-k)x^*(n-l)\right] = E\left[d(n)x^*(n-l)\right]$$

$$\sum_{k=0}^{M-1} h(k) E\left[x(n-k)x^*(n-l)\right] = E\left[d(n)x^*(n-l)\right]$$

$$\sum_{k=0}^{M-1} h(k)\gamma_{xx}(l-k) = \gamma_{dx}(l); \;\; l = 0, 1, ..., M-1 \tag{14.35}$$

The minimum mean square error is

$$\begin{aligned}\xi_{\min} &= E\left[e(n)d^*(n)\right] \\ &= E\left[\left\{d(n) - \sum_{k=0}^{M-1} h_{opt}(k)x(n-k)\right\}d^*(n)\right] \\ &= E\left[\left|d(n)\right|^2 - \sum_{k=0}^{M-1} h_{opt}(k)\gamma_{dx}(k)\right]\end{aligned} \tag{14.36}$$

where $h_{opt} = \Gamma_M^{-1}\gamma_d$

14.7 THE FORWARD–BACKWARD LATTICE METHOD

The m^{th} stage of the M^{th} order lattice filter is shown in Fig. 14.3. The equations for the filter can be written as

$$f_m(i) = f_{m-1}(i) + \gamma_m^{(f)} b_{m-1}(i-1)$$

$$b_m(i) = b_{m-1}(i-1) + \gamma_m^{(b)} f_{m-1}(1) \quad (14.37)$$

where $m = 1, 2, ..., M$

$\gamma_m^{(f)}$ = forward reflection coefficient of the m^{th} stage

$\gamma_m^{(b)}$ = backward reflection coefficient of the m^{th} stage

f_m = forward prediction error and

b_m = backward prediction error and

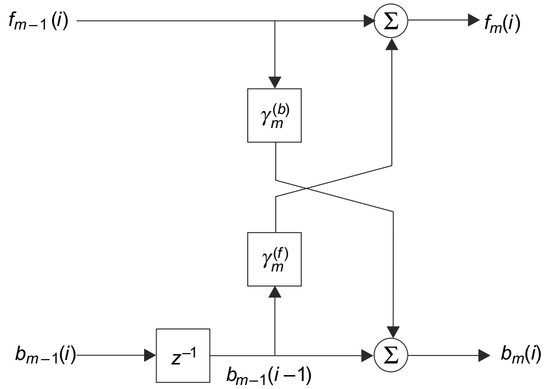


Fig. 14.3 *m^{th} stage of M^{th} Order Lattice Filter*

Minimum Forward Prediction Error

The mean square value of $f_m(i)$ is

$$\begin{aligned}\varepsilon_m^{(f)} &= E\left[f_m^2(i)\right] \\ &= E\left[f_m^2(i)\right] + \left(\gamma_m^{(f)}\right)^2 E\left[b_{m-1}^2(i-1)\right] + 2\gamma_m^{(f)} E\left[f_{m-1}(i)b_{m-l}(i-1)\right]\end{aligned} \tag{14.38}$$

The mean squared forward prediction error $f_M(i)$ can be minimised by differentiating $\varepsilon_m^{(f)}$ with respect $\gamma_m(f)$ and equating to zero. Therefore,

$$\gamma_{m,0}^{(f)} = -\frac{E\left[f_{m-1}(i)b_{m-1}(i-1)\right]}{E\left[b_{m-1}^2(i-1)\right]} \tag{14.39}$$

Minimum Backward Prediction Error

$$\begin{aligned}\varepsilon_m^{(b)} &= E\left[b_m^2(i)\right] \\ &= E\left[b_m^2(i-1)\right] + \left(\gamma_m^{(b)}\right)^2 E\left[f_{m-1}^2(i)\right] + 2\gamma_m^{(b)} E\left[b_{m-1}(i-1)f_{m-l}(i)\right]\end{aligned} \tag{14.40}$$

The optimal value of backward reflection coefficient $\gamma_m^{(b)}$ is obtained by differentiating the mean square value of the backward prediction error $b_m(i)$ with respect to $\gamma_m^{(b)}$ and equating it to zero. The result is

$$\gamma_{m,0}^{(b)} = -\frac{E\left[f_{m-1}(i)b_{m-1}(i-1)\right]}{E\left[f_{m-1}^2(i-1)\right]} \tag{14.41}$$

Time Update Recursion

If the filter input is stationary, then the reflection coefficients are set to their optimal values at each stage

$$E\left[f_m^2(i)\right] = E\left[b_m^2(i-1)\right],\ 1 \le m \le M \tag{14.42}$$

Therefore, $$\gamma_{m,0}^{(f)} = \gamma_{m,0}^{(b)} \tag{14.43}$$

If the filter input is non-stationary, the reflection coefficients have unequal values. The estimators for the expectations assuming that n samples are considered.

Expectation	Estimator
$E\left[f_{m-1}(i)b_{m-1}(i-1)\right]$	$\frac{1}{n}\sum_{i=1}^{n} w^{n-i} f_{m-1}(i)b_{m-1}(i-1)$
$E\left[f_{m-1}^2(i)\right]$	$\frac{1}{n}\sum_{i=1}^{n} w^{n-i} f_{m-1}^2(i)$
$E\left[b_{m-1}^2(i-1)\right]$	$\frac{1}{n}\sum_{i=1}^{n} w^{n-i} b_{m-1}^2(i-1)$

The weighting factor w lies in the range $0 < w < 1$, If the input is stationary, then $w = 1$.

Estimates for Forward and Backward Reflection Coefficients

The estimates are given by

$$\gamma_m^{(f)}(n) = -\frac{K_{m-1}(n)}{E_{m-1}^{(b)}(n-1)} \tag{14.44}$$

$$\gamma_m^{(b)}(n) = -\frac{K_{m-1}(n)}{E_{m-1}^{(f)}(n-1)} \tag{14.45}$$

where $$K_{m-1}(n) = \sum_{i=1}^{n} w^{n-i} f_{m-1} b_{m-1}(i-1)$$

$$E_{m-1}^{(f)}(n) = \sum_{i=1}^{n} w^{n-i} f_{m-1}^2(i)$$

and $$E_{m-1}^{(b)}(n) = \sum_{i=1}^{n} w^{n-i} b_{m-1}^2(i)$$

Since the filter input is zero for $i \le 0$,

$$\sum_{i=1}^{n} w^{n-1-i} b_{m-1}^2(i) = \sum_{i=1}^{n} w^{n-i} b_{m-1}^2(i-1) \tag{14.46}$$

Hence, $$\begin{aligned} K_{m-1}(n) &= \sum_{i=1}^{n-1} w^{n-i} f_{m-1}(i)b_{m-1}(i-1) + f_{m-1}(n)b_{m-1}(n-1) \\ &= \sum_{i=1}^{n-1} w^{n-1-i} f_{m-1}(i)b_{m-1}(i-1) + f_{m-1}(n)b_{m-1}(n-1) \\ &= wK_{m-1}(n-1) + f_{m-1}(n)b_{m-1}(n-1) \end{aligned} \tag{14.47}$$

Similarly, $$E_{m-1}^{(f)}(n) = wE_{m-1}^{(f)}\, f_{m-1}^2(n) \tag{14.48}$$

and $$E_{m-1}^{(b)}(n) = wE_{m-1}^{(b)}\, b_{m-1}^{2}(n) \tag{14.49}$$

Order Update Recursions

The prediction errors are given by

$$\begin{aligned} f_m(i) &= f_{m-1}(i) + \gamma_m^{(f)}(n) b_{m-1}(i-1), \quad 1 \le i \le n \\ b_m(i) &= b_{m-1}(i) + \gamma_m^{(b)}(n) f_{m-1}(i), \quad 1 \le i \le n \end{aligned} \tag{14.50}$$

where $\gamma_m^{(f)}(n)$ = forward reflection coefficient of the m^{th} stage

$\gamma_m^{(b)}(n)$ = backward reflection coefficient of the m^{th} stage

and $\gamma_m^{(f)}(n)$ and $\gamma_m^{(b)}(n)$ are considered as constants for the time interval $1 \le i \le n$.

Estimate for Forward and Backward Prediction Error

$$\begin{aligned} E_m^{(f)}(n) &= \sum_{i=1}^{n} w^{n-i} f_m^2(i) \\ &= \sum_{i=1}^{n} w^{n-i} \left[f_{m-1}(i) + \gamma_m^{(f)}(n) b_{m-1}(i-1) \right]^2 \\ &= \sum_{i=1}^{n} w^{n-i} f_{m-1}^2(i) + 2\gamma_m^{(f)}(n) \sum_{i=1}^{n} w^{n-i} f_{m-1}(i)\, b_{m-1}(i-1) + \left[\gamma_m^{(f)}(n)\right]^2 \\ &\quad \sum_{i=1}^{n} w^{n-i} b_{m-1}^2(i-1) \\ &= E_{m-1}^{(f)} - \frac{2K_{m-1}(n)}{E_{m-1}^{(b)}(n-1)} K_{m-1}(n) + \left[\frac{K_{m-1}(n)}{E_{m-1}^{(b)}(n-1)} \right] E_{m-1}^{(b)}(n-1) \\ &= E_{m-1}^{(f)}(n) - \frac{K_{m-1}^2(n)}{E_{m-1}^{(b)}(n-1)} \end{aligned} \tag{14.51}$$

Similarly, $$E_m^{(f)}(n) = E_m^{(b)}(n) = E_{m-1}^{(f)}(n-1) - \frac{K_{m-1}^2(n)}{E_{m-1}^{(f)}(n)} \tag{14.52}$$

GRADIENT ADAPTIVE LATTICE METHOD 14.8

Let $\varepsilon_m(n)$ be defined as the sum of the mean square values of the forward and backward prediction errors at the output of the m^{th} stage lattice filter at time n and $\gamma_m(n)$ be the reflection coefficient

$$\varepsilon_m(n) = E\left[f_m^2(n) \right] + E\left[b_m^2(n) \right] \tag{14.53}$$

where $$\begin{aligned} f_m(n) &= f_{m-1} + \gamma_m(n) b_{m-1}(n-1) \\ b_m(n) &= b_{m-1}(n-1) + \gamma_m(n) f_{m-1}(n) \end{aligned} \tag{14.54}$$

$$\begin{aligned} \varepsilon_m(n) &= E\left[f_{m-1}^2(n) \right] + \gamma_m^2(n) E\left[b_{m-1}^2(n-1) \right] + 2\gamma_m(n) E\left[f_{m-1}(n) b_{m-1}(n-1) \right] \\ &\quad + E\left[b_{m-1}^2(n-1) \right] + \gamma_m^2(n) E\left[f_{m-1}^2(n) \right] + 2\gamma_m(n) E\left[b_{m-1}(n-1) f_{m-1}(n) \right] \\ &= \left\{ E\left[f_{m-1}^2(n) \right] + E\left[b_{m-1}^2(n-1) \right] \right\} \left[1 + \gamma_m^2(n) \right] + 4\gamma_m(n) E\left[b_{m-1}(n-1) f_{m-1}(n) \right] \end{aligned} \tag{14.55}$$

where $\varepsilon_m(n)$ depends on time n because the reflection coefficient $\gamma_m(n)$ varies with respect to time, $f_{m-1}(n-1)$ and $b_{m-1}(n)$ are the forward predictor error and the delayed backward prediction error respectively at the input of the m^{th} stage.

The gradient is given by

$$\nabla_m(n) = \frac{\partial \varepsilon_m(n)}{\partial \gamma_m(n)}$$

$$= 2\gamma_m(n) E\left[f_{m-1}^2(n) + b_{m-1}^2(n-1)\right] + 4E\left[f_{m-1}(n) b_{m-1}(n-1)\right] \quad (14.56)$$

The instantaneous estimate for the gradient $\Delta_m(n)$ is

$$\hat{\nabla}_m(n) = 2\gamma_m(n)\left[f_{m-1}^2(n) + b_{m-1}^2(n-1)\right] + f_{m-1}(n) b_{m-1}(n-1) \quad (14.57)$$

The time update recursion for the m^{th} stage is

$$\gamma_m(n+1) = \gamma_m(n) - \frac{1}{2}\mu_m(n)\hat{\nabla}_m(n)$$

$$= \gamma_m(n) - \mu_m(n)\left\{\gamma_m(n)\left[f_{m-1}^2(n) + b_{m-1}^2(n-1)\right] + 2f_{m-1}(n) b_{m-1}(n-1)\right\} \quad (14.58)$$

where $\mu_m(n)$ is the varying step size parameter

Let $$\mu_m(n) = \frac{\beta}{E_{m-1}(n)} \quad (14.59)$$

where β is a constant with $0 < \beta < 2$ and $E_{m-1}(n)$ is an estimate value in Eq. (14.58).

An estimate which is frequently used is

$$E_m(n-1) = (1-w)\sum_{k=0}^{n} w^{n-k}\left[f_{m-1}^2(n-1) + b_{m-1}^2(n-1)\right] \quad (14.60)$$

where w is an exponential weighting factor with $0 < w < 1$. An advantage of this estimate is that $E_{m-1}(n)$ may be evaluated efficiently with the recursion

$$E_{m-1}(n) = wE_{m-1}(n-1) + (1-w)\left[f_{m-1}^2(n-1) + b_{m-1}^2(n-1)\right]$$

This procedure is referred to as the Gradient Adaptive Lattice (GAL) algorithm.

APPLICATIONS OF ADAPTIVE FILTERING 14.9

In this section, a brief introduction to the typical applications of adaptive filter in real world problem is presented. Due to the ability of adaptive filter to operate satisfactorily in non-stationary environments, it is considered as a part of many DSP applications where the statistics of the incoming signals are unknown or time varying. Adaptive filter performs a range of varying tasks, namely, system identification, noise cancellation, signal prediction, etc. in applications like channel equalization, echo cancellation or adaptive beam forming. Each application differs in the way the adaptive filter chooses the input signal and the desired signal.

14.9.1 System Identification

Consider the set-up shown in Fig. 14.4 for the system identification application. The requirement is to develop an adaptive FIR

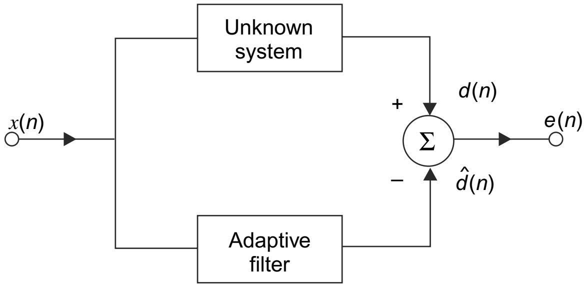


Fig. 14.4 *System Identification*

filter with adjustable coefficients. Let $x(n)$ be the common input signal to excite the unknown system and the FIR filter, $d(n)$ and $\hat{d}(n)$ be the output of the dynamic system and FIR filter respectively. Assume that the unknown system has an impulse response given by $h(k)$ for $k = 0, 1, 2, ..., N$ and zero for $k < 0$.

$$\hat{d}(n) = \sum_{k=0}^{M-1} h(k)\,x(n-k) \tag{14.61}$$

The error signal is given by

$$e(n) = d(n) - \hat{d}(n) = \sum_{l=0}^{N-1} h(l)x(k-1) - \sum_{i=0}^{N-1} h(i)x(k-i) \tag{14.62}$$

For calculating the coefficients of the adaptive filter, the mean square error criterion is used. The error signal has to be minimised using the objective function

$$\min \xi(n) = \min E\left(e^2(n)\right) = \sum_{n=0}^{\infty}\left[d(n) - \sum_{k=0}^{M-1} h(k)x(n-k)\right] \tag{14.63}$$

and the filter coefficients are selected. Here, the adaptive filter aim is to model the unknown system. Some real world applications of the system identification problem include modelling of multipath channel, control systems, seismic exploration, etc.

14.9.2 Adaptive Noise Cancellation

Consider the adaptive filter setup as shown in Fig. 14.5 for noise-cancellation application. The reference signal recorded by a primary microphone consists of a desired signal $x(n)$ which is corrupted by an additive noise $v_1(n)$. The input signal of the adaptive filter is a noise signal, $v_2(n)$ recorded by a secondary microphone that is correlated with the interference signal $v_1(k)$, but uncorrelated with $x(n)$. Some real-world examples of noise cancellation with adaptive filters are acoustic echo cancellation in hearing aids, noise cancellation in hydrophones, power line interference cancellation in electrocardiography, etc.

In this application, the error signal is given by

$$e(n) \quad x(n) \quad v_1(n) \quad \hat{d}(n)$$

The resulting MSE is then given by

$$E[e^2(n)] = E[x^2(n)] + E\left\{[v_1(n) - \hat{d}(n)]^2\right\}$$

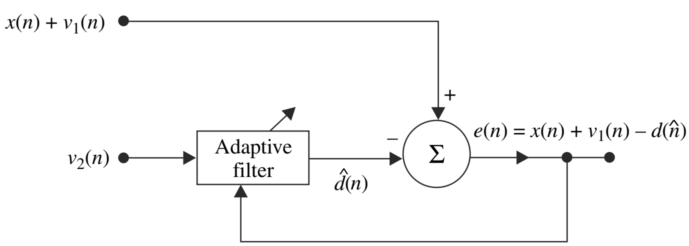


Fig. 14.5 *Noise Cancellation Using Adaptive Filter*

where minimising the MSE is equivalent to minimising the second term $E\{[v_1(n) - \hat{d}(n)]^2\}$. Then, the adaptive filter having $v_2(n)$ as the input signal is able to perfectly predict the signal $v_1(n)$. The minimum MSE is given by $E[e^2(n)]$, where the error signal in this situation is the desired signal $x(n)$. The effectiveness of the noise cancellation scheme depends on the high correlation between $v_1(n)$ and $v_2(n)$.

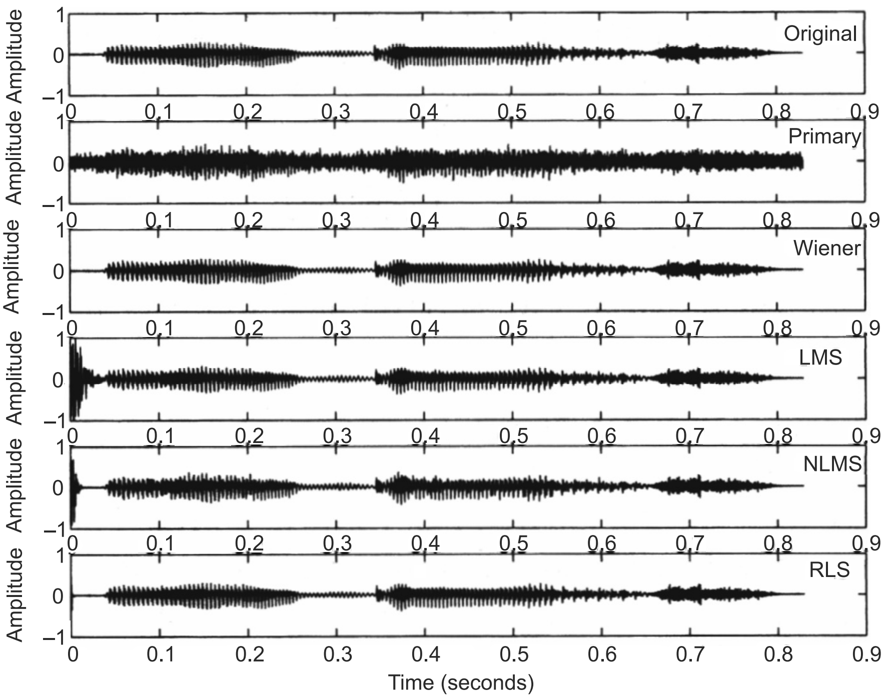


Fig. 14.6 *Performance of Adaptive Algorithms in Noise Cancellation*

Figure 14.6 shows the convergence performance of various algorithms in noise cancellation. Among the three algorithms considered, LMS shows the slow convergence. An improved convergence is achieved by the NLMS algorithm. RLS performs the best in terms of convergence compared to LMS and NLMS algorithms. It can be further added that the theoretical performance of noise cancellation depends on the following factors:

(i) Signal-to-noise ratio of the primary microphone.
(ii) Coherence between the noise in primary and the secondary microphones.

14.9.3 Adaptive Equalisation

It consists of estimating transfer function to compensate for the distortion introduced by the channel. This is also known as inverse filtering. The equaliser $\{H(z)\}$ structure is shown in Fig.14.7. Symbols $\{x(n)\}$ are transmitted through the channel and corrupted by additive complex-valued white noise $\{v(n)\}$. The received signal $\{u(n)\}$ is processed by the FIR equaliser to generate estimates $\{\hat{d}(n)\}$, which are fed into a decision device. The equaliser possesses the following two modes of operation:

(i) A training mode during which a delayed replica of the input sequence is used as a reference sequence, and
(ii) A decision directed mode during which the output of decision-device replaces the reference sequence. The purpose of using adaptive equaliser is to compensate the channel distortion so that the detected signal will be reliable.

In the training mode, a known test signal, normally a pseudo noise sequence is transmitted. Using a synchronised version of the test signal at the receiver side and comparing this signal with received signal, the resultant error signal gives the information about the channel. This error signal is used to adjust the coefficients of the equaliser.

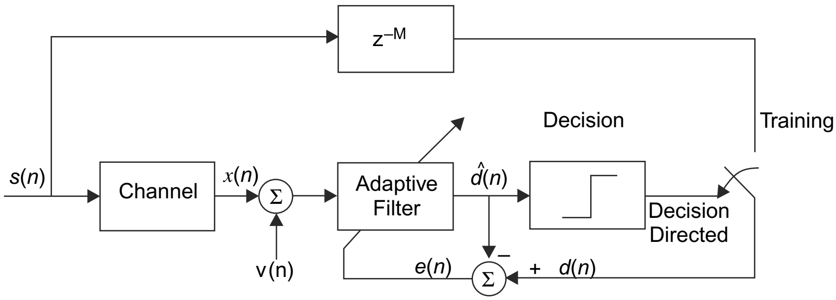


Fig. 14.7 *Adaptive Equalisation*

After this training process is done, the adaptive equaliser can be continuously adjusted in the decision-directed mode. The error signal is obtained from the final receiver estimate. The output of the adaptive equaliser is sent to the decision device receiver to obtain estimate. The estimate of the error signal is used to adjust the coefficients of the adaptive equaliser which shows slow variations in the channel and other variations introduced in the system. After the determination of appropriate coefficients of the adaptive filter, adaptive channel equalisation system switches to decision-directed mode. In this mode, adaptive channel equalisation system decodes the signal and produces a new signal, which is an estimate of the signal $s(n)$ except for a delay of M taps.

14.9.4 Adaptive Prediction Filter

In the signal prediction application, the adaptive filter input consists of a delayed version of the desired signal as illustrated in Fig. 14.8. The MSE is given by

$$E\left\{\left[x(k)-H^T x(k-L)\right]^2\right\}$$

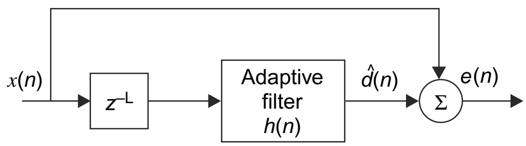


Fig. 14.8 *Signal Prediction*

The minimisation of the MSE leads to an FIR filter, whose coefficients are the elements of H. This filter is able to predict the present sample of the input signal using previous sample values $x(k-L)$, $x(k-L-1)$, ..., $x(k-L-N)$. The resulting FIR filter can then be considered as a model for the signal $x(n)$ when the MSE is small. The minimum MSE is given by,

$$\xi_{\min} = r(0) = H_0^T \begin{bmatrix} r(L) \\ r(L+1) \\ \vdots \\ r(L+N) \end{bmatrix}$$

where H_0 is the optimum predictor coefficient vector and $r(L) = E[x(k)x(k-L)]$ for a stationary process. A typical predictor's application is in linear prediction coding of speech signal, where the predictor's task is to estimate the speech parameters. These parameters H are part of the coding information that is transmitted or stored along with other information inherent to the speech characteristics, such as pitch period, among others. The adaptive signal predictor is also used for Adaptive Line Enhancement (ALE), where the input signal is a narrowband signal (predictable) added to a wideband signal. After convergence, the predictor output will be an enhanced version of the narrowband signal.

Another application of the signal predictor is the suppression of narrowband interference in a wideband signal. The input signal, in this case, has the same general characteristics of the ALE. The output signal of interest is the error signal.

14.9.5 Adaptive Line Enhancer (ALE)

The adaptive line enhancer can be used to predict a narrow-band signal that is corrupted by a wide band noise. A set-up for an the adaptive line enhancer is shown in Fig.14.9.

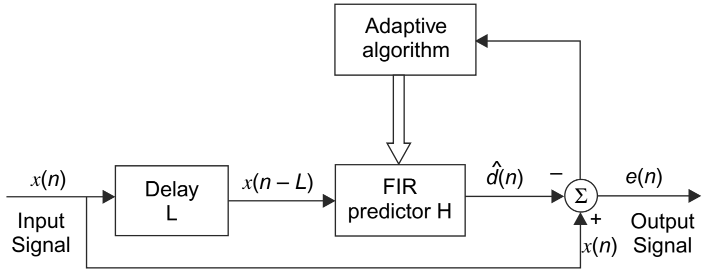


Fig. 14.9 *Adaptive Line Enhancer*

Let $x(n)$ be the input signal, passed through a delay of L sample element with a delay L, and its output is $x(n-L)$. This is passed through a predictor. The predicted output $d(n)$ is subtracted from the input signal which results in the error signal.

$$e(n) = x(n) - \hat{d}(n)$$

The error signal controls the predictor coefficients. After convergence, the predictor output will be an enhanced version of the input narrowband signal. Thus, the adaptive line enhancer is used to suppress broadband noise components and pass only the narrow band signals with less attenuation. The MSE is given by

$$E\left(e^2(n)\right) = E\left(x(n) - H^T x(n-L)\right)^2$$

Upon minimisation of MSE, the FIR adaptive predictor is able to predict the present sample of the input signal using the past "L" samples. Such an adaptive linear predictor finds application in adaptive speech coding, adaptive interference suppression etc.

14.9.6 Data Transmission over Telephone Channels

The block diagram of a full duplex data transmission over telephone channels is shown in Fig. 14.10(a).

The data from terminals A and B are transmitted over the telephone channels using modems which act as the interface between the terminals and telephone channel. The transmission is full duplex because there is simultaneous transmission in both directions.

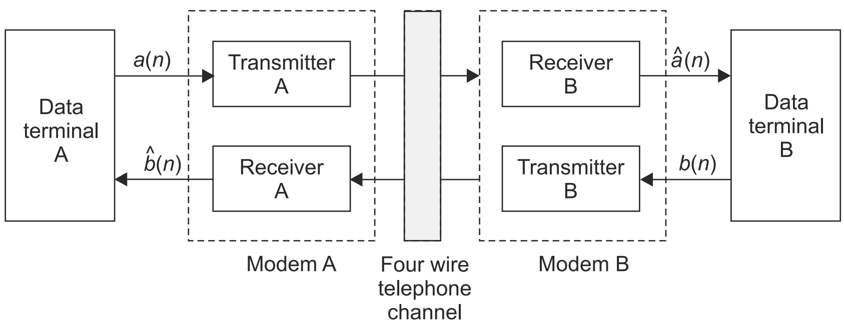


Fig. 14.10(a) *Full Duplex Data Transmission (Telephone Channels)*

The transmitted signals from A and B are given by

$$S_A(t) = \sum_{k=0}^{\infty} a(k)\, p\left(t - KT_s\right)$$

where $p(t)$ is the pulse signal shown in Fig 14.10(b).

If a subscriber gets a line from a telephone department for transmission of data between terminals A and B, a four-wire line is provided, which means two dedicated channels are available. This avoids cross-talk.

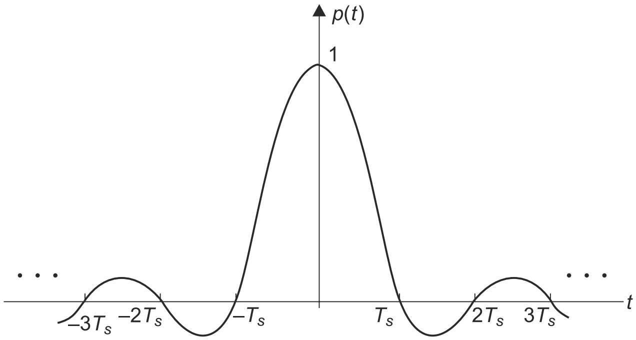


Fig. 14.10(b) *Pulse Signal*

For low volume, infrequent transmission of data this is not cost effective, instead a dial-up switched telephone network can be used. In this new network, the local link between the subscriber and the local telephone office is a two-wire line which is called the local loop. At the local telephone office, this two-wire line is linked to the main four-wire line. The main line also called as trunk line connects various telephone offices through hybrid. Isolation between the transmitter and receiver channels can be obtained by tuning the hybrid using transformer coupling. A part of the signal on the transmitter side leaks to the receiver side because of impedance mismatch between the hybrid and telephone channels.

The echo signal can be reduced by using echo suppressor/echo canceller.

An echo signal at terminal A caused by the hybrid A is a near-end and echo signal at terminal B caused by hybrid A is a far-end echo. Both these echoes must be removed by echo cancellers.

Adaptive Echo Cancellation over Telephone Channels

Echo is a phenomenon wherein there is a reflection of signal from the destination back to the listener. The echo tends to be most noticeable on long-distance calls, especially those over satellite links where transmission delays can be a significant fraction of a second. In some cases, echo occurs due to impedance mismatch due to which some power is returned back to the speaker. Figure 14.9 shows the schematic of a telephone circuit using an adaptive filter to cancel echo.The echo is caused by a leakage of the incoming voice signal from speaker A to the output line through the hybrid circuit. This leakage adds to the output signal coming from the microphone of speaker B.

Consider the four-wire and two-wire transmission in the telephone connections. At the hybrid circuit (connects a 4 to a 2-wire transmission), an echo is generated. Assume that a call is made over a long distance using satellites. There is a delay of 270 ms in the satellite communication. When A speaks to B, the speech signal takes the upper transmission path and a part of this signal is returned through the lower transmission path. The returned echo signal has the delay of 540 ms.

In Fig. 14.11, the echo cancellation is done by finding an estimate of the echo and subtracting it from the return signal. The return signal is given by

$$y(n) = \sum_{k=0}^{\infty} h(k)\, x(n-k) + v(n) \tag{14.64}$$

where $x(n)$ is the speech of speaker A

$v(n)$ is the speech of speaker B+ noise

$h(n)$ is the impulse response of the echo path

The estimate of the echo is

$$\hat{y}(n) = \sum_{k=0}^{M} \hat{h}(k)\, x(n-k) \tag{14.65}$$

where $\hat{h}(k)$ is the estimate of the impulse of the echo path.

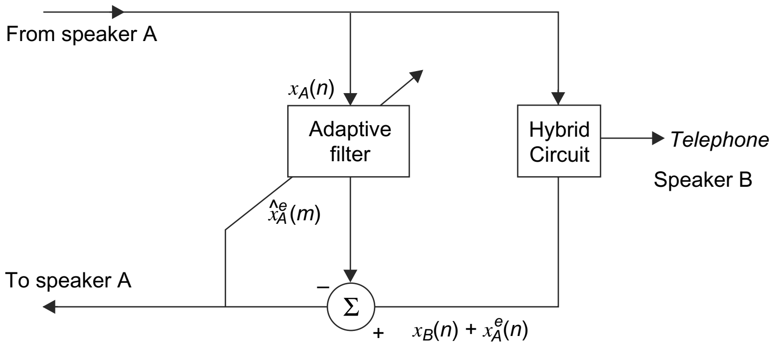


Fig. 14.11 *Block Diagram of an Adaptive Echo Cancellation System*

The echo canceller is basically an adaptive linear filter. The coefficients of the filters are adapted so that the energy of the signal on the line is minimised. Since the signals $x_A(n)$ and $x_B(n)$ are uncorrelated, the energy on the telephone line from B to A is minimised when the echo canceller output, $\hat{x}_A(n)$, is equal to the echo. $x_A^e(n)$, on the line. The echo canceller can be an infinite impulse response or finite impulse response filter. Assuming that the signal on the line from speaker B to speaker A, $y_B(n)$, is composed of the speech of speaker B, $x_B(n)$, plus the echo of speaker A, $x_A^e(n)$,

$$y_B(n) = x_B(n) + x_A^e(n)$$

Generally, speech and echo signals not simultaneously present on a phone line, simplifies the adaptation mechanism. Assuming that, the echo path is modelled with finite impulse response filter coefficients $h_k(n)$, the output estimate of the synthesised echo signal $\hat{x}_A^e(n)$ can be expressed as

$$\hat{x}_A^e(n) = \sum_{k=0}^{P} h_k(n) x_A(n-k)$$

where $\hat{x}_A^e(n)$, is an estimate of the echo of speaker A on the line from speaker B to speaker A. After subtracting the estimate of the echo, the residual echo signal $e(n)$, is given by

$$e(n) = y_B(n) - \hat{x}_A^e(n) = x_B(n) + x_A^e(n) - \sum_{k=0}^{P} h_k(n) x_A(n-k)$$

The echo is present only when speaker A is talking and speaker B is silent. Therefore,

$$e(n) = x_A^e(n) - \hat{x}_A^e(n) \quad = x_A^e(n) - \sum_{k=0}^{P} h_k(n) x_A(n-k)$$

The magnitude depends on the ability of the echo-canceller to synthesise a replica of the echo, and this in turn depends on the adaptation algorithm.The echo-canceller coefficients may be adapted using one of the variants of the least mean LMS or RLS adaptation algorithms.The time-update equation describing the adaptation of the filter coefficient vector using LMS algorithm is

$$h(n) = h(n-1) + \mu\, e(n)\, x_A(n)$$

or by using NLMS algorithm

$$h(n) = h(n-1) + \mu \frac{e(n)}{x_A^T(n) x_A(n)} x_A(n)$$

where $x_A(n) = [x_A(n) \ldots x_A(n-P)]$ and $h(n) \quad h_0(n)\ h_1(n) \ldots h_{P\ 1}(n)$ are the input signal vector and the coefficient vector of the echo canceller, and $e(n)$ is the error signal. The quantity $x_A^T(n)\, x_A(n)$ represents the energy of the input speech of the speaker A that is fed to the adaptive filter.

Types of Echo Canceller

Different configurations of echo canceller are

(i) Symbol rate echo canceller

(ii) Nyquist rate echo canceller

The difference between the two is that in the Nyquist-rate echo-cancellers, the echo canceller works at the Nyquist-rate but in the symbol-rate echo canceller, the symbol rate is considered. Figures 14.12(a) and (b) show the block diagram of symbol-rate echo canceller and Nyquist-rate echo canceller respectively.

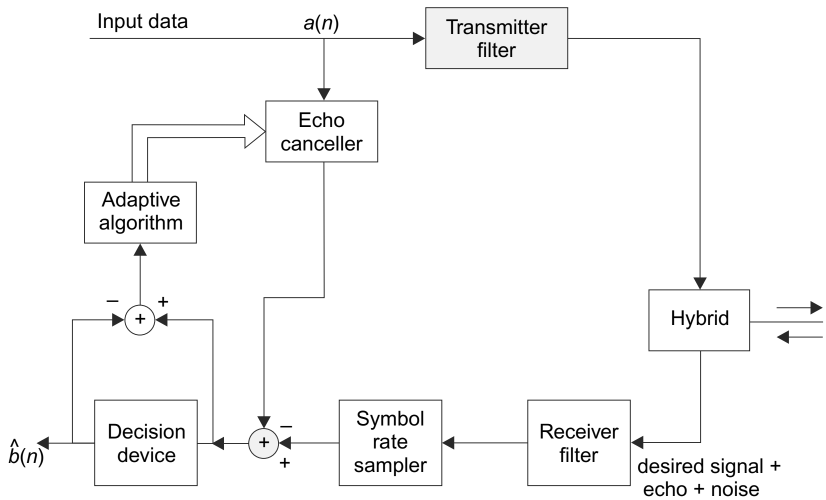


Fig. 14.12(a) *Symbol-Rate Echo Canceller*

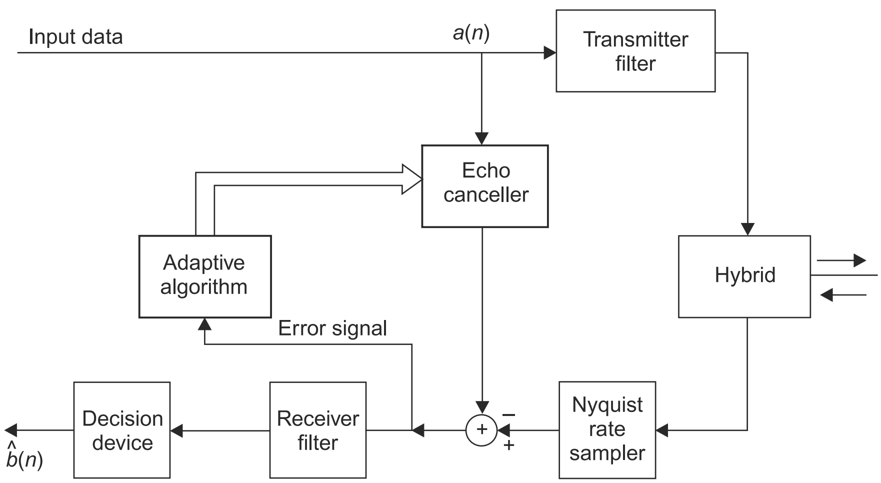


Fig. 14.12(b) *Nyquist Rate Echo Canceller*

REVIEW QUESTIONS

14.1 What are the limitations of optimum filters?
14.2 Compare optimum filters versus adaptive filters.
14.3 What are the characteristics of adaptive filters? What is the need for adaptivity?
14.4 What is the need for adaptive equalisation in a digital communication system? Explain.
14.5 Explain how a narrowband signal can be extracted when it is embedded in a broad band noise.
14.6 Explain how noise introduced in the system can be minimised.
14.7 How is the effect of echo minimised in a telephone communication?
14.8 Explain in detail the LMS algorithm for FIR adaptive filtering.
14.9 Derive the mean square error in RLS algorithm.
14.10 Discuss the steepest descent method used to adaptively estimate $d(n)$ from $x(n)$ by minimising the mean-squared error (MSE).
14.11 Define the Sign Error LMS algorithm.
14.12 Define the Recursive Least Squares (RLS) algorithm's error function.
14.13 Derive the mean square error in RLS algorithm with optimisation.
14.14 Explain how dynamic system can be modelled.
14.15 Using MMSE criterion, derive the expression for optimal filter coefficients.
14.16 Explain the forward-backward lattice algorithm.
14.17 Discuss the gradient adaptive lattice algorithm.
14.18 Give some examples of applications where adaptive filtering is done.
14.19 Consider the adaptive FIR filter shown in Fig. Q14.19. The system $H(z)$ is characterized by $S(z)=\dfrac{1}{1-0.9z^{-1}}$. Determine the optimum coefficients of the adaptive FIR filters $B(z) = b_0 + b_1z^{-1}$ which minimise the mean square error. The additive noise is white with variance $\sigma_w^2 = 0.1$.

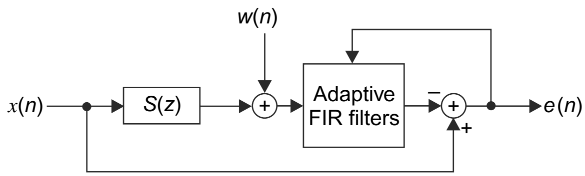


Fig. Q14.19

14.20 Sketch and label a generic adaptive filter with an input signal $x[n]$, an output $y[n]$, a desired signal $d[n]$, and an error signal $e[n]$, and describe its aim.
14.21 Consider the random process $x(n) = gv(n) + w(n)$, $n = 0, 1, \ldots, (M-1)$, where $v(n)$ is a known sequence, g is a random variable with $E(g) = 0$ and $E(g^2) = G$. The process $w(n)$ is a white noise sequence with

$$\gamma_{ww}(n) = \sigma_w^2\delta(n)$$

Determine the coefficients of the linear estimator for g, i.e.

$$\hat{g} = \sum_{n=0}^{M-1} h(n)x(n)$$

which will minimise the mean square error

$$\varepsilon = E\left[(g-\hat{g})^2\right]$$

14.22 Give some examples of applications where adaptive filtering is done.

14.23 Determine the coefficients a_1 and a_2 for the linear predictor shown in Fig. Q14.23, given that the autocorrelation $r_{xx}(m)$ of the input signal is

$$r_{xx}(m) = a^{|m|}, \quad 0 < a < 1$$

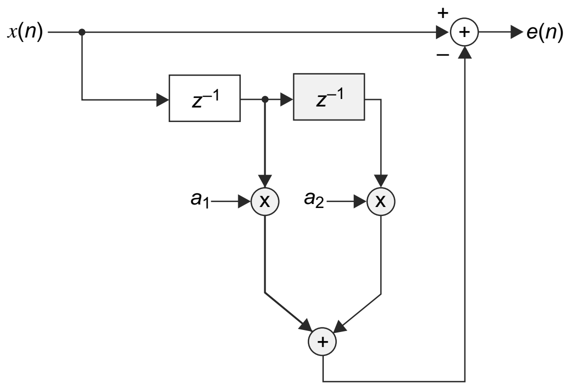


Fig. Q14.23

14.24 Explain the adaptive filter architecture shown in Fig. Q14.24 with a relevant application.

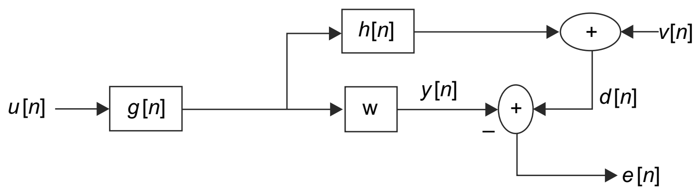


Fig. Q14.24

Chapter 15

Applications of Digital Signal Processing

INTRODUCTION 15.1

Digital signal processing techniques are used in a variety of areas which include speech processing, image processing, RADAR, SONAR, etc. These techniques are applied in spectral analysis, channel vocoders, homomorphic processing systems, speech synthesisers, linear prediction systems, analysing the signals in radar tracking, etc.

APPLICATIONS OF DSP IN BIOMEDICAL ENGINEERING 15.2

Digital signal processing techniques are widely used in the biomedical field to facilitate the monitoring, diagnosis and analysis of abnormalities in the body. Its important areas of application are (i) Removal of artifacts, (ii) Electrocardiogram (ECG), (iii) Electroencephalogram (EEG), (iv) Phonocardiogram (PCG), and (v) Speech processing.

15.2.1 Removal of Artifacts

Removal of artifacts is very important in the biomedical field, as almost all biosignals are extremely weak in nature and often buried in noise. The amplitude of biosignals is usually in the range of μV to mV, which necessitates sophisticated instrumentation and data acquisition techniques to acquire these signals from the body. The presence of different types of noises further distort the signals, and make the data acquisition process even more cumbersome. Thus, it is evident that the most important task in the biomedical applications is the careful removal of noises from the original biosignal, using appropriate digital signal processing techniques. The types of noises include random noises, power line and physiological interferences, motion artifacts and improper placement of electrodes. The usage of ac lines which have the power line frequency of 50 Hz in India or 60 Hz abroad affects the biosignals, as this power line frequency falls within the bandwidth of most of the biosignals. Power line can also be named as structured noise, because the spectral content of this noise is known. But noises raised from physiological interferences may not have characteristic or structured waveforms. The examples of physiological interferences are:

(i) While recording ECG, the EMG signals related to coughing, sneezing or heavy breathing may also interfere.
(ii) Fetal ECG signals of interest getting affected by maternal ECG signals.
(iii) Interference of continuing EEG activity in the study of evoked response potentials.
(iv) While recording PCG (heart sounds), sounds related to heavy breathing may also get mixed with.

Choices of Filters

Appropriate selection of a filter is essential as the biosignal is embedded completely in noises. As the signal of interest provides vital diagnostic information, it has to be extracted very carefully. In order to satisfy these requirements digital filters are widely used. The choice of a particular digital filter depends on the type of the biosignals. The types of filters include,

(a) Time domain filters
  (i) Synchronized filtering
  (ii) Moving Averaging filtering (FIR)
  (iii) Derivative operators
(b) Frequency domain filters (IIR)
(c) Optimal filters (Wiener)
(d) Adaptive filters

(a) *Time Domain Filters* Linear filters work better when the signals and noises are readily distinguishable, but fail to perform when they overlap. Random noises have no distinguishable spectral pattern and they readily overlap with the spectrum of signals. These noises are better dealt with the time domain signal processing techniques such as synchronised averaging and moving averaging filters. In this method, a particular part or repetitive event within a signal waveform serves as reference point. For example, to filter ECG waveform, QRS complex is the readily noticeable sharpest repetitive peak and it can be taken as a reference point. Various parts of the signal are extracted, aligned or in other words synchronised together as per the reference point and averaged by the number of events. When it is not possible to have multiple realisation of an event in a particular biosignal, the moving averaging filters such as Von Hann or Hanning filters are used instead of synchronised filters. The patient movement and improper contact of electrodes cause base line drift problem in the ECG waves. The position of base line is important for measuring the ST segments, which give vital information about heart blocks. To avoid base line drift, derivative based operators can be used. Whenever there is a fast change in the base line, it can be detected by the derivative operator and it can be duly compensated.

(b) *Frequency Domain Filters (IIR)* Frequency domain filters are used because of their easiest implementation, and to provide specific low-pass, high-pass, band-pass or band reject characteristics. Power line interference noise not only contains the fundamental frequency of 50 Hz, but it also includes its harmonics. Noise due to power line interference can be removed by using a notch filter, and its harmonics can be removed by using comb filters.

Electromyogram (EMG) signals are relatively high frequency signals compared to ECG signals, and their influence in ECG recording can be eliminated by using the Butterworth low-pass filters and band-pass filters. As the ECG signal frequency lies below 100 Hz, any frequencies above 100 Hz may be attributed to EMG signals. Thus a Butterworth low-pass filter or band-pass filter of cut-off frequencies within 100 Hz minimises the influence of the EMG signals.

(c) *Optimal Filters (Wiener)* Wiener filters are used whenever the statistical characteristics of the biosignal and noise process are known. The output of wiener filter is much useful and optimised, over conventional filters. As the models of the noise and signal PSDs are used for deriving the wiener filters, there is no need of specifying cutoff frequencies as done in the frequency domain filters.

(d) *Adaptive filters* Most of the physiological interferences are non stationary signals. And because of its non stationary nature the spectral contents of these types of noises are not well defined. Adaptive filters are used when the spectral contents of the noises and signals overlap significantly. They are extensively used in the extraction of desired fetal ECG from the maternal ECG. This procedure is explained in the sub-section 15.2.2.

15.2.2 ECG Applications

Extraction of Fetal ECG by Adaptive Cancellation of the Maternal ECG

The problem of obtaining the fetal ECG from the mother's abdomen is that, the maternal ECG is also being extracted and added along with the desired fetal ECG. In the solution described by Widrow et al. while the combined ECG of mother fetus is obtained from the single abdomen lead, the maternal ECG is also obtained separately using multiple (four) chest electrodes. As the maternal cardiac vector is projected in different ways on the individual electrodes, the characteristics of the maternal ECG from the abdominal leads would be different from that obtained from chest leads. Maternal ECG obtained from chest leads are given to adaptive filters as a reference and the maternal ECG from the combined ECG is filtered out to extract the fetal ECG.

Detection of 'QRS' Complex

In the ECG diagnosis, QRS complex is with an important step, as it reveals the information about the function of ventricular pumping and the R-R interval which is normally used for counting heart rate. As the QRS complex has the steepest slope in the cardiac cycle, this fast rate of change can be detected by a derivative based method along with a suitable filter combination. The most popular DSP method for QRS detection is the Pan-Tompkins algorithm which uses a series of filters, derivation, squaring, and integration followed by adaptive threshold procedures.

Detection of 'P' and 'T' Waves

Detection of 'P' and 'T' waves is a difficult and challenging procedure as these signals are weak and have no constant shapes. The procedure includes the elimination of QRS, 'T' peak detection, rectification, thresholding and cross correlation to output 'P' wave.

15.2.3 EEG Applications

The task of event and spike detection in EEG is tedious, as these signals are usually acquired simultaneously from multiple channels. DSP techniques such as autocorrelation, crosscorrelation and matched filters are used for this purpose.

Detection of EEG Rhythms

The EEG signal may contain α, β, δ and θ waves which can be detected or extracted separately. These waves provide information about the brain activities of a person. The auto correlation function can be used to detect the EEG rhythms, as it can display peaks at intervals corresponding to the period of any repetitive event present in a signal. For example, the presence of a peak around 0.1s indicates the presence of α rhythms. The cross correlation function can be used to extract common rhythms between two EEG channels or from separate time events.

Detection of EEG Spikes

The abrupt spikes present in EEG provide valuable information about the epilepsy conditions. The matched filters are used for the detection of spikes in the EEG signals. This filter performs correlation between the input EEG signal and a reference spike and wave template, and provides the peaks at the time of incidence of spikes.

15.2.4 PCG Applications

Identifications of Heart Sounds S1 and S2

The QRS complex is first detected using Pan-Tompkins algorithms, and one period of the PCG is identified within the interval between two successive QRS complexes. Then, Lenher and Rangayyan method is used for detecting dicrotic notch. The onset of S2 is obtained from the knowledge about the location of dicrotic notch.

Segmentation of PCG into its Systolic and Diastolic Parts

The S1 and S2 sounds are obtained using the method mentioned earlier. The detected S1-S2 interval provides the information about the systolic part of the PCG cycle and the interval between S2 and next onset of S1 provides the information about the diastolic part of the PCG cycle. Using this information, the PCG can be segmented into its systolic and diastolic parts.

15.2.5 Speech Processing

Extraction of Vocal-tract Response from a Voiced-speech Signal

The voiced speech signals can be represented in discrete time as convolution of glottal waveform and vocal tract response. The vocal tract filter may be assumed to be a linear, shift invariant filter. A homomorphic filter is used for convolved signals. The output of the homomorpic filter is used to extract the basic terms corresponding to the vocal-tract response, from a voiced speech signal.

VOICE PROCESSING 15.3

There are different areas in voice processing like encoding, synthesis and recognition. In a speech signal some amount of redundancy is present, which can be removed by encoding. Synthesis is required at the receiver side because in the transmitter, compression/coding is done. Recognition involves recognising both the speech and the speaker.

15.3.1 Speech Signal

A speech signal consists of periodic sounds, interspersed with bursts of wide band noise and sometimes short silences. The vocal organs are in motion continuously, hence the signal generated is not stationary. But short segments of 50 ms are treated to be approximately stationary.

These signals are generated by the vibration of the vocal cords. The muscles in the larynx stretch these cords, which vibrate when air is forced, thus producing sound in the form of a pulse train. This passes through the pharynx cavity and tongue and is expelled either at the mouth or nasal cavity depending on the position of velum. Figure 15.1 shows the mechanism of human speech production.

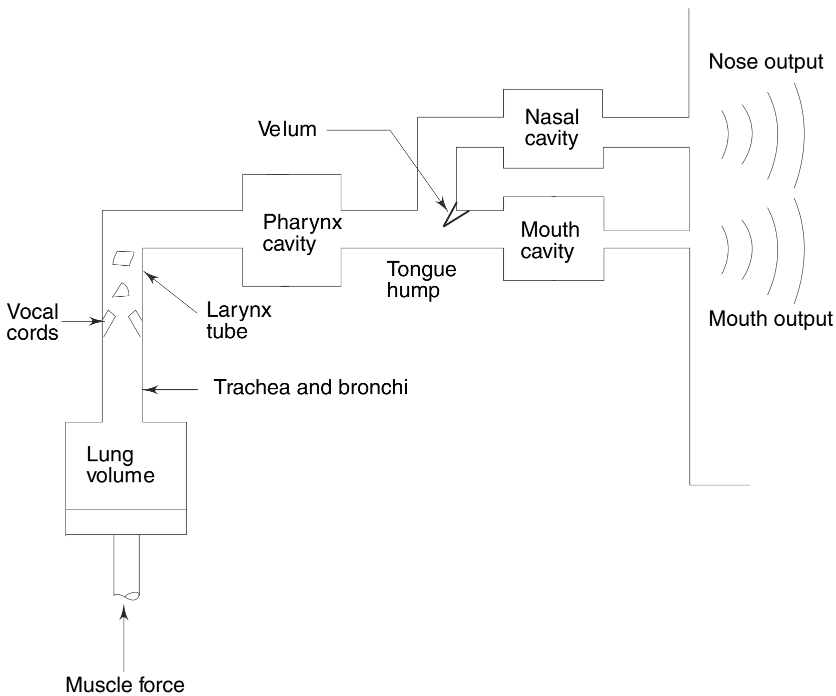


Fig. 15.1 *Mechanism of Human Speech Production*

When air passes through narrow constrictions in the vocal tract, a turbulent flow is produced. Otherwise pressure is built up along the tract behind a point of total constriction.

The vocal system can be modelled with a periodic signal excitor, a variable filter representing the vocal tract, switch to pass the signal either through nasal cavity or mouth and a wide band noise source to represent fricatives (more or less continuous turbulence). Figure 15.2 shows models of the vocal organs.

The human vocal tract can be considered as a non-uniform acoustical tube which extends from the glottis to the lips. In an adult male, the vocal tract is 17 cm long. The first quarter wave resonance occurs at

$f_1 = \frac{1}{4}\frac{c}{l}$ Hz, where c is the speed of sound, i.e. 350 m/sec and l is the average length of the vocal track,

$$= \frac{1}{4}\frac{350}{17\times 10^{-2}} \approx 500 \text{ Hz}$$

The normal resonant modes of vibration of the vocal tract are called ***formants***.

In many speech processing systems, the source of excitation and the vocal tract system are assumed to be independent. A simple speech prediction model is shown in Fig. 15.2(b).

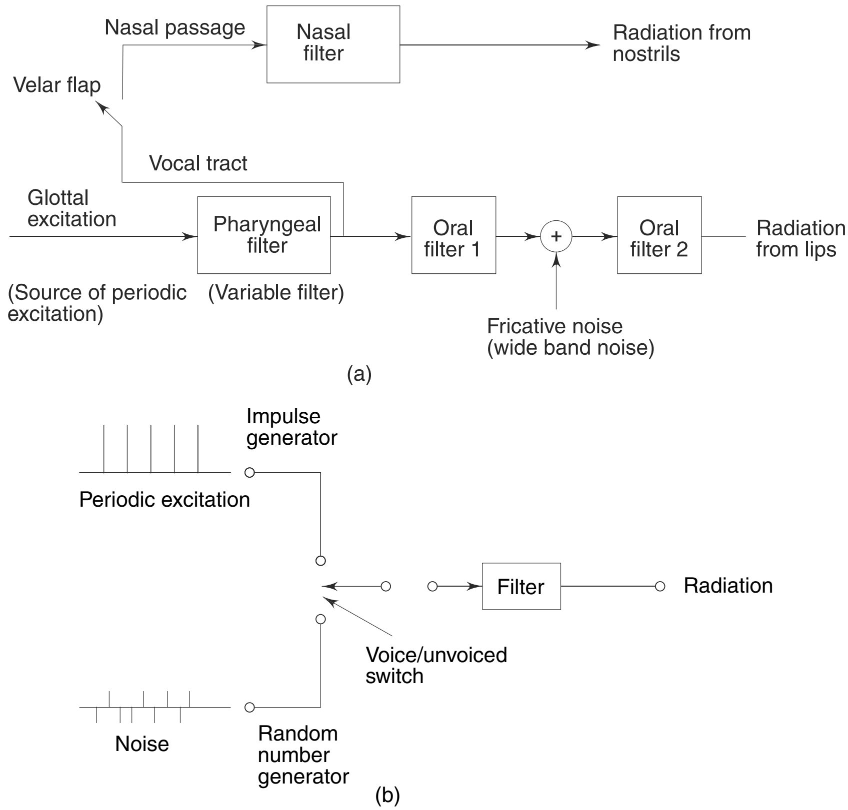


Fig. 15.2 *Models of Vocal Organs*

The excitation sources are an impulse generator and a random number noise generator. The rate at which the impulse generator generates pulse is equal to the rate of oscillation of the vocal cords. The filter represents the vocal tract system. For representing quasi-random turbulence and pressure build-up, a random number generator is used. The sources are given as input to a linear, time varying digital filter. The filter represents the vocal tract system whose coefficients are varied in short intervals of time.

15.3.2 Digital Representation of Speech Signals

Various digital methods are used for efficient encoding and transmission of speech signals. Digital methods are used for the following reasons:

(i) Degrading effects due to noise and interference can be controlled

(ii) With VLSI technology, cost and size of the hardware are reduced.

The various waveform coding methods for speech encoding are pulse code modulation (PCM), differential PCM (DPCM), delta modulation (DM), adaptive DPCM, etc.

The main objective of encoders is to represent the speech signal with minimum number of bits without degrading the information carried by the speech signals. Low bit rate speech encoders are available which models the speech source. Adaptive filters with linear predictive coding are used in these systems.

The vocal tract filter can be modelled as a linear all-pole filter, the system being defined by

$$H(z)=\frac{G}{1+\sum_{k=1}^{p} a_k z^{-k}}$$

where p — number of poles
G — filter gain
$\{a_k\}$ — filter coefficients

Modelling Voiced and Unvoiced Speech Sounds

(i) *Voiced Speech Sound* Consider the speech signal for a short duration. It is periodic with frequency f_0 and the pitch period is $\frac{1}{f_0}$. Thus the voiced speech sound can be obtained by exciting the all-pole filter by a periodic impulse train with period $\frac{1}{f_0}$.

(ii) *Unvoiced Speech Sound* These are obtained by using a random noise generator to excite the all pole filter.

Figure 15.3(a) shows the model for the generation of speech signals. The spectrum encoder has to determine the following:

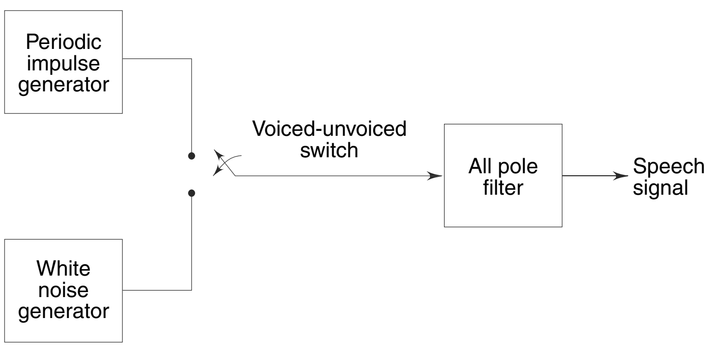


Fig. 15.3(a) *Model for Speech Signal Generation*

(i) Proper excitation function
(ii) Pitch period for voiced signals
(iii) Gain, G
(iv) Filter coefficients, left a_k.

Figure 15.3(b) shows the speech encoder. From the data, the model parameters are adaptively determined. Speech synthesis is done using the model and the error signal is obtained as the difference between the actual and synthesised signals. Alongwith the model parameters, the error signal is also encoded and transmitted. The speech signal can be retrieved from the model parameters and the error signal.

Linear prediction is used for obtaining the model and error signal. Figure 15.3(c) shows the pole parameters of the estimation model.

The output of the FIR filter is

$$\hat{x}(n) = -\sum_{k=1}^{p} a_k x(n-k)$$

and the error signal is

$$e(n) = x(n) - \hat{x}(n)$$

$$e(n) = x(n) + \sum_{k=1}^{p} a_k x(n-k)$$

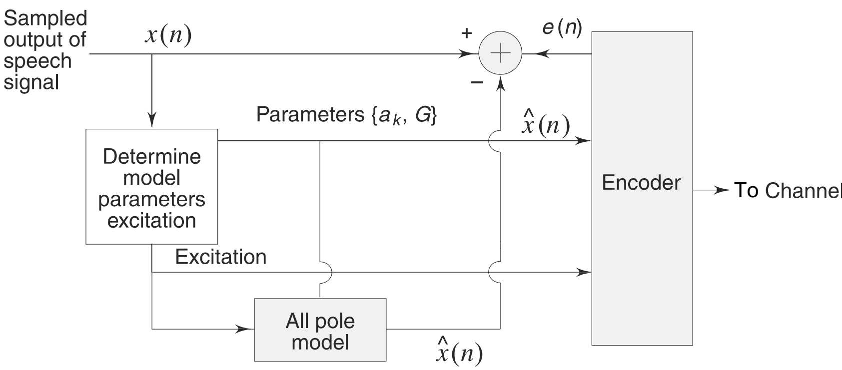


Fig. 15.3(b) *Speech Encoder*

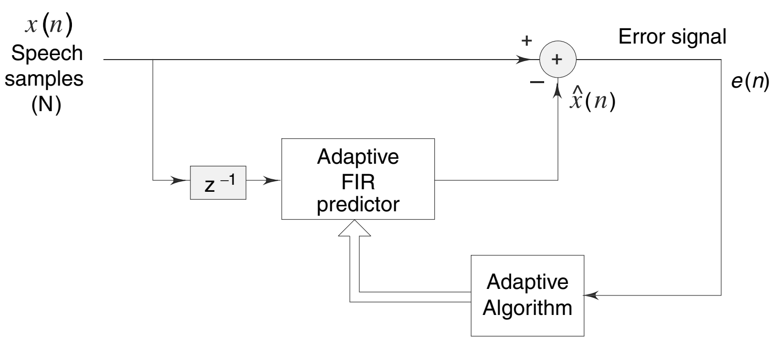


Fig. 15.3(c) *Pole Parameter Estimation in LPC*

If the least squared criteria is applied, the optimised result gives,

$$\sum_{k=1}^{p} a_k r_{xx}(l-k) \approx -r_{xx}(l), \quad l = 1, 2, ..., p$$

where $r_{xx}(l)$ is the autocorrelation of $x(n)$

$$x(n) = -\sum_{k=1}^{p} a_k x(n-k) + Gv(n)$$

where $v(n)$ is the input sequence

$$Gv(n) = x(n) + \sum_{k=1}^{p} a_k x(n-k)$$

$$G^2 \sum_{n=0}^{N-1} v^2(n) = \sum_{n=0}^{N-1} e^2(n)$$

Consider the excitation to be normalised.

Then, $G^2 = \sum_{m=0}^{N-1} e^2(n) = r_{xx}(0) + \sum_{k=1}^{p} a_k r_{xx}(k)$

From the least squared optimisation, the residual energy arrived at is used to set the values of G^2.

15.3.3 Analysis of Speech Signals

Frequency domain analysis can be done by using FFT. It requires a data segment with suitable length such that the harmonics can be resolved and the data within that length should be approximately stationary. Windowing can be done to extract the required length of the data segment. The pitch of the human voice ranges from a minimum of 80 Hz (males) to a maximum of 400 Hz (females). If three periods of 80 Hz are considered, then the selected data length is for 37.5 ms.

15.3.4 Short Time Spectrum Analysis

Consider the discrete time signal, *x(nT)*, $-\infty < n < \infty$. The Fourier transform of *x(nT)* is

$$\mathcal{F}\{x(nT)\} = \sum_{n=-\infty}^{\infty} x(nT)\, e^{-j\omega nT} \tag{15.1}$$

For a time varying signal, like the speech signal, the measure of energy content of the speech waveform is

$$X(\omega, nT) = \sum_{r=-\infty}^{n} x(rT) h(nT - rT)\, e^{-j\omega rT} \tag{15.2}$$

This represents the infinite time Fourier transform at time *nT* using the windowing technique.

$$X(\omega, nT) = [x(nT)e^{-j\omega nT}]^* h(nT) \tag{15.3}$$

Let $$X(\omega, nT) = a(\omega, nT) - jb(\omega, nT) \tag{15.4}$$

where $$a(\omega, nT) = \sum_{r=-\infty}^{\infty} x(rT)h(nT - rT)\cos(\omega rT)$$

$$b(\omega, nT) = \sum_{r=-\infty}^{\infty} x(rT)h(nT - rT)\sin(\omega rT) \tag{15.5}$$

The window is used to represent an ideal low-pass filter with cut-off frequency ω_c. Figure 15.4 represents the technique for STFT. The speech energy spreads over the band $(\omega - \omega_c)$ to $(\omega + \omega_c)$. The frequency response of an ideal low-pass filter is shown in Fig.15.5.

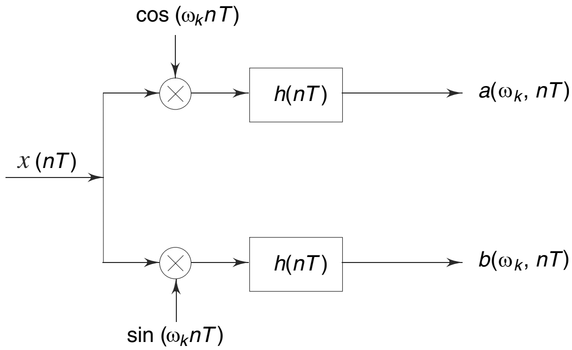


Fig. 15.4 *Technique for Short Time Spectral Analysis for Speech*

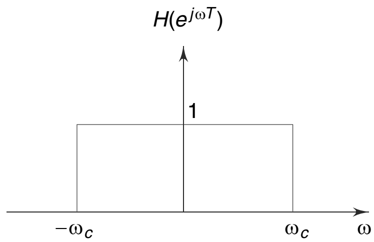


Fig. 15.5 *Ideal Low-pass Filter for Short Time Spectral Analysis*

The inverse short time transform at a finite set of N frequencies, over the band $0 \le \omega T \le 2\pi$ gives $h(nT)$ with the impulse response of an FIR filter. It is non-zero for $0 \le n \le M-1$ and the centre frequencies are

$$\omega_{k=} \frac{2\pi k}{NT}, \quad k = 0, 1, ..., N-1 \tag{15.6}$$

Therefore, $X(\omega_k, nT) = \sum_{r=n-M+1}^{n} x(rT)h(nT - rT)e^{-j\omega_k rT}$

$$= \sum_{m=0}^{\left[\frac{M}{N}\right]+1} \sum_{r=n-(m-1)N+1}^{n-mN} x(rT)h(nT - rT)e^{-j\omega_k rT} \tag{15.7}$$

where $[(M/N) + 1]$ is the greatest integer less than or equal to M/N.

Let $l = n - mN - r$

$$X(\omega_k, n\text{T}) = \sum_{m=0}^{\left[\frac{M}{N}\right]+1} \sum_{l=0}^{N-1} x(nT - rT - mNT)h(lT - mNT)e^{-j\omega_k(l-n+mN)T}$$

$$= e^{-j\left(\frac{2\pi}{N}\right)kn} \sum_{l=0}^{N-1} \sum_{m=0}^{\left[\frac{M}{N}\right]+1} \begin{Bmatrix} x(nT - lT - mNT)x \\ h(lt + mNT)e^{j\left(\frac{2\pi}{N}\right)kl} \end{Bmatrix} \tag{15.8}$$

$$X(\omega_k, nT) = e^{-j\left(\frac{2\pi}{N}\right)kn} \sum_{l=0}^{N-1} g(l, n)e^{j\left(\frac{2\pi}{N}\right)lk} \tag{15.9}$$

where $$g(l,n)=\sum_{m=0}^{\left[\frac{M}{N}\right]+1} x(nT-lT-mNT)h(lT+mNT) \tag{15.10}$$

This analysis is done usually with a bank of digital filters.

15.3.5 Speech Analysis Synthesis System

The main objective is to measure the outputs of bandpass filter banks and reconstruct the speech from these signals. Figure 15.6(a) shows the analysis-synthesis system. Let $x(nT)$ be the speech input and $y(nT)$ be the reconstructed synthetic waveform. The impulse response of the bandpass filter bank is $h_k(nT)$, $k = 1, 2,..., M$. $y(nT)$ is obtained by summing the individual bandpass filter outputs

$$y_k(nT),\ k = 1, 2,..., M$$

$$h_k(nT) = h(nT)\cos(\omega_k nT) \tag{15.11}$$

where $h(nT)$ is the impulse response of a lowpass filter.

Then $y_k(nT)$ will be

$$y_k(nT)=\sum_{r=-\infty}^{\infty} x(rT)h(nT-rT)\cos(\omega_k(nT-rT)) \tag{15.12}$$

$$= \text{Re}[e^{j\omega_k nT} X(\omega_k, nT)]$$

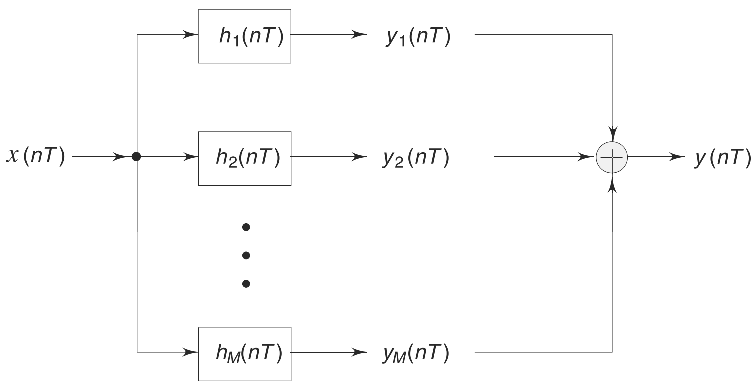


Fig. 15.6 *(a) Analysis Synthesis System for Short Time Spectral Analysis*

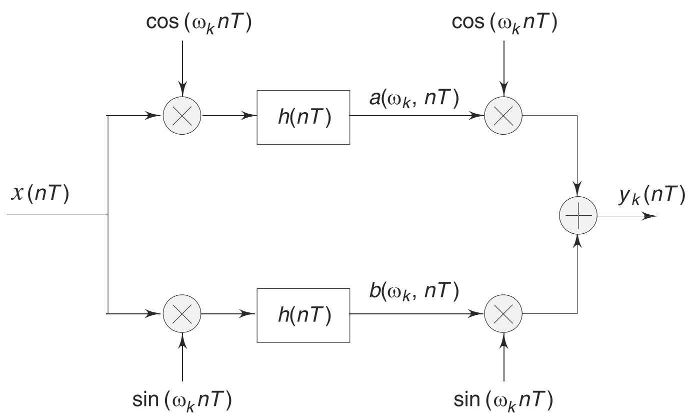


Fig. 15.6 *(b) Realisation of $y_k(nT)$*

where $$X(\omega_k, nT) = \sum_{r=-\infty}^{n} x(rT)h(nT - rT)e^{-j\omega_k rT} \tag{15.13}$$

Representing $X(\omega_k, nT)$ in terms of real and imaginary components,

$$y_k(nT) = a(\omega_k, nT)\cos(\omega_k nT) + jb(\omega_k, nT)\sin(\omega_k nT) \tag{15.14}$$

The realisation of $y_k(nT)$ is shown in Fig.15.6(b).

15.3.6 Compression and Coding

A voice signal has the frequency range of 300 to 3000 Hz. It is sampled at a rate of 8 kHz and the word length of the digitised signal is 12 bits.

The redundancy present in the voice signals can be reduced by signal compression and coding.

The different voice compression and coding techniques are

(i) Waveform coding – non-uniform, differential, adaptive quantisation.
(ii) Transform coding – transform the voice signal to an orthogonal system and then coding the transform.
(iii) Frequency band encoding – frequency range of voice signals are divided into discrete channels and then each channel is coded separately.
(iv) Parametric methods – linear prediction.

15.3.7 Channel Vocoders

The channel vocoder (voice coder) is an analysis synthesis system. Figure 15.7 shows the channel vocoder. A filter bank is used to separate the bands. There are about 8 to 10 filters. Using level detectors

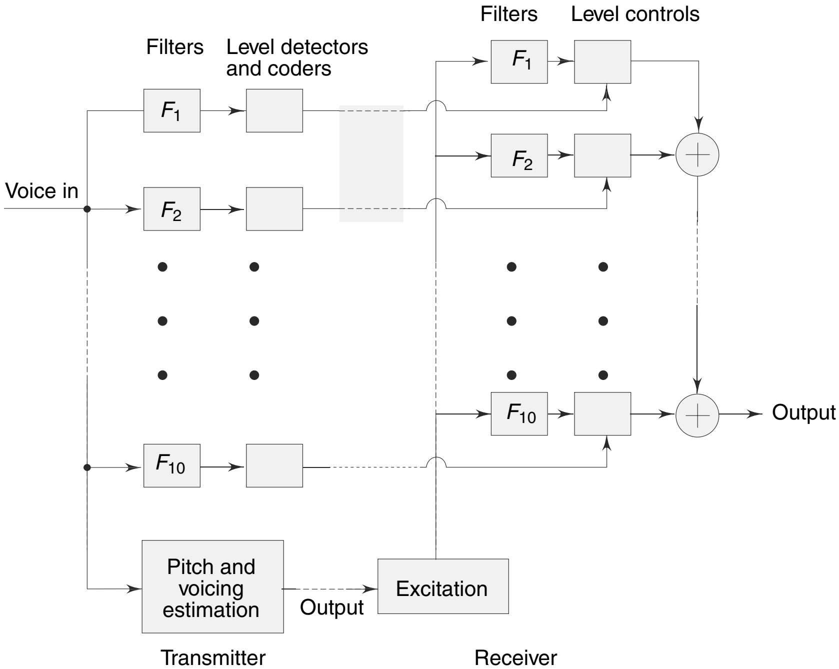


Fig. 15.7 *Channel Vocoder*

and coders, the amplitude of the filter outputs are encoded. Pitch and voicing information are also sent along with them. A wide band excitation signal is generated at the receiving end using the transmitted pitch and voicing information. For an unvoiced signal, the excitation consists of a periodic signal with appropriate frequency. For a voiced signal, the excitation is a white noise. At the receiver side, a matching filter bank is available, so that the output level matches the encoded value. The individual outputs are combined to produce the speech signal.

15.3.8 Sub-band Coding

In digital signal transmission, the analog signal is converted into a digital signal and after processing, it may be transmitted over a limited bandwith channel or the digital signal can be stored. If we employ signal compression, then the efficiency of transmission or storage can be improved.

Consider the telephone communication system. The bandwidth of the audio (speech) signal is around 4 kHz. Let this signal be sampled at the rate of 8 kHz and then passed through an 8-bit A/D converter. The output of the A/D converter is a 64 kbps digital signal. Suppose, this signal is to be transmitted over the T-1 carrier channel, a maximum of 24 such signals can be transmitted by time division multiplexing (TDM). If the audio signal is compressed to 32 kbps, then the number of signals that can be transmitted over the same channel can be doubled, i.e. 48 signals can be transmitted.

If the audio signal spectrum is analysed, then the non-uniform distribution of signal energy in the frequency band can be noted. Taking advantage of this fact, signal compression can be achieved using sub-band coding. Figure 15.8 shows the sub-band coding process.

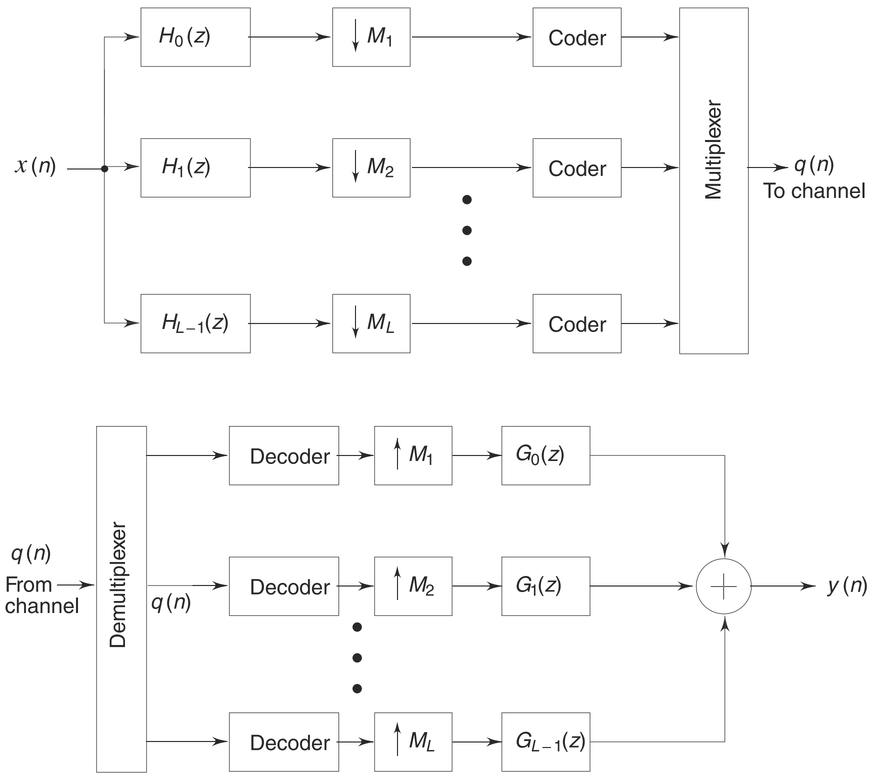
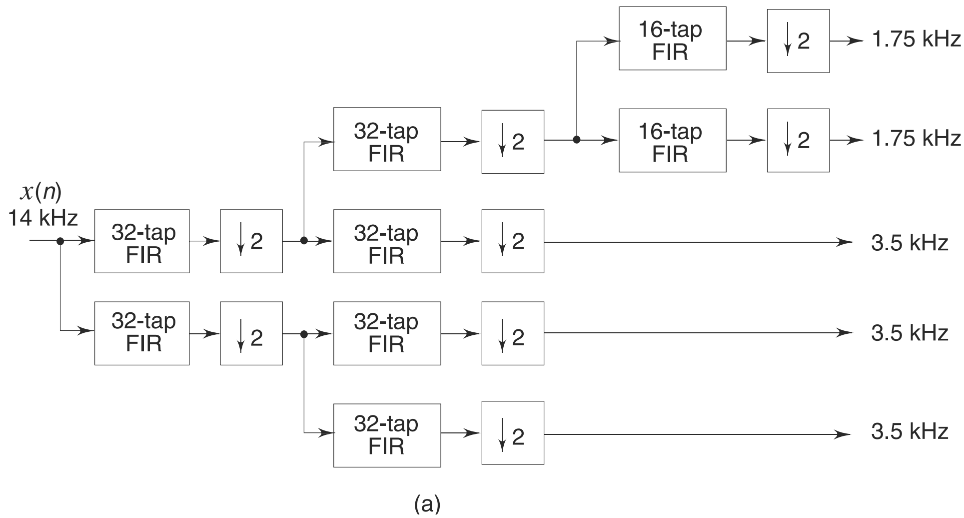


Fig. 15.8 *Sub-band Coding Process*

The signal *x(n)* is split into many narrowband signals which occupy contiguous frequency bands using analysis filter bank. These signals are down-sampled, yielding sub-band signals, which are then compressed using encoders. The compressed signals are multiplexed and transmitted. On the receiving side, reverse operations are carried out. The received signal is demultiplexed, decoded, up-sampled and then passed through a synthesis filter bank. The outputs of the synthesis filter bank are combined to yield the signal *y(n)*. By using sub-band coding, better compression ratio is achieved, because each sub-band signal can be represented using a different number of bits. This is possible because of the signal's spectral characteristics.



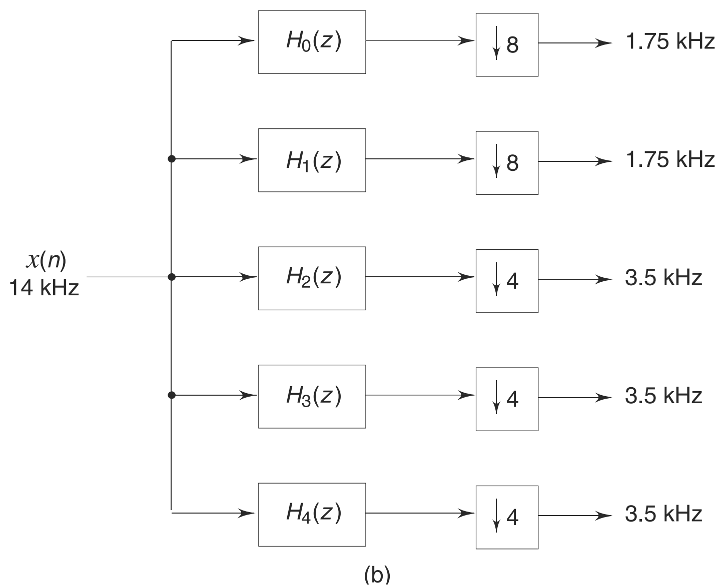


Fig. 15.9 *(a) Three Stage Realisation of the Five-band Analysis Bank and (b) Its Equivalent Representation*

A 7 kHz wide band audio signal is coded using sub-band coding based on a five band QMF bank (based on the investigation results of Richards and Jayant) as shown in Table 15.1. The frequency ranges are 0–875Hz, 875–1750 Hz, 1750–3500 Hz, 3500–5250 Hz and 5250–7000 Hz. The realisation of the structure is shown in Fig. 15.9. Adaptive differential PGM is used for encoding the sub-band signals.

Table 15.1

Frequency Channels (in Hz)	*Bit allocation*
0 – 875	5 bits/sample
875 – 1750	5 bits/sample
1750 – 3500	4 bits/sample
3500 – 5250	4 bits/sample
5250 – 7000	3 bits/sample

Hence the total bit rate of the encoder is 56 kbps.

15.3.9 Voice Privacy

In telephone communications, privacy is required for protection against eavesdropping. Let us consider a method which is based on a simultaneous sequential time and frequency segment permutation (TFSP) scheme.

The long speech time segment is divided into B contiguous blocks. Then each time segment block is split into F contiguous frequencies. The total number of time-frequency segments available is FB as shown in Fig. 15.10.

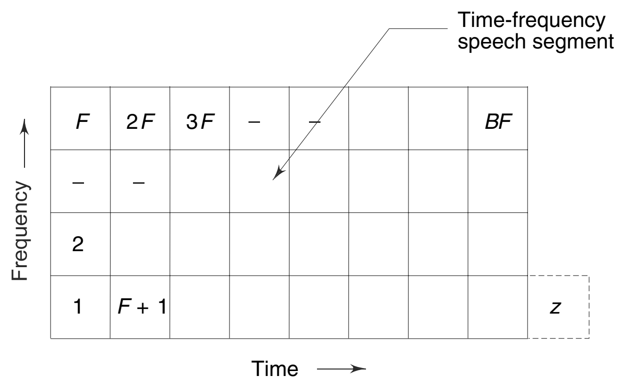


Fig. 15.10 *Time-Frequency Speech Segment Matrix*

This information is stored in a scrambler memory. Using a random number generator, the segment in the memory for transmission is determined. Figure 15.11 shows the full duplex analog voice privacy system.

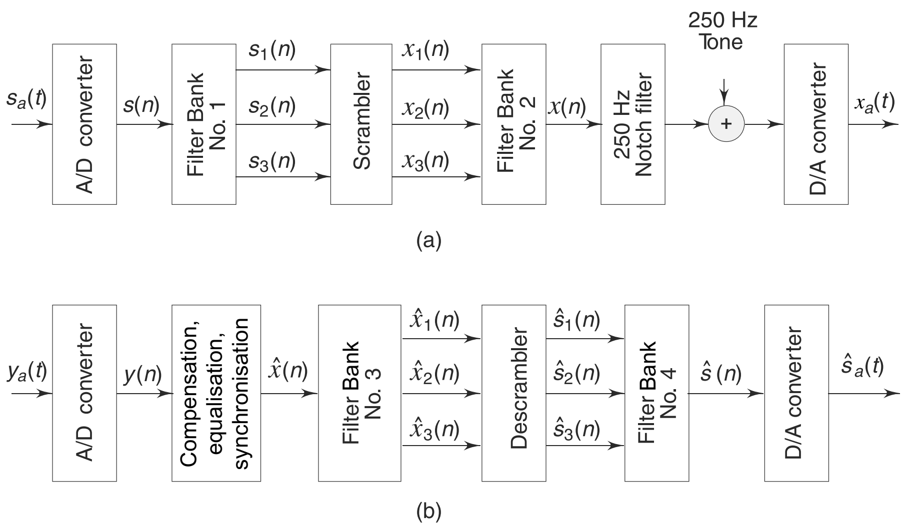
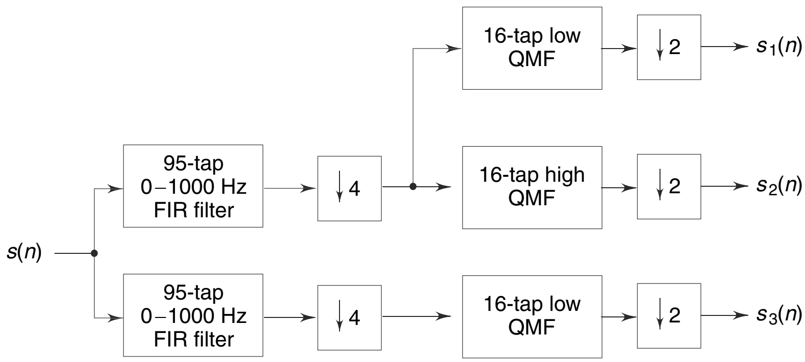
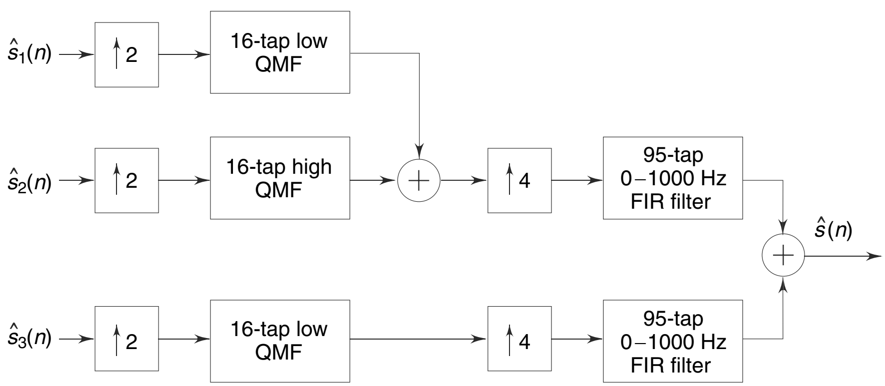


Fig. 15.11 *Full Duplex Analog Voice Privacy System (a) Transmitter and (b) Receiver*

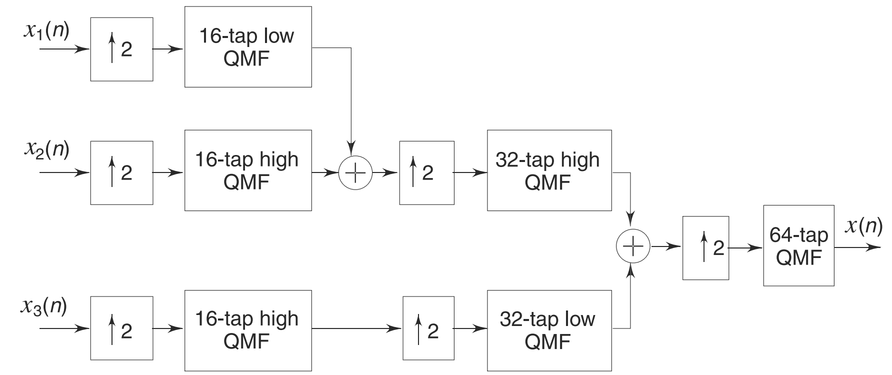
The analog voice signal $s_a(t)$ is passed through an anti-aliasing low-pass filter and then sampled to obtain a digital signal at 8 kHz sampling rate. The digital signal $s(n)$ then passes through a filter bank I (3-channel analysis filter bank) where it separates into sub-bands with frequencies 0−500 Hz, 500–1000 Hz and 2000–2500 Hz. They are downsampled by a factor of eight, and blocked into segments (segment size−5 ms) resulting in $s_1(n)$, $s_2(n)$ and $s_3(n)$. With a pseudorandom key, these signals are scrambled to yield $x_1(n)$, $x_2(n)$ and $x_3(n)$. These signals are combined through filter bank-II (3-channel synthesis filter) into either a 500−2000 Hz band or a 2000−3500 Hz band after upsampling. For frequency shift compression, a 250 Hz tone signal is continuously transmitted. The combined signal is converted to an analog signal, $x_A(t)$ when passed through a D/A converter which operates at 8 kHz. At the receiver end the received signal $y_a(t)$ $[x_a(t)$ + channel noise] is converted to a digital signal $y(n)$ by a D/A converter which also operates at 8 kHz. To get the replica of $x(n)$, the signal $y(n)$ is passed through an equaliser, synchroniser and compensation block. These signals $\hat{x}_1(n)$, $\hat{x}_2(n)$ and $\hat{x}_3(n)$ are passed through filter bank III (analysis filter bank) which provides perfect reconstruction with filter bank II. These signals are descrambled with filter bank IV (synthesis filter bank). $\hat{s}_1(n)$ close to $s(n)$ is obtained when passed through a D/A converter and the analog signal is obtained (Refer to Fig. 15.12(a) to (d)).



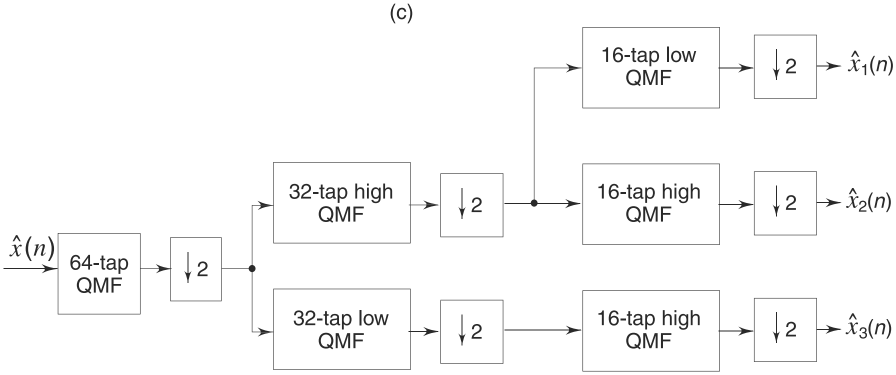
(a)



(b)



(c)



(d)

Fig. 15.12 *(a) Filter Bank, (b) Filter Bank IV, (c) Filter Bank II and (d) Filter Bank III*

Digital FM Stereo

A simple block diagram of a digital FM stereo generator is shown in Fig. 15.13.

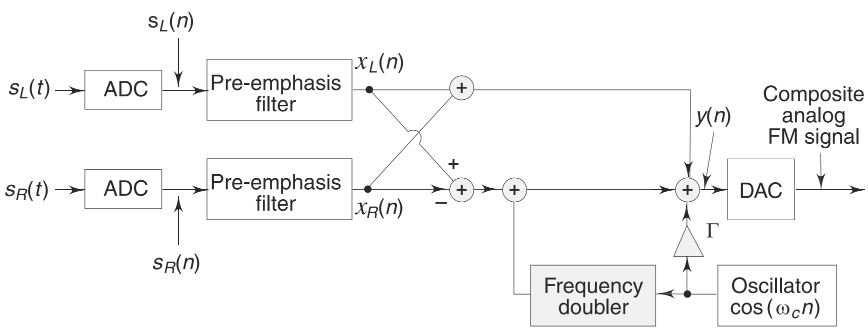


Fig. 15.13 *FM Stereo Transmitter*

The analog output of the left and right microphones are $s_L(t)$ and $s_R(t)$. These are converted to digital signals $s_L(n)$ and $s_R(n)$ by A/D converters. The low frequency components have high amplitudes while the high frequency components have low amplitudes. To improve the SNR, pre-emphasis filters are included. The following signals are transmitted:

- Sum of pre-emphasised signal $x_L(n)$ and $x_R(n)$ in baseband form
- Difference signal $x_L(n) - x_R(n)$, which is modulated using DSB-SC technique
- Pilot carrier

The combined signal $y(n)$ is

$$y(n) = [x_L(n) + x_R(n)] + [x_L(n) - x_R(n)]\cos(2\omega_c n) + \Gamma\cos(\omega_c n)$$

where

$$\omega_c = 2\pi\frac{F_C}{F_T}$$

F_C – Pilot carrier frequency (19 kHz)

F_T – Sampling frequency (≈ 32 kHz)

Γ – Gain constant for pilot signal

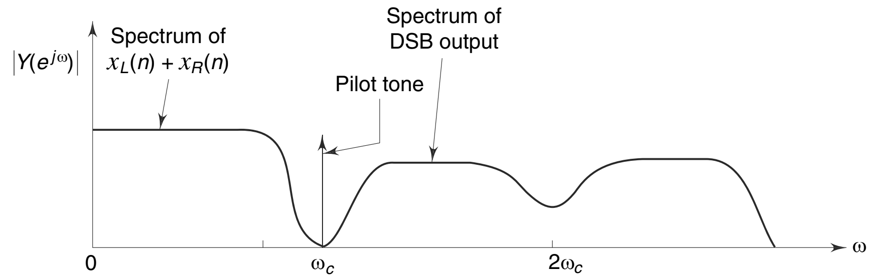


Fig. 15.14 *Spectrum of a Composite FM Stereo Broadcast Signal*

APPLICATIONS TO RADAR 15.4

A radar is used for detecting stationary/moving objects. Figure 15.15 shows the block diagram of a modern radar system. In the transmitter side, the signals are generated and transmitted through the antenna. When these signals hit the target object, a portion of the signal is echoed back. This echo signal is received by the radar system. Depending on the time duration between the transmitted and received signals, the distance at which the target is located can be identified.

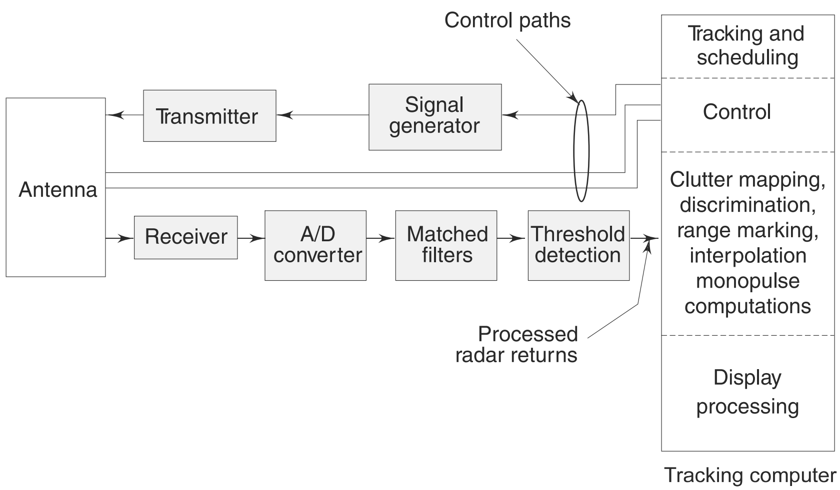


Fig. 15.15 *Block Diagram of a Modern Radar System*

The important components of a radar system are the antenna, the tracking computer and the signal processor. The tracking computer is the brain of the system. The tracking computer has the following major functions:

(i) Schedules the appropriate antenna positions and transmitted signals as a function of time.
(ii) Keeps track of important targets.
(iii) Runs the display.

The major functions of signal processor are

(i) Matched filtering
(ii) Removal of useless information-threshold detection

The tracking computer controls the entire operation of the radar system. Some important radar parameters are discussed in the following section.

Antenna Beamwidth

The beamwidth of an antenna is generally expressed by

$$\beta \alpha \frac{\lambda}{D} \tag{15.15}$$

where

β – beamwidth
λ – wavelength
D – antenna width

For a pencil beam, the antenna geometry is symmetric, β is the same in both the horizontal and vertical dimensions.

Range

The maximum unambiguous range is

$$R_{\max} = \frac{cT}{2} \tag{15.16}$$

where

c – velocity of light $\approx 3 \times 10^8$ m/s
T – pulse repetition interval

Range Resolution

If two targets are present near each other, then the ability of the radar to detect these targets are measured by the range resolution ΔR. If the signal is of constant frequency, then ΔR is determined by the pulse width. If the pulse width is narrowed, then range resolution can be improved but the maximum range is reduced by decreasing the average power.

Doppler Filtering

Moving targets can be identified by using the Doppler effect. When a continuous sine wave of frequency f_0 is transmitted and the target is moving with a constant velocity, then the received echo signal frequency is $f_0 + \Delta f$.

$$\Delta f = \frac{2v}{c} f_0$$

$$= \frac{2v}{\lambda}$$

where f_0 – carrier frequency
v – target velocity
λ – wavelength

Pulsed Doppler signals can be used to obtain both range and velocity resolution.

15.4.1 Signal Design

Transmitting narrow pulse provides good range but poor velocity measurement. A wide pulse of single frequency gives good velocity but bad range information.

Consider the radar model shown in Fig. 15.16. Let the signal be generated digitally and transmitted through an analog filter. The transmitted signal is $s(t)$. The received signal is $s(t-\tau)e^{j2\pi f\,(t-\tau)}$ which is delayed and frequency shifted. The received signal is passed through an analog filter. A/D converter and then through a digital matched filter. The input signal to the matched filter is $s(nT_s-\tau)e^{j2\pi f}(nT_s-\tau)$.

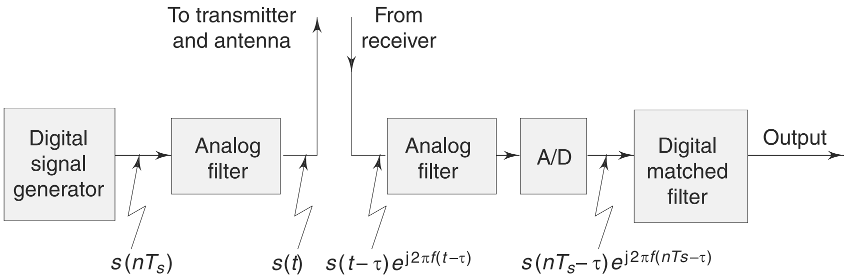


Fig. 15.16 *Block Diagram of a Radar Model*

A long duration signal is required for preserving radar power. But for preserving range resolution, narrow signals are required. This problem can be resolved by designing long duration signals with short duration correlation functions. When the received signal is passed through the appropriate matched filter, a sharp pulse will be available at the filter output. The digital filter can be matched to the signal return for zero range and zero Doppler. Hence its impulse response can be $s^*(-n\,\tau_s)$.

APPLICATIONS TO IMAGE PROCESSING 15.5

2D signal processing is helpful in processing the images. The different processing techniques are image enhancement, image restoration and image coding.

Image enhancement focuses mainly on the features of an image. The various feature enhancements are sharpening the image, edge enhancement, filtering, contrast enhancement, etc. Linear filtering emphasises some spectral regions of the signal. Histogram modification is done on pixel-by-pixel basis and finds its application in contrast equalisation or enhancement. Figure 15.17 shows some image enhancement operations.

Image restoration is for reconstructing an image which is affected by noise, blurred by the movement of the camera or sensor operation. Figure 15.18 shows the basic image restoration process. If the

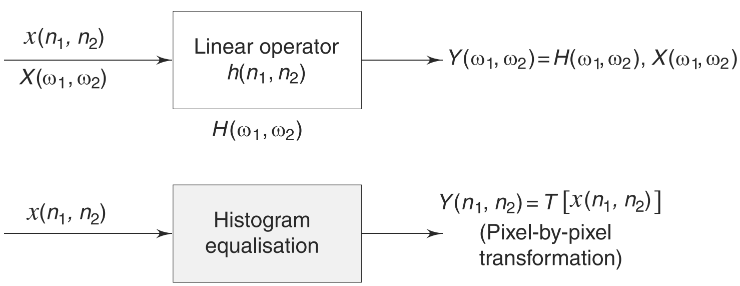


Fig. 15.17 *Image Enhancement Techniques*

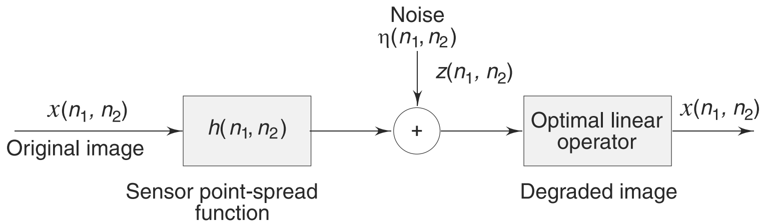


Fig. 15.18 *Basic Image Restoration Process*

image is not affected by noise, the degraded image is passed through an inverse filter. If additive noise is present, then the least square algorithm has to be used along with blurring filter.

Image compression is used in reducing the redundancy present in the image. The number of bits required to store the image information are reduced. The various image compression techniques are

(i) transform coding
(ii) predictive signal coding, and
(iii) sub-band coding

Figure 15.19 shows the various coding techniques. The pulse code modulation is a simple coding technique. The various steps are sampling, quantising and coding. The circuitry is simple. In differential pulse code modulation, compression is done by quantising the prediction error to lesser bits when compared with that of PCM. In transform coding, a coefficient array with unequal variances is obtained by transforming the input array. Optimum bit allocation scheme is used. In sub-band coding the signal is split into different frequency bands of unequal energy. Each frequency band can be encoded in the efficient bit allocation algorithm.

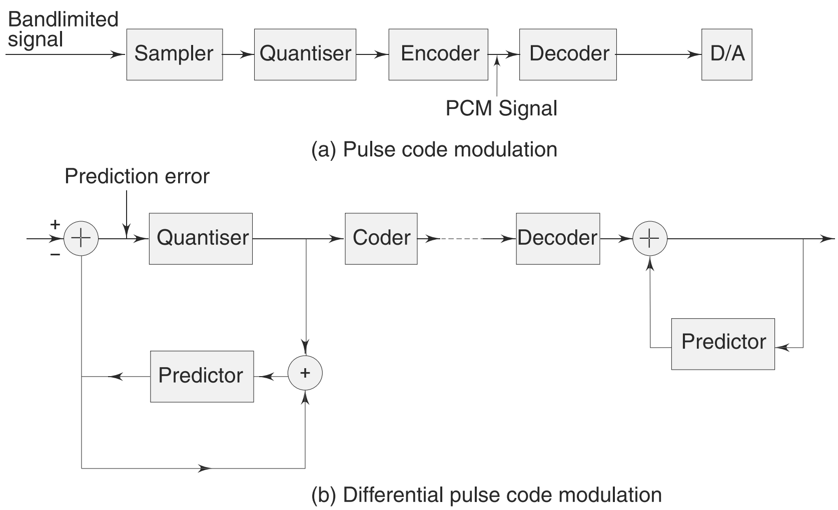


Fig. 15.19 *Image Compression Techniques*

INTRODUCTION TO WAVELETS 15.6

A wavelet is a "small wave", which has its energy concentrated in time. It gives a tool for the analysis of transient, nonstationary, or time-varying phenomena. It not only has an oscillating wavelike characteristic but also has the ability to allow simultaneous time and frequency analysis with a flexible mathematical foundation.

Traditional signal processing techniques such as Fourier transform and short time Fourier transform (STFT) are poorly suited for analysing signals which have abrupt transitions superimposed on lower frequency backgrounds such as speech, music and bio-electric signals. On the other hand, the wavelet transform, a new tool, provides a novel approach to the analysis of such signals. The wavelet transform has a multi-resolution capability. The multi-resolution signal processing used in computer vision, sub-band coding developed for speech and image compression, and wavelet series expansions developed in applied mathematics are recognised as different views of signal theory. The wavelet theory provides a unified framework for a number of techniques which had been developed independently for various signal processing applications.

15.6.1 Time-Frequency Representations

Stationary signals, whose spectral characteristics do not change with time, are represented as a function of time or frequency. Non-stationary signals, which involve both time and frequency, especially the auditory and visual perceptions require time-frequency analysis. The time-frequency analysis involves mapping a signal which is a one-dimensional function of time into an image which is a two-dimensional function of time and frequency, that displays the temporal localisation of the spectral components of signal. The modification required in Fourier transform is localising the analysis, so that it is not necessary to have the signal over $(-\infty, \infty)$ to perform the transform. Hence, this signal has to be mapped as a time-varying spectral representation.

The short time Fourier transform (STFT) can map a one-dimensional function $f(t)$ into the two-dimensional function STFT(τ, f). Here $f(t)$ is assumed to be a stationary signal when viewed through a temporal window, $w(t)$, which is a complex function. Hence, STFT (τ, f) is defined as

$$\text{STFT}\,(\tau, f) = \int_{-\infty}^{\infty} f(t) w^*(t-\tau) e^{-j2\pi f t}\,\mathrm{d}t \tag{15.17}$$

which is the Fourier transform of the windowed signal $f(t)\, w^*(t - \tau)$, where τ is the centre position of the window and the asterisk represents complex conjugation.

The STFT is based on windows of fixed duration. Since the window duration is fixed, the frequency resolution is also fixed, in conformity with the uncertainty principle. For many real world signals such as music and speech, the fixed resolution of the STFT in time and frequency entails serious disadvantages. The wavelet transform circumvents the disadvantages of STFT and it is a form of mapping that has the ability to trade off time resolution for frequency resolution and vice-versa. An advantage of the wavelet transform is that the size of window varies. In order to isolate signal discontinuities, one would like to have a very short basis function. At the same time, in order to obtain detailed frequency analysis, one would like to have a very long basis function. One way to achieve this is to have a short high frequency basis function and a long low frequency function. The wavelet transform does not have single set of basis functions like the sine and cosine functions. Instead, it has an infinite set of possible basis functions.

The wavelet transform is capable of providing the time frequency information simultaneously, i.e. time-frequency representation of the signal. The wavelet transform handles frequency logarithmically rather than linearly, resulting in a time-frequency analysis with the constant $\Delta f/f$, where Δf is the bandwidth and f is the mid-band frequency.

In wavelet analysis, the time-domain signal is passed through various high-pass and low-pass filters, which filter out either high frequency or low frequency portions of the signal. This procedure is repeated until a small portion of the signal corresponding to some frequency is removed from the signal.

Let us consider a signal which has frequencies up to 1000 Hz. The signal is split into two parts by passing the signal through a low-pass (0–500 Hz) and a high-pass (500–1000 Hz) filters. Either the low-pass portion or the high-pass portion or both are taken and the process is repeated again. This process is called *decomposition.*

When the low-pass portion is split again, there are three sets of signals, corresponding to the frequencies 0–250 Hz, 250–500 Hz and 500–1000 Hz. If the lowest frequency signal is split, then there are four sets of signals corresponding to 0–125 Hz, 125–250 Hz, 250–500 Hz and 500–1000 Hz. The process has to be continued until the decomposed signal corresponds to a certain predefined level. These bunch of signals actually represent the same signal, but all correspond to different frequency bands. Now, these signals are represented in a 3D graph, i.e. time on one axis, frequency in the second and amplitude in the third axis. This shows, which frequency exists at a particular point of time.

The uncertainty principle states that it is not exactly known which frequency exists at that time instance, but it provides what frequency bands exist at what time intervals. The spectral components which exist at any given interval of time can be investigated. Wavelet transforms have this variable resolution when compared with STFT which has a fixed resolution at all times.

Higher frequencies are better resolved in time and lower frequencies are better resolved in frequency, i.e. a certain higher frequency component can be located better in time with less relative error and low frequency component can be located better in frequency.

The continuous wavelet transform grid is shown in Fig. 15.20 and the discrete wavelet transform grid is shown in Fig. 15.21. The top row shows that at higher frequencies, there are more samples corresponding to smaller intervals of time, which implies that higher frequencies can be resolved better in time. The bottom row corresponds to lower frequencies, which contains less number of points and hence low frequencies are not resolved well in time.

In a discrete-time case, the time resolution of the signal works similar to the continuous case, but now, the frequency information has different resolutions at every stage too.

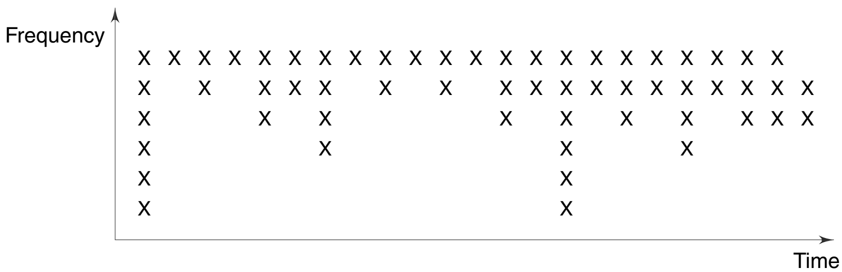


Fig. 15.20 *Continuous Wavelet Transform*

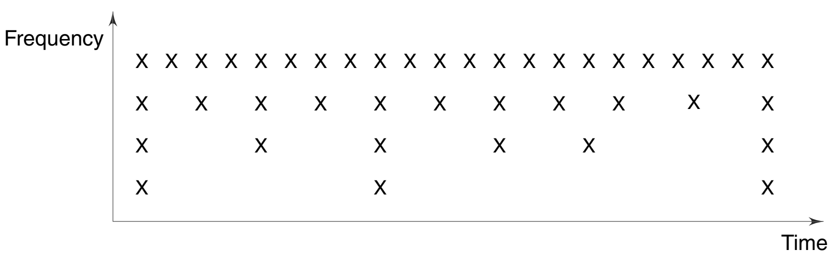


Fig. 15.21 *Discrete Time Wavelet Transform*

15.6.2 Continuous-Time Wavelet

A real or complex-valued continuous function $\psi(t)$ is called a ***basic wavelet*** or ***mother wavelet*** if it satisfies the following two properties and one admissibility condition.

(i) $$\int_{-\infty}^{\infty} \Psi(t)\mathrm{d}t = 0 \tag{15.18}$$

This means that the function integrates to zero. It also suggests that the function is oscillatory and hence there is no need to use sine and cosine waves as in Fourier analysis.

(ii) $$\int_{-\infty}^{\infty} |\Psi(t)|^2 \,\mathrm{d}t < \infty \tag{15.19}$$

It means that the function is square integrable or equivalently has finite energy, i.e. most of the energy in $\psi(t)$ is confined to a finite duration.

(iii) Admissibility condition is given by

$$C \equiv \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|}\mathrm{d}\omega; \quad 0 < C < \infty \tag{15.20}$$

This admissibility condition is helpful in formulating a simple inverse wavelet transform.

15.6.3 Continuous Wavelet Transform (CWT)

Consider $f(t)$ to be a square integrable function i.e., $L^2(R)$ and $\psi(t)$ be a wavelet satisfying the properties mentioned in Eqs. (15.17) and (15.18). The continuous wavelet transform (CWT) of $f(t)$ with respect to $\psi(t)$ is defined as:

$$W(a,b) = \int_{-\infty}^{\infty} f(t)\frac{1}{\sqrt{|a|}}\Psi^*\left(\frac{t-b}{a}\right)\mathrm{d}t \tag{15.21}$$

where a is a scale factor (dilation parameter) and b is the time delay (translation variable). The scale factor a governs its frequency content, the delay parameter b gives the position of the wavelet $\psi_{a,b}(t)$ and * denotes complex conjugation. It is important to note that both $f(t)$ and $\psi(t)$ belong to $L^2(R)$, the set of square integrable functions. The above equation can be written in a more compact form by defining $\psi_{a,b}(t)$ as

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\Psi\left(\frac{t-b}{a}\right) \tag{15.22}$$

Now,

$$W(a,b) = \int_{-\infty}^{\infty} f(t)\Psi^*_{a,b}(t)\mathrm{d}t \tag{15.23}$$

Here, the normalising factor $\left(\frac{1}{\sqrt{|a|}}\right)$ ensures that the energy is same for all a and b, i.e.

$$\int_{-\infty}^{\infty} |\Psi_{a,b}(t)|^2 \,\mathrm{d}t = \int_{-\infty}^{\infty} |\Psi(t)|^2 \,\mathrm{d}t \tag{15.24}$$

From the above discussion, one can understand that the CWT operates on a function of one real variable and transforms it to a function of two real variables. Thus, it can be seen as a mapping operator from a single variable set of functions in $L^2(R)$ to two variable set of functions, i.e. $W_\psi[f(t)] \equiv W(a,b)$, where ψ in the subscript denotes that the transform depends not only on the function $f(t)$ but also on the wavelet, i.e. CWT is with respect to $\psi(t)$ on $f(t)$.

15.6.4 Inverse CWT

Given $W(a, b) = W_\psi[f(t)]$, it is possible to obtain $f(t)$ by applying the inverse CWT. The inverse continuous wavelet transform is given by

$$f(t) = \frac{1}{C}\int_{a=-\infty}^{\infty}\int_{b=-\infty}^{\infty}\frac{1}{|a|^2}W(a,b)\Psi_{a,b}(t)\,da\,db \tag{15.25}$$

where $C = \int_{-\infty}^{\infty}\frac{|\Psi(\omega)|^2}{|\omega|}\,d\omega;\ 0 < C < \infty$, which is the admissibility condition.

The wavelet should satisfy this condition. Thus, the wavelet transform is providing a weighting function for synthesising a given function $f(t)$ from translates and dilates of the wavelet much as the Fourier transform provides a weighting function for synthesising from sine and cosine functions.

15.6.5 Properties of CWT

The various properties of the CWT are given below.

(i) Linearity

If $f(t)$ and $g(t) \in L^2(R)$ and α and β are scalars, then

$$W_\psi[\alpha f(t) + \beta g(t)] = \alpha W_\psi[f(t)] + \beta W_\psi[g(t)] \tag{15.26}$$

(ii) Translation

The CWT of a translated function $f(t-\tau)$ is given by

$$W_\psi[f(t-\tau)] = W(a, b-\tau) \tag{15.27}$$

(iii) Scaling

The CWT of a scaled function, i.e. $\frac{1}{\sqrt{\alpha}}f\left(\frac{t}{\alpha}\right)$ is given by

$$W_\psi\left[\frac{1}{\alpha}f\left(\frac{t}{\alpha}\right)\right] = W\left[\frac{a}{\alpha}, \frac{b}{\alpha}\right] \tag{15.28}$$

(iv) Wavelet Shifing

If $\psi'(t) = \psi(t-\tau)$, then

$$W_\psi[f(t)] = W(a, b + a\tau) \tag{15.29}$$

Note that the translation property is the result of shifting of the signal $f(t)$ while this is due to the shift of wavelet itself.

(v) Wavelet Scaling

Let $\psi'(t) = \frac{1}{\sqrt{|\alpha|}}\left[\psi\left(\frac{t}{\alpha}\right)\right]$, then

$$W_{\psi'}[f(t)] = W(a\alpha, b) \tag{15.30}$$

(vi) Linear Combination of Wavelets

Let $\psi_1(t)$ and $\psi_2(t)$ be two wavelets. If α and β are scalars, then their linear combination, $a\,\psi_1(t) + b\,\psi_2(t)$ is also a wavelet. The CWT with respect to such linear combination of wavelet is given by

$$W_{\alpha\psi_1 + \beta\psi_2}[f(t)] = \alpha W_{\psi_1}[f(t)] + \beta W_{\psi_2}[f(t)] \tag{15.31}$$

(*vii*) *Energy Conservation*

The CWT has an energy conservation property, i.e. similar to Parseval's formula of the Fourier transform.

Let $f(t) \in L^2(R)$ have its continuous wavelet transform as $W(a, b)$, then

$$\int_{-\infty}^{\infty} |f(t)|^2 \, \mathrm{d}t = \frac{1}{c} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |W(a,b)|^2 \frac{\mathrm{d}\alpha \mathrm{d}b}{a^2} \tag{15.32}$$

(*viii*) *Localisation Properties*

The continuous wavelet transform has some localisation properties, in particular, sharp time localisation at high frequencies.

Time localisation Consider a Dirac pulse at time t_0, $\delta(t - t_0)$ and a wavelet $\psi(t)$. The CWT transform of the Dirac pulse is

$$W(a,b) = \frac{1}{\sqrt{\alpha}} \int \psi\left(\frac{t-b}{\alpha}\right) \delta(t - t_0) dt$$

$$= \frac{1}{\sqrt{\alpha}} \psi\left(\frac{t_0 - b}{a}\right) \tag{15.33}$$

For a given scale factor a_0, i.e. a horizontal line in the wavelet domain, the transform is equal to the scaled (and normalised) wavelet reversed in time and centered at the location of the Dirac. For small a's, the transform "zooms-in" to the Dirac with good localisation for very small scales.

Frequency localisation Consider the sinc wavelet, i.e. a perfect bandpass filter. Its magnitude spectrum is 1 for $|\omega|$ between π and 2π. Consider a complex sinusoid of unit magnitude and at frequency ω_0. The highest frequency wavelet that passes the sinusoid having a scale factor of π/ω_0 (gain of $\sqrt{\pi/\omega_0}$) while the low frequency wavelet that passes the sinusoid having a scale factor of $2\pi/\omega_0$ (gain of $\sqrt{2\pi/\omega_0}$).

(*ix*) *Reproducing Kernel*

The CWT is a very redundant representation since it is a 2-D expansion of a 1-D function. Consider the space V of a square integrable function over the plane (a, b) with respect to $\mathrm{d}a\, \mathrm{d}b/a^2$. Only a subspace H of V corresponds to wavelet transforms of functions from $L^2(R)$.

If a function $W(a, b)$ belongs to H, i.e. it is the wavelet transform of $f(t)$, then $W(a, b)$ satisfies

$$W(a_0, b_0) = \frac{1}{C} \int\int K(a_0, b_0, a, b) W(a,b) \frac{da\, db}{a^2} \tag{15.34}$$

where $K(a_0, b_0, a, b) = \langle \psi_{a_0 b_0}, \psi_{a,b} \rangle$ is the reproducing kernel.

15.6.6 Discrete Wavelet Transform

The discrete wavelet transform (DWT) corresponding to a CWT function $W(a, b)$ can be obtained by sampling the co-ordinates (a, b) on a grid as shown in Fig. 15.22. This process is called the ***dyadic sampling*** because the consecutive values of discrete scales as well as the corresponding sampling intervals differs by a factor of two. Then the dilation takes the values of the form $a = 2^k$ and translation takes the values of the form $b = 2^k l$ where k and l are integers. The values of $d(k, l)$ represent the discretised values of CWT $W(a, b)$ at $a = 2^k$ and $b = 2^k l$. The two-dimensional square $d(k, l)$ is commonly referred to as the discrete wavelet transform of $f(t)$. The $f(t)$ can be found using the following equation.

$$f(t) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} d(k,l) 2^{-k/2} \psi(2^{-k} t - l) \tag{15.35}$$

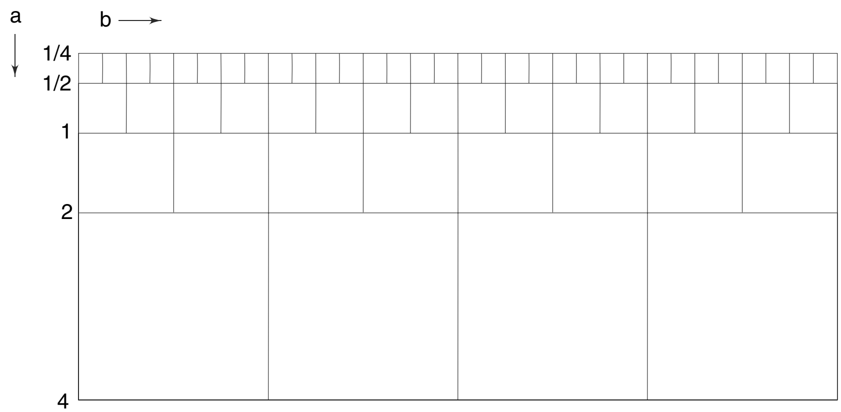


Fig. 15.22 *Time-Frequency Cells that Correspond to Dyadic Sampling*

15.6.7 Comparison of Fourier Transform and Wavelet Transform

Fourier transforms are useful mathematical tools for the analysis of signals whose statistical properties are constant over time or space. The reason being, the Fourier transform represents the signal as the sum of sine and cosine functions which have infinite duration in time. Instead of representing the signals with these functions, localised waves, or wavelets can be used. Representing non-stationary signals as the sum of basic functions which are localised in time leads to more compact representations and also provides better insight into the properties of the signals than what Fourier transform provides.

In wavelet analysis, signals are represented using a set of basis functions. These are derived from a single prototype function called the "mother wavelet". These basis functions, or wavelets, are formed by translating and dilating the mother wavelet. This can also be stated as the basis function formed by shifting and scaling the mother wavelet in time. Hence, the wavelet transform can be viewed as a decomposition of a signal in the time-scale plane.

15.6.8 Application of Wavelet Transform

Applications of wavelet transform include signal compression, image compression, video compression, sound synthesis, statistical analysis, adaptive filtering, solving linear equations, and so on.

Data Compression

The wavelet transform finds wide application in data compression. The reason being DWT is closely related to sub-band decomposition which was already being used for compression. Such sub-band decomposition technique is useful for image, audio and video compressions.

Compression is the process of starting with a source of data in digital form and creating a representation for it that uses fewer bits than the original. This will reduce storage requirements or transmission time when such information is communicated over a distance.

(i) *Image Compression* Wavelet based image compression technique is one of the transform coding methods in which the given signal or image is first transformed to a different domain and then quantised. An image will be decomposed into approximate and detail coefficients, when DWT is applied. Suppose, an *n*th level decomposition of an image is obtained, then the lower levels of decomposition would correspond to higher frequency sub-bands and the *n*th level would correspond to the lowest frequency sub-bands. Generally, most of the energy in an image is concentrated in the low frequency region. Thus, as one moves from higher levels, i.e. region of approximate co-efficients to lower levels, i.e. region of detail co-efficients of sub-band decomposition, there will be a decrease in the energy content of the sub-band decomposition. In other words, most of wavelet co-efficients in the detail region have very low values and these can be quantised to zero without affecting the perceived quality of the image significantly. In addition, there are spatial self-similarities across the sub-bands, i.e. the similarities occupy the same spatial position in each sub-band, but at different resolutions. The embedded zero-tree wavelet (EZW) coding algorithm exploits these properties to give excellent compression results.

(ii) *Video Compression* Today's main stream MPEG (Moving Picture Experts Group) standard is geared towards transmission of sequences of images. These algorithms use a type of compression known as discrete cosine transform (DCT). MPEG compression solutions are more expensive, Houston Advanced Research Centre (Houston, TX), claims HARC-C can achieve still-image compression ratios of about 300 to 1 and video compression ratios up to 480 to 1.

(iii) *Audio Compression* Wavelet based audio compression can be achieved by using wavelet based digital filters in sub-band decomposition. In this, the wavelet packet decomposition of audio frames is obtained and then quantised to keep the distortion below the masking threshold for the frame. The masking threshold of a frame is the minimum sound pressure level (SPL) at which the tone becomes audible. This technique depends on choosing the most appropriate wavelet from a library of wavelets adaptively.

Sound Synthesis

Sound synthesis is a very important application of WT. Victor Wickerhauser has suggested that wavelet packets could be useful in sound synthesis. His idea is that wavelet packet generator could replace a large number of oscillators. Through experimentation, a musician could determine combinations of wave packets that produce especially interesting sounds. Wickerhauser feels that sound synthesis is a natural use of wavelets. A sample of the notes produced by the instrument could be decomposed into its wavelet packet coefficients. Reproducing the note would then require reloading those coefficients into a wave packet generator and playing back the result.

Computer and Human Vision

Marr's theory was that image processing in the human visual system has a complicated hierarchical structure that involves several layers of processing. At each processing level, the retinal system provides a visual representation that scales progressively in a geometrical manner. According to his theory intensity changes occur at different scales in an image, so that their optimal detection requires the use of operators of different sizes and sudden intensity changes produce a peak or trough in the first derivative of the image. These two hypotheses require that a vision filter have two characteristics: it should be a differential operator and capable of being tuned to act a any desired scale. Marr's operator is referred to as a ***Marr wavelet.***

Fingerprint Compression

The US Federal Bureau of Investigation has collected about 30 million sets of fingerprints consisting of inked impressions on paper cards. Because a number of jurisdictions are experimenting with digital storage of the prints, incompatibilities between data formats have recently become a problem. This problem led to a demand in the criminal-justice community for a digitisation and compression standard.

In 1993, the FBI's criminal justice information services division developed standards for fingerprint digitisation and compression.

A large amount of memory is required to store the data of digital fingerprints. Fingerprint images are digitised at a resolution of 500 pixels per inch with 256 levels of gray scale information per pixel. A single fingerprint is about 700,000 pixels and needs about 0.6 Mbytes to store. A pair of hands, then, requires about 6 Mbytes of storage. So digitising the FBI's current archive would result in about 200 terabytes of data. Also, it is important to note that the cost of storing these uncompressed images would be about few hundred million American dollars. Obviously, data compression is important to bring these numbers down.

The data compression standard WSQ (wavelet/scalar quantisation) implements a hand-tuned custom wavelet basis developed after extensive testing on a collection of fingerprints. The best compression ratio achieved with these wavelets is 26 : 1.

Denoising Noisy Data

In diverse fields, scientists are faced with the problem of recovering a true signal from incomplete, indirect, or noisy data. To solve this problem, a technique called wavelet shrinkage and thresholding can be used.

The technique works in the following way. When a data set is decomposed using wavelets, filters that act as averaging filters, and others that produce details are used. Some of the resulting wavelet coefficients correspond to details in the data set. If the details are small, they can be omitted without substantially affecting the main features of the data set. The idea of thresholding, is to set all coefficients to zero, which are less than a particular threshold. These coefficients are used in an inverse wavelet transformation to reconstruct the data set. Figure 15.23 (a) and (b) show the signal, and the denoised signal obtained using the thresholding technique. The technique is a significant step forward in handling noisy data because the denoising is carried out without smoothing out the sharp structures. The result is a cleaned-up signal that still shows important details.

Similarly, in order to denoise the image, the following steps have to be incorporated.

(i) transform the image to the wavelet domain using Coiflets with three vanishing moments.

(ii) apply a threshold at two standard deviations, and

(iii) inverse-transform the image to the signal domain.

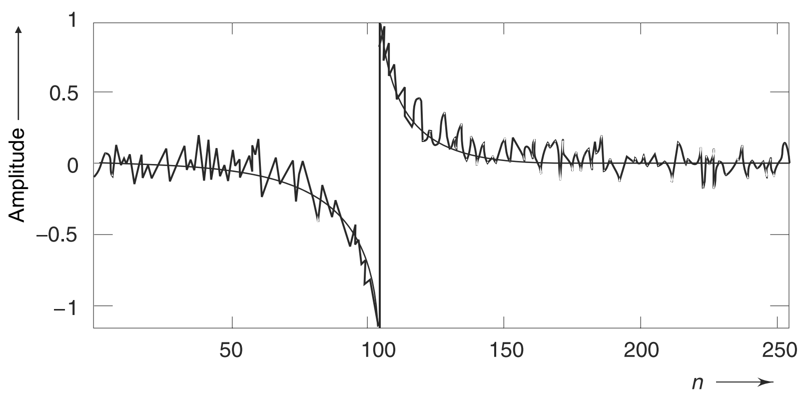


Fig. 15.23 (*a*) *Noisy Signal*

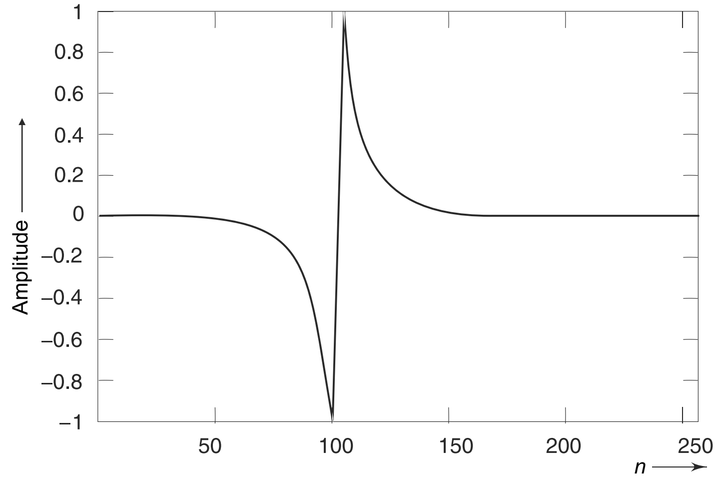


Fig. 15.23 (*b*) *Denoised Signal*

15.7 WIRELESS COMMUNICATION

Orthogonal Frequency Division Multiplexing (OFDM) is a special type of multicarrier modulation technique used in wireless communication systems. It offers several advantages like resilient to multipath fading, immune to intersymbol interference, less complexity, etc., and believed to be a promising technique for future broad band wireless communication. OFDM technique was proposed in late 1950s with the introduction of Frequency Division Multiplexing for data communications. In 1966 Chang patented the structure of OFDM and published the concept of using orthogonal overlapping multi-tone signals for data communications.

In a conventional serial data-transmission system, the information-bearing symbols are transmitted sequentially, with the frequency spectrum of each symbol occupying the entire available bandwidth. OFDM uses the principles of frequency division multiplexing (FDM) to allow multiple messages to be sent over a single radio channel in a controlled manner, allowing an improved spectral efficiency. This is done by converting the data into parallel stream and transmitted by Quadrature Amplitude Modulated

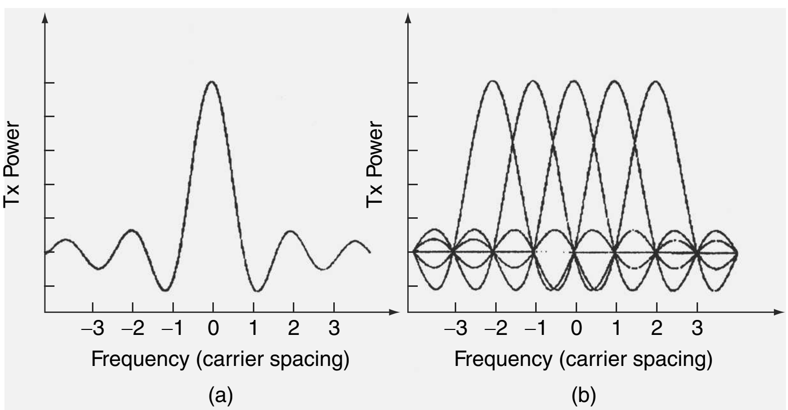


Fig. 15.24 (*a*) *Unfiltered QAM Signal Spectrum and* (*b*) *OFDM Signal Spectrum*

(QAM) subcarriers. OFDM transmits data using a large number of narrow bandwidth subcarriers that are regularly spaced in frequency. The frequency spacing and time synchronisation of the subcarriers are chosen in such a way that the subcarriers overlap each other in the frequency domain. However, they do not cause interference to each other since they are placed orthogonal to each other.

Figure 15.24(a) shows the spectrum of an unfiltered QAM signal. It is a sinc function with zero crossing points at multiples of the reciprocal of the QAM symbol period, T_s. Figure 15.24(b) shows an OFDM signal spectrum, where the sub-carrier spacing is $1/T_s$.

The information is modulated onto a subcarrier by varying the subcarrier's phase, amplitude, or both. In the most basic form, a subcarrier may be present or disabled to indicate a one or zero bit of information. However, either phase shift keying (PSK) or quadrature amplitude modulation (QAM) is typically employed.

15.7.1 Typical OFDM System

A typical OFDM system is shown in Fig. 15.25. The incoming serial data is first converted from serial to parallel and grouped into 'x' bits. The parallel data are modulated in a base band fashion by the IFFT, and converted back to serial data for transmission. A guard interval is inserted between symbols to avoid intersymbol interference (ISI). The discrete symbols are converted to analog and low-pass filtered for RF up-conversion. The receiver performs the inverse process of the transmitter. One tap equaliser is used to correct channel distortion. The tap coefficients of the filter are calculated based on channel information.

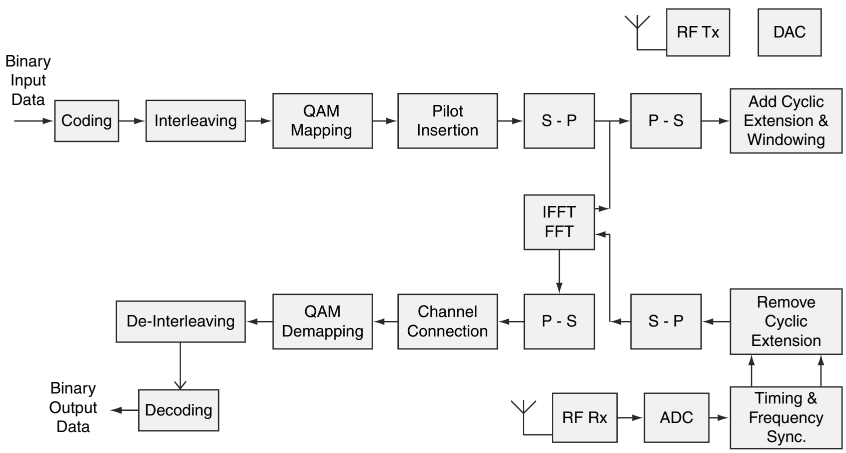


Fig. 15.25 *OFDM Transreceiver*

15.7.2 Inverse Discrete Fourier Transform

Considering an OFDM signal which consists of many parallel QAM sub-carriers as shown in Fig. 15.26, the signal can be mathematically expressed as

$$x(t_m) = \sum_{n=0}^{N-1} (a_n \cos \omega_n t_m + b_n \sin \omega_n t_m)$$

where a_n and b_n are the in-phase and quadrature terms of the QAM signal, and $\omega_n = 2\pi n/N.\Delta t$ is the sub-carrier frequency, $t_m = m.\Delta t$ and Δt is the symbol duration of the input serial data $(a_n + jb_n)$. Generation and demodulation of such a large number of subcarriers require arrays of coherent sinusoidal generators that can become unreasonably complex and expensive. Weinstein exploited the idea of using a Discrete

Fourier Transform (DFT) for implementation of the generation and reception of OFDM signals, since the above equation happens to be the real part of the Inverse Discrete Fourier Transform (IDFT) of the original data $(a_n + jb_n)$. The inverse of the subcarrier spacing, $N.\Delta t$, is defined as the OFDM useful symbol duration, which is N times longer than that of the original data symbol duration Δt.

Since the OFDM signal is the IDFT of the original data, the original data is defined in the frequency domain and the OFDM signal is defined in the time domain. With the technological advances in VLSI, the IDFT can be implemented using Fast Fourier Transform (FFT) algorithm.

Let us consider four subcarriers as shown in Fig. 15.26. All four of these modulated carriers are added to generate an OFDM signal, that is, often produced by an outlined block called the IFFT as shown in Fig. 15.27. The generated OFDM signal and its variation compared to the underlying constant amplitude subcarriers are shown in Fig. 15.28.

Using Inverse FFT to Generate the OFDM Symbol

The function of FFT / IFFT is briefly examined. Forward FFT takes a random signal, multiplies it successively by complex exponentials over the range of frequencies, sums each product and plots the results as a coefficient of that frequency. The coefficients are called a

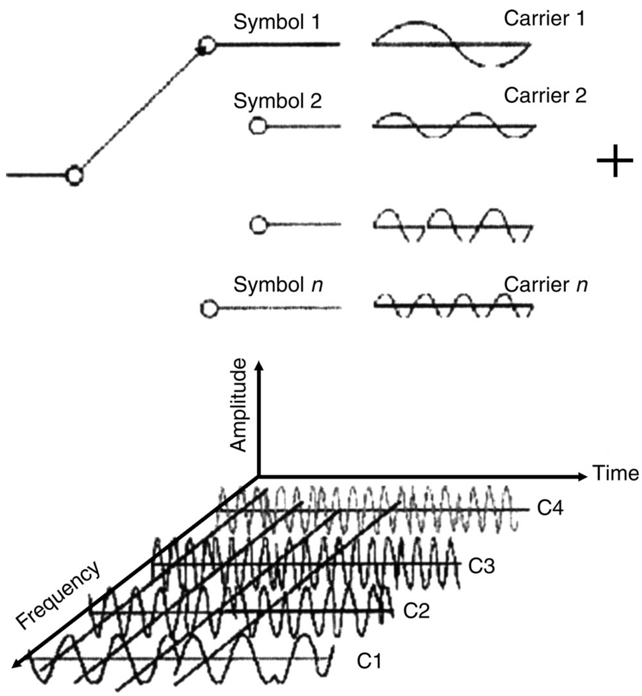


Fig. 15.26 *OFDM Signal in Time and Frequency Domains*

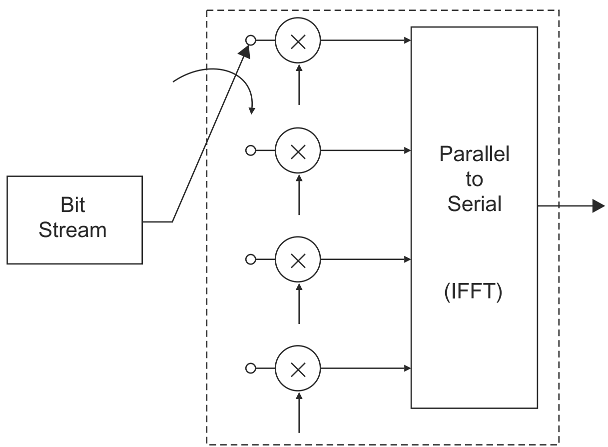


Fig. 15.27 *Functional Diagram of an OFDM Signal Creation*

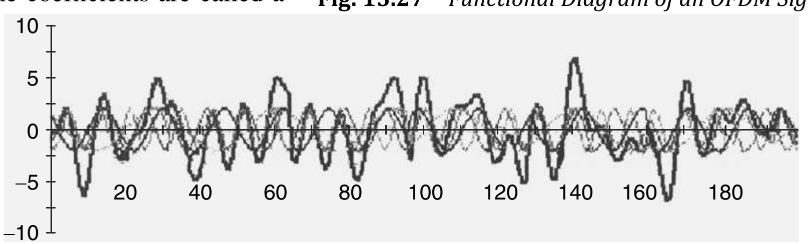


Fig. 15.28 *Generated OFDM Signal and its Variation Compared to the Underlying Constant Amplitude Subcarriers*

spectrum and they represent the frequency present in the output signal. The FFT in complex sinusoids may be expressed as

$$X(k)=\sum_{n=0}^{N-1} x(n)\sin\left(\frac{2\pi kn}{N}\right)+j\sum_{n=0}^{N-1} x(n)\cos\left(\frac{2\pi kn}{N}\right) \tag{15.36}$$

where $x(n)$ is the value of the signal at time n, k is the index of the frequencies over N frequencies, $X(k)$ is the value of the spectrum for the k^{th} frequency. The signal $x(n)$ in time domain and signal $X(k)$ in frequency domain are shown in Fig.15.29.

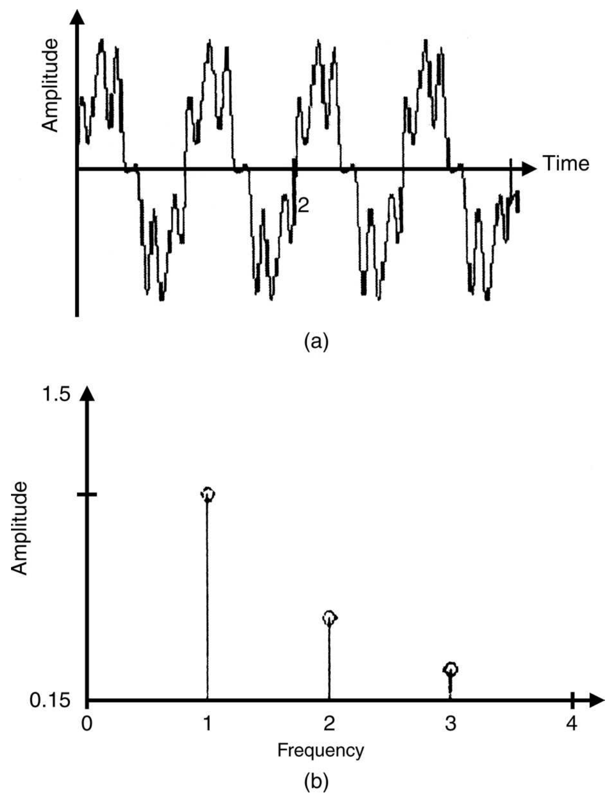


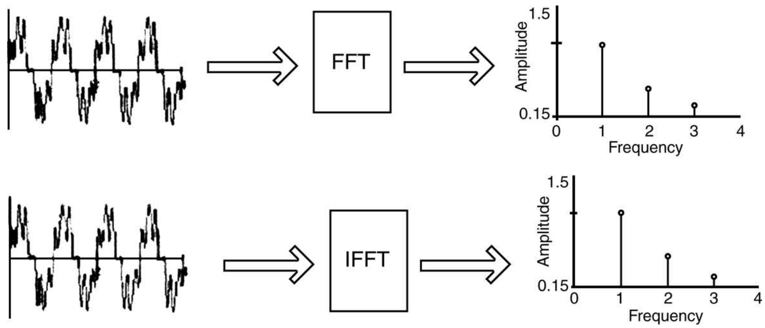
Fig. 15.29 *Signal in (a) Time Domain and (b) Frequency Domain*

The inverse FFT takes this spectrum and converts the whole thing back to time domain signal by again successively multiplying it by a range of sinusoids. The equation of IFFT is

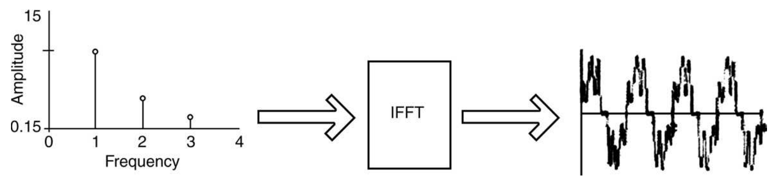
$$x(n)=\sum_{n=0}^{N-1} X(k)\sin\left(\frac{2\pi kn}{N}\right)-j\sum_{n=0}^{N-1} X(k)\cos\left(\frac{2\pi kn}{N}\right) \tag{15.37}$$

The difference between Eqs. (15.36) and (15.37) is the type of coefficient taken by the sinusoids. The coefficients by convention are defined as time domain samples $X(k)$ for the FFT and $x(n)$ frequency bin values for the IFFT.

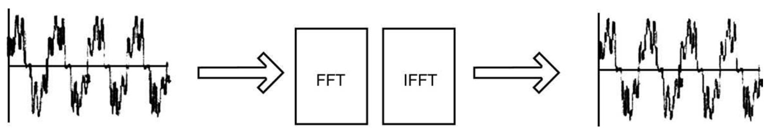
The two processes are a linear pair. The pair is commutable and hence they can be reversed and they will still return the original input. Using both FFT and IFFT in sequence will give the original result back as shown in Fig. 15.30.



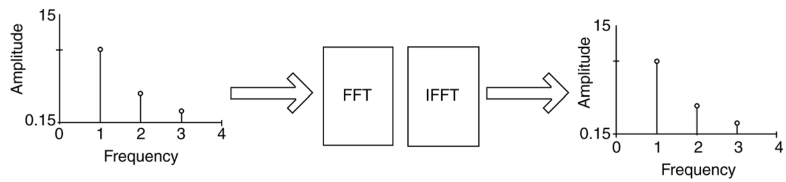
(a) *A Time Domain signal and Similar Spectrum Out of FFT and IFFT*



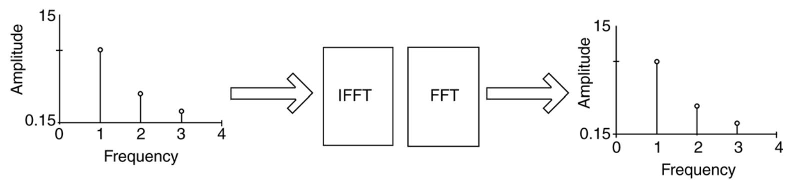
(b) *A Frequency Domain Signal Comes Out as a Time Domain Signal Out of IFFT*



(c) *The Pair Returns Back the Original Input*



(d) *The Pair Returns Back the Original Input*



(e) *The Pair Returns Back the Original Input*

Fig. 15.30 *A Matched Linear Pair of FFT and IFFT*

15.7.3 Coded OFDM

OFDM has recently received increased attention due to its capability of supporting high-data rate communication in frequency selective fading environments which cause ISI. Instead of using a complicated equaliser as in the conventional single carrier systems the ISI in OFDM can be eliminated by adding a guard interval which significantly simplifies the receiver structure. However, in order to take advantage of diversity provided by multipath fading, appropriate frequency interleaving and coding is necessary. Therefore, coding becomes an inseparable part in most of the OFDM applications. The channel coding helps to protect transmitted data and improves the system performance. Trellis coded modulation (TCM) combined with frequency and time interleaving is one of the coded OFDMs and is the most effective means for a frequency selective fading channel.

In a TCM encoder, each symbol of n bits is mapped into a constellation of $n + 1$ bits, using a set-partitioning rule. This process increases the constellation size and adds additional redundance to trellis-code the signal. A TCM can be decoded with a soft decision Viterbi decoding algorithm.

The advantage of Trellis codes is that it produces improvements in the signal-to-noise ratio. However, their performance is hindered by impulsive or burst noise. Transmission errors have a strong time and frequency correlation. Interleaving breaks the correlation and enables the decoder to eliminate of reduce local fading throughout the band, by providing diversity in the time domain.

15.7.4 The Advantages and Disadvantages of OFDM

The OFDM has the advantages and disadvantages as described below:

Advantages OFDM is highly suitable as a modulation technique for many digital communication applications including high speed wireless systems. OFDM is extremely efficient in mitigating common problems affecting high speed communications like multipath fading (leading to ISI). OFDM makes efficient use of the spectrum by allowing overlap due to the use of orthogonal subcarriers. By making use of a set of orthogonal subcarriers closely interspaced, the spectral efficiency of the OFDM transmission system is close to its theoretical maximum.

OFDM is more resistant to frequency selective fading compared to single carrier systems. OFDM effectively eliminates intersymbol interference (ISI) and intercarrier interference (ICI) through the use of a cyclic prefix technique. Using adequate channel coding and interleaving, symbols lost due to the frequency selective nature of the channel can be recovered. Channel equalisation is simpler in OFDM compared to adaptive equalisation techniques with single carrier systems. Using OFDM, it is possible to achieve maximum likelihood decoding with reasonable complexity. OFDM is less sensitive to sample timing offsets than the single carrier systems. It also provides good protection against cochannel interference and impulsive parasitic noise. Enhancement such as the use of guard periods and forward error correction algorithms can lead to improved bit error rates and better channel performance.

Disadvantages The OFDM signal has a noise like amplitude with a very large dynamic range and hence it requires RF power amplifiers with a high peak to average power ratio. It is more sensitive to carrier frequency offset and drift than single carrier systems. Orthogonality between subcarriers has to be maintained over a sample period, failing which ISI occurs.

REVIEW QUESTIONS

15.1 Give the areas in which signal processing finds its application.
15.2 Explain the various stages in voice processing.
15.3 How is a speech signal generated?
15.4 Give the model of human speech production system.
15.5 What is the need for short time spectral analysis?
15.6 Explain in detail the short time Fourier analysis for speech signals.

15.7 What is a vocoder? Explain with a block diagram.
15.8 Describe how targets can be detected using radar.
15.9 Give an expression for the following parameters ralated to radar.
(a) beamwidth and
(b) maximum unambiguous range
15.10 Explain Doppler effect.
15.11 Explain with a block diagram, the modern radar system.
15.12 Discuss signal processing in a radar system.
15.13 Give the various image processing applications.
15.14 Discuss the various coding techniques for images.
15.15 What is the need for image compression?
15.16 Give the basic block diagram of the image restoration process.
15.17 What are the various enhancement techniques in image processing?
15.18 What is sub-band coding?
15.19 Explain with an example the sub-band coding process.
15.20 Explain the process of digital FM stereo signal generation.
15.21 Let the analog pre-emphasis network of an FM stereo be defined by

$$G_a(s) = \frac{S+\Omega_1}{S+\Omega_1+\Omega_2}$$

Obtain the digital pre-emphasis network.
15.22 Explain how privacy can be achieved in telephone communications.
15.23 Define wavelet.
15.24 Distinguish between the short-time Fourier transform and wavelet transform.
15.25 What is decomposition in a wavelet?
15.26 How can better resolution be obtained through wavelet transform?
15.27 Define mother wavelet.
15.28 Define continuous wavelet transform with respect to wavelet.
15.29 Define inverse CWT.
15.30 State and explain the important properties of CWT.
15.31 Define discrete wavelet transform.
15.32 How is discrete wavelet transform obtained from dyadic sampling?
15.33 Discuss briefly the various applications of wavelet transforms.
15.34 What is an OFDM signal? Mention its uses.
15.35 Draw the spectrum of an OFDM signal.
15.36 Why do you need guard interval in OFDM?
15.37 What is an ISI?
15.38 Explain the function of an OFDM system with a suitable block diagram.
15.39 Explain how an OFDM signal is generated?
15.40 Explain how an OFDM signal is generated using inverse FFT?
15.41 What is the function of a matched linear pair of FFT and IFFT?
15.42 Explain the uses of coded OFDM?
15.43 What are the advantages and disadvantages of OFDM?

Chapter 16 Digital Signal Processors

INTRODUCTION 16.1

A digital signal processor is a specialised microprocessor targeted at digital signal processing applications. Digital signal processing applications demand specific features that paved the way for Programmable Digital Signal Processors (P-DSP). Unlike the conventional microprocessors meant for general-purpose applications, the advanced microprocessors such as Reduced Instruction Set Computer (RISC) processors and Complex Instruction Set Computer (CISC) processors may use some of the techniques adopted in P-DSP, or may even have instructions that are specifically required for DSP applications. P-DSP has an advantage over the conventional processor in terms of low power requirement, cost, real-time I/O capability and availability of high-speed on-chip memories.

The salient features required for efficient performance of DSP operations are:

(i) Multiplier and Multiplier Accumulator
(ii) Modified Bus Structure and Memory Access Schemes
(iii) Multiple Access Memory
(iv) Multiported Memory
(v) Very Long Instruction Word (VLIW) Architecture
(vi) Pipelining
(vii) Special Addressing Modes
(viii) On-Chip Peripherals

16.1.1 Advantages of RISC Processor

Owing to the reduced number of instructions, the chip area dedicated to the realisation of the control unit is considerably reduced. The control unit uses about 20% of the chip area in the RISC processor and about 30–40% of the chip area in the CISC processor. Due to the reduction in the control area, the CPU registers and the data paths can be replicated and the throughput of the processor can be increased by applying pipelining and parallel processing.

In an RISC processor, all the instructions are of uniform length and take same time for execution, thereby increasing the computational speed. As an RISC processor has fewer gates when compared to a CISC processor, the propagation delay is reduced and the speed is increased.

16.1.2 Advantages of CISC Processor

Since a CISC processor has a very rich instruction set, it can even support High Level Languages (HLL). The CISC processor uses complex instructions, whereas the RISC processor takes several instructions to execute the function. Hence, the memory requirement is increased for storing many RISC instructions as compared to storing one CISC instruction. It may also lead to increased computation time and make the program difficult to debug.

As HLL compilers are very costly compared to the P-DSP chips, DSP platforms without compilers are preferred and a majority of P-DSP is CISC processor based. With the advancement in technologies, the distinction between RISC and CISC processors in terms of cost and efficiency is diminishing and hence could be designed as per the demands.

16.1.3 Advantages of DSP Processors

DSP processors are often tuned to meet the needs of specific digital signal processing applications. This makes DSP processors strong in terms of performance. DSP processors are also very power efficient, especially when the DSP platforms are designed specifically for low-power, handheld applications such as Texas Instrument (TI) TMS320C5000. DSP processors are typically complemented by powerful and easy-to-use programming and debug tools. This allows for faster development cycles for the desired function and features. The development time can be reduced significantly with proper use of high level programming modules. A DSP processor offers shorter time to market due it its software programmability. For real-time signal processing, DSP processors are rated the best among programmable processors (DSP, RISC) because they have the best and most relevant tool sets to achieve real-time signal-processing functions.

MULTIPLIER AND MULTIPLIER ACCUMULATOR (MAC) 16.2

Array multiplication is one of the most important operations involved in digital signal processing applications. For example, convolution and correlation functions require array multiplication. This can be done with a single multiplier and adder as shown in Fig 16.1. For real-time signals, multiplication and accumulation is to be carried out using hardware elements. This can be achieved by having a dedicated Multiplier and Accumulator Unit (MAC) or to have multiplier and accumulator separate. In Texas Instruments DSP Processor (TMS320C5X), the output of the multiplier is stored in a product register and then added with the accumulator content.

In Fig. 16.1, the output at the nth sampling instant y_n is obtained by multiplying the array $x_n = (x_n, x_{n-1}, x_{n-2}..., x_{n-M+3}, x_{n-M+2}, x_{n-M+1})$ corresponding to the present and the past $M-1$ samples of the input with the array $h = (h_0, h_1, h_2, ..., h_{M-3}, h_{M-2}, h_{M-1})$ corresponding to the impulse response sequence. To obtain the output at '$n + 1$'th instant y_{n+1}, the input signal array x_{n+1} is multiplied with the array h. The vector x_{n+1} is obtained by shifting the array x_n towards right so that the $(n+1)^{th}$ sample of the input data x_{n+1} becomes the first element and all the elements of x_n are shifted towards right by 1 position so that the i^{th} element of x_n becomes the $(i+1)^{th}$ element of x_{n+1}. Instead of shifting the elements of x_n towards right all at a time after completing the vector multiplication, each of the elements may be shifted separately soon after the MAC operation that uses these elements is over. For example,

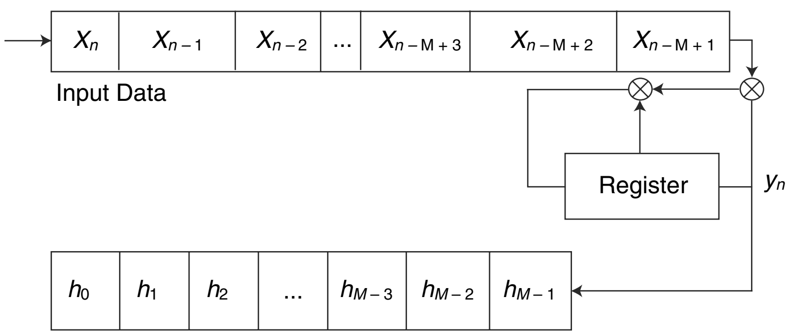


Fig. 16.1 *Implementation of Convolver with Single Multiplier/Adder*

after obtaining the product $x_{n-M+1}h_{M-1}$, the element x_{n-M} may be made equal to x_{n-M+1}. Similarly, after obtaining the product $x_{n-M+2}\,h_{M-2}$, the element x_{n-M+1} may be made equal to x_{n-M+2} and so on. In TMS320C5X, there is an instruction MACD pgm, dma, which multiplies the content of the program memory (pgm) and the content of the data memory with address (dma) and stores the result in the product register. This is then added to the accumulator content before the new product is stored. Then the dma's content is copied in the next location whose address is (dma + 1).

DSP PROCESSOR MEMORY ARCHITECTURE 16.3

DSP processors use special memory architectures, namely, *Harvard architecture* or *modified Von Neumann* architecture, which allow fetching multiple data and/or instructions at the same time. Traditionally, the general purpose processors (GPPs) have used a *Von Neumann* memory architecture in which there is one memory space connected to the processor core by one bus set consisting of an address bus and a data bus. The *Von Neumann* architecture is not good for digital signal processing applications, as some DSP algorithms require more memory bandwidth. For example, in order for a DSP processor to sustain a throughput of one FIR filter tap per instruction cycle, it must perform the following tasks within one instruction cycle.

- Fetch the MAC instruction.
- Read the appropriate sample value from the delay line.
- Read the appropriate coefficient value.
- Write the sample value to the next location in the delay line, in order to shift data through the delay line.

To perform the above task, the DSP processor must make a total of four accesses to memory in one instruction cycle. A processor with *Von Neumann* memory architecture, as shown in Fig. 16.2 (a), would consume a minimum of four instruction cycles for four memory accesses, thus making the goal of one FIR filter tap per cycle unrealisable. Even for the case of the simplest DSP operation, like an addition of two operands and storing the sum to memory, requires four memory accesses, and the *Von Neumann* architecture is not suitable for DSP applications.

The *modified Von Neumann* architecture allows multiple memory accesses per instruction cycle by running the memory clock faster than the instruction cycle. A typical DSP processor runs with an 80 MHz clock, which is divided by four to give 20 million instructions per second. However, the memory clock runs at 80 MHz. This effectively divides each instruction cycle into four 'machine states' and a memory access can be made in each machine state. This allows a total of four memory accesses per instruction cycle. Thus the *modified Von Neumann* architecture sustains a throughput of one FIR filter tap per instruction cycle.

The *Harvard* architecture, shown in Fig. 16.2(b), has two memory spaces, typically partitioned as program memory and data memory. Some modified versions allow some crossover between the two memory spaces. The processor core is connected to these memory spaces by two bus sets that allow simultaneous accesses to memory. The true *Harvard* architecture dedicates one bus for fetching instructions and the other to fetch operands. However, this is inadequate for DSP operations, which usually involve atleast two operands. Therefore, the *DSP Harvard* architecture generally permits the address bus to be used also for access of operands. The DSP operations very often involve fetching the instruction and the two operands at the same time. For this purpose, the *DSP Harvard* architectures also include a cache memory which can be used to store instructions that will be reused, leaving both the buses free for fetching operands. This extended *Harvard* architecture is referred to as *Super Harvard ARChitecture* (SHARC).

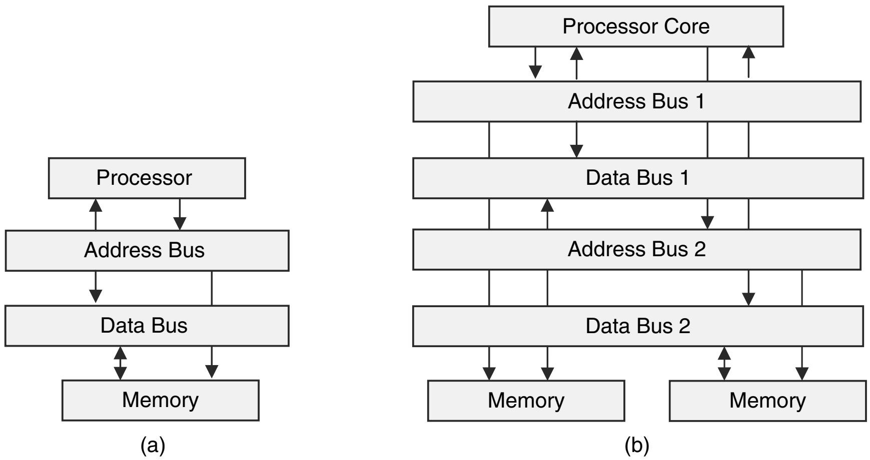


Fig. 16.2 *Memory Architecture: (a) Von Neumann, and (b) Harvard*

Another architecture used for Programmable DSPs, is the very long instruction word (VLIW) architecture. VLIW architecture has a number of processing units. The VLIW is accessed from memory and is used to specify the operands and operations to be performed by each of the data paths. The multiple functional units share a common multiported register file for fetching the operands and storing the results as shown in Fig 16.3. Read wire cross bar facilitates the parallel random access by the functional units. Execution of the operation in the functional units is carried out concurrently with the load/store operation of data between a RAM and the register file. The performance of this type of architecture depends on the degree of parallelism involved in the DSP algorithm and the number of functional units. If there are 8 functional units, the time required for convolution can be reduced by a factor of 8 compared to a single functional unit. The number of functional units is also limited by the hardware cost.

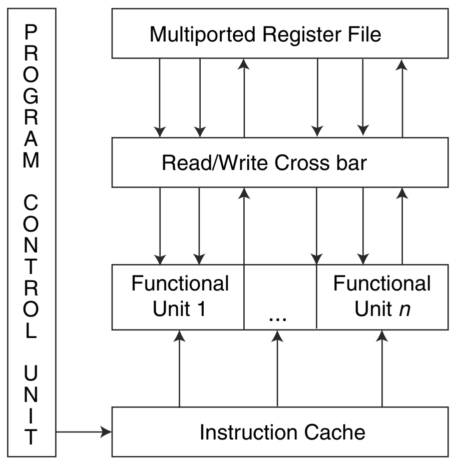


Fig. 16.3 *Block Diagram of the VLIW Architecture*

PIPELINING 16.4

In advanced microprocessors and programmable DSPs, instruction pipelining is adopted which increases the overall efficiency. Let us consider an instruction having four phases as given:

(i) Fetch — instruction is fetched from program memory
(ii) Decode — instruction decode
(iii) Execute — execution phase of the instruction
(iv) Write back — store the results

Assume that there are four functional units to carry out these phases consuming equal time for completion. Now for a processor with no pipelining, each functional unit will be busy only for 25% of the time. It is because, at a particular time, only one instruction is processed by the CPU. Table 16.1 describes when the functional units are busy for three instructions I1, I2, and I3.

Table 16.1 *Instruction cycles of processor with no pipelining*

Time	*Fetch*	*Decode*	*Execute*	*Write back*
1	I1			
2		I1		
3			I1	
4				I1
5	I2			
6		I2		
7			I2	
8				I2
9	I3			
10		I3		
11			I3	
12				I3

Table 16.2 *Instruction cycles of a processor with pipelining*

Time	*Fetch*	*Decode*	*Execute*	*Write back*
1	11			
2	12	11		
3	13	12	11	
4	14	13	12	11
5	15	14	13	12
6	16	15	14	13
7	17	16	15	14
8	18	17	16	15
9	19	18	17	16
10		19	18	17
11			19	18
12				19

It is also possible to keep the functional units busy all the time by simultaneous processing of instruction phases. For a processor with four functional units, when instruction I1 enters for decode phase, instruction I2 can enter for fetch phase. When I1 enters the execute phase, I2 enters the decode phase and I3 can enter the fetch phase, and so on. This is continued until all the functional units are busy. This simultaneous processing of instructions is called pipelining. Table 16.2 describes the execution of nine instructions with pipelining. It can be seen that at time T = 4, all the functional units are busy and are utilised 100%. Comparing Table 16.1 and 16.2, we see that at time T = 12, only three instructions are executed with no pipelining and nine instructions can be executed with pipelining.

SOME EXAMPLES OF DSP PROCESSORS 16.5

Digital signal processing has wide applications that vary from medicine to military and from communications to controls. These days, it has also worked its way into common use from home audio components to answering machines. The increasingly affordable digital signal processing has extended the functionality of embedded systems and plays a larger role in consumer products. This section lists some DSP chip manufacturers and their products.

Analog Devices

- ADSP-2100 Family (16-bit fixed-point)
- ADSP-21020 (32-bit floating-point)
- ADSP-2106x (32-bit floating-point)

AT&T

- DSP16xx (16-bit fixed-point)
- DSP32C/3210 (32-bit floating-point)

Motorola

- DSP56156/166 (16-bit fixed-point)
- DSP56001/2/4 (24-bit fixed-point)
- DSP96002 (32-bit floating-point)

NEC

- μPD77C25 (16-bit fixed-point)
- μPD77017 (16-bit fixed-point)
- μPD77220 (24-bit fixed-point)

Texas Instruments

- TMS320C1x (16-bit fixed-point)
- TMS320C2x (16-bit fixed-point)
- TMS320C3x (32-bit floating-point)
- TMS320C4x (32-bit floating-point)
- TMS320C5x (16-bit fixed-point)
- TMS320C8x (32-bit multiprocessor)

Zilog

- Z89Cxx (16-bit fixed-point)

Zoran Corporation

- ZR38000 (16-bit fixed-point)

Texas Instruments is the single largest manufacturer of DSP chips and enjoys almost 45% of the DSP market share. This chapter therefore focuses on the TI DSP processors.

OVERVIEW OF TMS320 FAMILY DSP PROCESSORS 16.6

The TMS320 family of 16/32-bit single-chip digital signal processors combines the operational flexibility of high-speed controllers and the numerical capability of array processors. Combining these two qualities, the TMS320 processors are inexpensive alternatives to custom-built VLSI and multichip bit-slice processors. The TMS320 family consists of two types of single-chip digital signal processors, namely, 16-bit fixed-point and 32-bit floating-point processors.

The DSPs possess the operational flexibility of high-speed controllers and the numerical capability of array processors. Hence the TMS320 processors are cheap alternatives to custom fabricated VLSI and multichip bit-slice processors. TMS320C5X belongs to the fifth generation of the TI's TMS320 family of DSPs. The first five generations of the TMS320 family C1X, C2X, C3X, C4X and C5X are 16-bit fixed-point processors. Instruction sets of the higher generation fixed-point processors are upward compatible to the lower generation fixed-point processors. For example, C5X can execute the instructions of both C1X and C2X. The 54X is upward compatible with 5X, C3X and C4X are 32-bit floating-point processors and C4X is upward compatible with C3X instruction set. The sixth generation C6X devices feature VelociTI™, an advanced very long instruction word (VLIW) architecture developed by TI and can execute 1600 MIPS. The eighth generation C8X devices, on a single piece of silicon, have a number of advanced DSPs (ADSPs) and an RISC master processor. Typical applications of the above families of TI DSPs are as follows:

C1X, C2X, C2XX, C5X, C54X: Hard disk drives, modems, cellular phones, active car suspensions and toys

C3X: voice mail, imaging, bar-code readers, motor control, 3D graphics or scientific processing, filters, analysers, hi-fi systems

C4X: parallel-processing clusters in virtual reality, image recognition telecom routing, and parallel processing systems

C6X: wireless base stations, pooled modems, remote-access servers, digital subscriber loop systems, cable modems and multichannel telephone systems

C8X: video telephony, 3D computer graphics, virtual reality and multimedia

The TI DSP chips have IC numbers with the prefix TMS320. If the next letter is C (e.g., TMS320C5X), it indicates that CMOS technology is used for the IC and the on-chip non-volatile memory is a ROM. If it is E (e.g., TMS320E5X), it indicates that the technology used is CMOS and the on-chip non-volatile memory is an EPROM. If it is neither (e.g., TMS3205X), it indicates that NMOS technology is used for the IC and the on-chip non-volatile memory is a ROM. Under C5X itself there are three processors, ‘C50, ‘C51 and ‘C5X, that have identical instruction sets but have differences in the capacity of on-chip ROM and RAM. The characteristics of some of the TMS320 family DSP chips are given in Table 16.3.

Table 16.3 *Characteristics of some of the TMS320 family DSP chips*

	‘C15	*‘C25*	*‘C30*	*‘C50*	*‘C541*
Cycle Time (ns)	200	100	60	50	25
On chip RAM	4K	4K	4K	2K	5K
Total memory	4K	128K	16M	128K	128K
Parallel Ports	8	16	16M	64K	64K

The instruction set of TMS320C5X and other DSP chips is superior to the instruction set of conventional microprocessors such as 8085, Z80, etc., as most of the instructions require only a single cycle for execution. The multiply accumulate operation used quite frequently in signal processing applications such as convolution requires only one cycle in DSP. The typical applications of the TMS320 processor family are listed in Table 16.4.

Table 16.4 *Typical applications of TMS320 family*

General-Purpose DSP	*Graphical Imaging*	*Instrumentation*
Digital Filtering	3 D Rotation	Spectrum Analysis
Convolution	Robot Vision	Function Generation
Correlation	Image Transformation/Compression	Pattern Matching
Hilbert Transform		Seismic Processing
Fast Fourier Transform	Pattern Recognition	Transient Analysis
Adaptive Filtering	Homomorphic Processing	Digital Filtering
Windowing	Image Enhancement	Phase Locked Loops
Waveform Generation	Workstations Animation / Digital Map	
Voice / Speech Processing	*Control*	*Military*
Voice Mail	Disk Control	Secure Communication
Speech Vocoding	Servo Control	RADAR Processing
Speech Recognition	Robot Control	SONAR Processing
Speaker Verification	Laser Printer Control	Image Processing
Speech Enhancement	Engine Control	Navigation
Speech Synthesis	Motor Control	Missile Guidance
Text-to-Speech Conversion		Radio Frequency Modems

Telecommunications		Automotive
Echo Cancellation	FAX	Engine Control
ADPCM Transcoders	Cellular Telephones	Vibration Analysis
Digital PBXs	Speaker Phones	Antiskid Brakes
Line Repeaters	X.25 Packet Switching	Adaptive Ride Control
Channel Multiplexing	Video Conferencing	Global Positioning
Adaptive Equalisers	MODEMS	Voice Commands
DTMF Encoding/ Decoding	Data Encryption	Digital Radio
Digital Speech Interpolation	Spread Spectrum Communication	Navigation
Consumer	*Industrial*	*Medical*
RADAR Detectors	Robotics	Hearing Aids
Power Tools	Numeric Control	Patient Monitoring
Digital Audio / TV	Security Access	Ultrasound Equipment
Music Synthesiser	Power line Monitors	Diagnostic Tools
Educational Toys		Prosthetics Fetal Monitors

The TMS320 processor family possesses the following characteristics:

(i) Very flexible instruction set
(ii) Inherent operational flexibility
(iii) High-speed performance
(iv) Innovative and parallel architectural design
(v) Cost effectiveness
(vi) Miniaturisation
(vii) Higher end application orientation

Each generation of TMS320 devices has a CPU and a variety of on-chip memory, and peripherals configured into one processor so that the overall system cost is greatly reduced and device becomes compact.

FIRST-GENERATION TMS320C1X PROCESSOR 16.7

The TMS320C1X generation combines the high performance and specialised features necessary in digital signal processing applications. The design features of the different devices in the TMS320C1X family are given in Table 16.5.

Table 16.5 *Different devices in first-generation of the TMS320C1X family*

Processor	*Design Features*
TMS320C10	A CMOS 20 MHz version of the TMS32010
TMS320C10–14	A 14 MHz version of the TMS320C10
TMS320C15	A TMS320C10 with expanded ROM and RAM
TMS320C16	A 35 MHz expanded memory version of the TMS320C15
TMS320C17	A TMS320C15 with serial and coprocessor ports

16.7.1 Architecture

The unique versatility and real-time performance of the first-generation TMS320C1X processor facilitates flexible design approaches in a variety of applications as given in Table 16.4.

Table 16.6 illustrates the internal hardware of the TMS320C16 processor. The TMS320C1X provides 144/256 words of 16-bit on-chip data RAM and 1.5K/4K words of program ROM/EPROM to support program development. The TMS320C16 has an internal 8K of ROM and can

Table 16.6 *Illustrates the internal hardware of TMS320C16 processor*

Unit	*Function*
Accumulator (ACC)	A 32-bit accumulator divided into a high-order word (bits 31 through 16) and a low-order word (bits 15 through 0). Used for storage of ALU output
Arithmetic Logic Unit (ALU)	A 32-bit 2s-complement arithmetic logic unit with two 32-bit input ports and one 32-bit output port feeding the accumulator
Auxiliary Registers (AR0, AR1)	Two 16-bit registers used for data memory addressing and loop count control. Nine LSBs of each register are configured as up/down counters
Auxiliary Register Pointer (ARP)	A status bit that indicates the currently active auxiliary register
Central Arithmetic Logic Unit (CALU)	The grouping of the ALU, multiplier, accumulator and shifters
Data Bus (D(15-0))	A 16-bit bus used to route data to and from RAM
Data Memory Page Pointer (DP)	A status bit that points to the data RAM address of the current page. A data page contains 128 words
Data RAM	256 words of on-chip random access memory containing data
External Address Bus (A(15-0)/PA (2-0))	A 16-bit bus used to address external program memory. The three LSBs are port addresses in the I/O mode
Interrupt Flag (INTF)	A single-bit flag that indicates an interrupt request has occurred (is pending)
Interrupt Mode (INTM)	A status bit that masks the interrupt flag
Multiplier (MULT)	A 16 × 16-bit parallel hardware multiplier
Overflow Flag (OV)	A status bit flag that indicates an overflow in arithmetic operations
Overflow Mode (OVM)	A status bit that defines a saturated or unsaturated mode in arithmetic operations
P Register (PR)	A 32-bit register containing the product of multiply operations
Program Bus (P(15-0))	A 16-bit bus used to route instructions from program memory
Program Counter (PC(11-0))	A 16-bit register used to address program memory. The PC always contains the address of the next instruction to be executed. The PC contents are updated following each instruction decode operation
Program ROM	8K words of on-chip read-only memory (ROM) containing the program code
Shifters	Two shifters: The ALU barrel shifter that performs a left-shift of 0 to 16-bits on data memory words loaded into the ALU, and the accumulator parallel shifter that performs a left-shift of 0, 1, or 4 places on the entire accumulator and places the resulting high-order bits into data RAM
Stack	An 8 × 16-bit memory used to store the PC during interrupts or calls
Status Register (ST)	A 16-bit status register that contains status and control bits
T Register (TR)	A 16-bit register containing the multiplicand during multiply operations

access 64K of external program space. This on-chip program ROM allows program execution at full speed without the need for high-speed external program memory. External RAM or ROM can be interfaced to the TMS320C1X for those applications requiring external program memory space. This creates multiple functions for external RAM based systems.

The TMS320C1X device provides three separate address spaces for program memory, data memory and input/output (I/O). The TMS320C1X device provides two 16-bit auxiliary registers (AR0-AR1) for indirect addressing of data memory, temporary data storage, and loop control. The content of auxiliary registers can be saved in and loaded from data memory with the SAR (store auxiliary register) and LAR (load auxiliary register) instructions. Three forms of instruction operand addressing, viz. direct, indirect and immediate can be used.

SECOND-GENERATION TMS320C2X PROCESSOR 16.8

The basic architecture of TMS320C2X processor is based upon that of the TMS320C2010, with a number of improvements, including increased address spaces, larger on-chip memories, hardware loop support, and a MAC instruction. The design features of the different devices in the TMS320C2X family are given in Table 16.7.

Table 16.7 *Different devices in the TMS320C2X family*

Processor	*Design Features*
TMS32020	An NMOS processor with source code compatible with TMS32010 has 109 enhanced instructions. It has 544 words on-chip data memory, on-chip serial port and hardware timer.
TMS320C25	A CMOS processor has 133 enhanced instructions, eight auxiliary registers, an eight-level hardware stack, 4K words of on-chip program ROM and a bit-reversed indexed-addressing mode. Low power dissipation and instruction cycle time of less than 100 s.
TMS320C25-50	It is a high-speed version of the TMS320C25, capable of an instruction cycle time of less than 80 ns.
TMS320CE25	It is identical to the TMS320C25, with on-chip 4K words program ROM is replaced with on-chip 4K work program EROM

The TMS320C2X family utilises a modified Harvard architecture for speed and flexibility. The TMS320C2X processor has used single cycle multiply/accumulate (MAC) instructions to increase the throughput of the device. The MAC feature has been accomplished by means of data move option, up to eight auxiliary registers with a dedicated arithmetic unit, and faster I/O necessary for data-intensive signal processing.

The 32-bit ALU and accumulator perform a wide range of arithmetic and logical instructions, the majority of which execute in a single clock cycle. The ALU executes a variety of branch instructions dependent on the status of the ALU or a single bit in a word. One input to the ALU is always provided from the accumulator, and the other input may be provided from the Product Register (PR) of the multiplier or the input scaling shifter, which has fetched data from the RAM on the data bus. After the ALU has performed the arithmetic or logical operations, the result is stored in the accumulator.

The TMS320C2X processor uses separate program and data memory spaces. Each memory space has its own 16-bit address and data buses on-chip. On-chip program ROM is connected to the program address and program data buses. The size and organisation of on-chip ROM and RAM varies depending on the family member.

The second-generation processor has a 16 × 16-bit hardware multiplier capable of computing a signed or unsigned 32-bit product in a single machine cycle. The multiplier has a 16-bit T-register (TR) to hold one of the operands for multiplier and a 32-bit P-register (PR) that holds the product. Single-cycle MAC instruction allows both operands to be processed simultaneously and stores the data in internal or external memory. The data can be transferred to the multiplier at each cycle via the program and data buses.

The TMS320C2X has six registers as follows:

(i) A serial port data receive register
(ii) A serial port data transmit register
(iii) A time register
(iv) A period register
(v) An interrupt mask register
(vi) A global memory allocation register

The TMS320C2X processor has three external maskable user interrupts (INT0, INT1, INT2) and three internal interrupts (generated by serial port, by the timer and by the software interrupt instruction).

The TMS320C2X processor allows flexible configurations as follows:

(i) A stand-alone processor
(ii) A multiprocessor with devices in parallel
(iii) A slave/host multiprocessor with global memory space
(iv) A peripheral processor interfaced via processor-controlled signals to another device

THIRD-GENERATION TMS320C3X PRPCESSOR 16.9

The TMS320C3X processors execute at up to 60 million floating-point operations per second. Due to high degree of on-chip parallelism in processor, allow users to perform up to 11 operations in a single instruction.

The processors possess the following features for high performance and ease of use:

(i) Perform parallel multiply and arithmetic unit operations on integer or floating-point data in a single cycle
(ii) General purpose register file
(iii) Program cache

Table 16.8 *Different devices in TMS320C3X family*

Device	*Frequency (MHz) – (max)*	*Cycle Time (ns)*	*On-chip memory*			*Off-chip memory -parallel*	*Peripherals*		
			RAM	ROM	Cache		Serial	DMA Channels	Timers
C30	50	40	2K	4K	64	16M 32			
8K 32	2	1	2						
C31	60	33	2K	Boot loader	64	16M 32	1	1	2
C32	60	33	512	Boot loader	64	16M 32/16/8	1	2	2

The TMS320C3X offers enhanced address space, multiprocessor interface, internally and externally generated wait states, two external interface ports (one on the C31 and the C32) two timers, two serial ports, (one on the C31 and the C32) and multiple-interrupt structure. A most advanced facility offered by TMS320C3X is the implementation of HLL. The device level comparison of TMS320C3X family is given in Table 16.8.

The high performance of TMS320C3X is achieved through the precision and wide dynamic range of the floating-point units, large on-chip memory, a high degree of parallelism and the DMA controller.

The central processing unit (CPU) of TMS320C3X processor consists of the following components:

(i) Floating-point/integer multiplier
(ii) Arithmetic logic unit (ALU)
(iii) 32-bit barrel shifter
(iv) Internal buses (CPU1/ CPU2 and REG1/ REG2)
(v) Auxiliary register arithmetic units (ARAUs)
(vi) CPU register file

The TMS320C3X processor supports the following interrupts:

(i) Four external interrupts ($\overline{INT3} - \overline{INT0}$)
(ii) Number of internal interrupts
(iii) A non-maskable external $\overline{RESET}$ signal to interrupt either the DMA or the CPU

The TMS320C3X processor supports the following addressing modes:

(i) Register addressing
(ii) Short immediate addressing
(iii) Direct addressing
(iv) Indirect addressing

The TMS320C3X processor supports independent bi-directional serial port, which can be configured to transfer 8, 16, 24, or 32-bits of data per word. The system clock for each serial port can be generated internally or externally. The serial ports can also be configured as timers.

The on-chip DMA controller can read from or write to any location in the memory map without interfering with the CPU operation. The DMA controller contains its own address generators, source and destination registers, and transfer counter. Dedicated DMA address and data buses reduce the conflicts between the CPU and the DMA controller.

FOURTH-GENERATION TMS320C4X PROCESSOR 16.10

The TMS320C4X processor has high performance CPU and DMA controller with up to six communication ports to meet the needs of multiprocessor and I/O-intensive applications. The TMS320C4X processor has several key features as given below:

(i) It can deliver up to 40 MIPS / 80 MFLOPS with a maximum I/O bandwidth of 488 Mbytes/s.
(ii) It has 2K words of on-chip SRAM, 128 words of program cache and a boot loader.
(iii) Two external buses provide an address reach of 4G words of unified memory space.
(iv) Two memory-mapped 32-bit timers
(v) 6 and 12 channel DMA

The TMS320C4X processor has a register-based architecture. The CPU of TMS320C4X processor consists of the following components:

(i) Floating-point/integer multiplier

(ii) Arithmetic Logic Unit (ALU)
(iii) 32-bit barrel shifter
(iv) Internal buses (CPU1/ CPU2 and REG1/REG2)
(v) Auxiliary register arithmetic units (ARAUs)
(vi) CPU register file

FIFTH-GENERATION TMS320C5X PROCESSOR 16.11

The TMS320C5X processor is a fixed-point digital signal processor in the TMS320 family. The TMS320C5X processor provides improved performance over TMS320C1X processor and TMS320C2X processor generations while maintaining upward compatibility of source code between the devices. The CPU of TMS320C5X processor is based on TMS320C25 CPU and incorporated additional architectural enhancement that allow the device to run twice as fast as TMS320C5X processor. The TMS320C5X processor generation is tabulated in Table 16.9.

Table 16.9 *Different devices in TMS320C5X family*

Device	*On-chip memory*			*Power supply*	*I/O Ports*	
	DARAM	*SARAM*	*ROM*		*Serial*	*Parallel*
′C50	1056	9K	2K	5	2 (TDM)	64K
′LC50	1056	9K	2K	3.3	2 (TDM)	64K
′C51	1056	1K	8K	5	2 (TDM)	64K
′LC51	1056	1K	8K	3.3	2 (TDM)	64K
′C52	1056	-	4K	5	1	64K
′LC52	1056	-	4K	3.3	1	64K
′C53	1056	3K	16K	5	2 (TDM)	64K
′C53S	1056	3K	16K	5	2	64K
′LC53	1056	3K	16K	3.3	2 (TDM)	64K
′LC53S	1056	3K	16K	3.3	2	64K
′LC56	1056	6K	32K	3.3	2 (Buffered)	64K
′C57S	1056	6K	2K	5	2 (Buffered)	64K
′LC57	1056	6K	32K	3.3	2 (Buffered)	64K
′LC57S	1056	6K	32K	3.3	2 (Buffered)	64K

Enhanced features of TMS320C5X processor are as follows:

(i) Modular architectural design for increased performance and versatility
(ii) Advanced integrated circuit processing technology for increased performance and low power consumption
(iii) Source code compatibility with ′C1X, ′C2X and ′C2XX processors for fast and easy performance upgrades

The architecture of TMS320C5X processor architecture is shown in Fig. 16.4. The TMS320C5X processor has separate program and data buses to allow simultaneous access to program instructions and data, providing a high degree of parallelism. For example, while data is multiplied, a previous

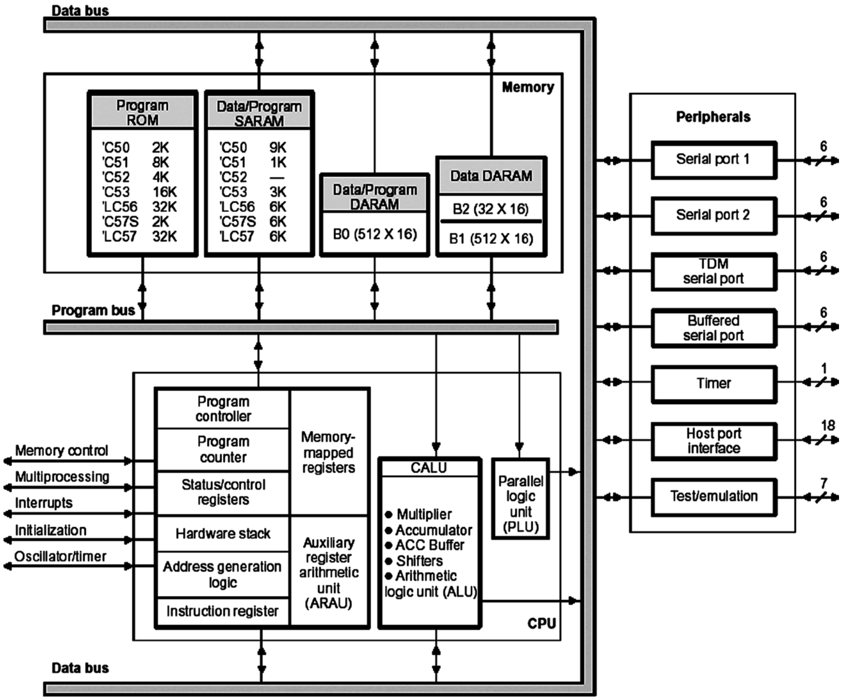


Fig. 16.4 *Architecture of TMS320C5X Processor*

product can be loaded into, added to or subtracted from the accumulator and at the same time, a new address can be generated.

The major buses of the TMS320C5X processor are

(i) Program bus (PB)
(ii) Program address bus (PAB)
(iii) Data read bus (DB)
(iv) Data read address bus (DAB)

The program bus (PB) carries the instruction code and immediate operands from program memory space to the CPU. The program address bus (PAB) provides the addresses to program memory space for both reads and writes. The data read bus (DB) interconnects the various elements of the CPU to data memory space. The program and data buses work together to transfer data from on-chip memory and internal or external program memory to the multiplier for single cycle MAC operations.

16.11.1 Central Processing Unit

The Central Processing Unit of TMS320C5X processor consists of the following:

(i) Central Arithmetic Logic Unit (CALU)
(ii) Parallel Logic Unit (PLU)
(iii) Auxiliary Register Arithmetic Unit (ARAU)
(iv) Memory-mapped registers
(v) Program Controller

Central Arithmetic Logic Unit The central arithmetic logic unit (CALU) used to perform 2s complement arithmetic. The CALU consists of a 16 × 16-bit multiplier, 32-bit arithmetic logic unit (ALU), a 32-bit accumulator (ACC), a 32-bit accumulator buffer (ACCB) and additional shifters at the output of both the accumulator and the product register (PREG).

Parallel Logic Unit The parallel logic unit (PLU) operates parallel with ALU. The PLU performs Boolean operations or the bit manipulations required of high-speed controllers. The PLU can set, clear, test, or toggle bits in a status register, control register, or any data memory location. The PLU provides a direct logical path to data memory values without affecting the contents of the ACC or PREG. The results of a PLU function are written back to the original data memory location.

Auxiliary Register Arithmetic Unit The CPU includes an unsigned 16-bit arithmetic logic unit that calculates indirect addresses by using inputs form the auxiliary registers (ARs), index register (INDX), and auxiliary register compare register (ARCR). The ARAU can auto index the current AR while the data memory location is being addressed and can index either by ±1 or by the contents of the INDX. As a result, accessing data does not require the CALU for address manipulation; therefore, the CALU is free for other operations in parallel.

Memory-Mapped Registers The TMS320C5X processor has 96 registers mapped into page 0 of the data memory space. The family members of TMS320C5X processor have 28 CPU registers and 16 I/O port registers but have different numbers of peripheral and reserved registers. Since the memory-mapped registers are a component of the data memory space, they can be written to and read from in the same way as any other data memory location.

Program Controller The program controller contains logic circuitry that decodes the operational instructions, manages the CPU pipeline, stores the status of CPU operations, and decodes the conditional operations. The basic elements of program controller are

(i) Program Counter
(ii) Status and Control Registers
(iii) Hardware Stack
(iv) Address Generation Logic
(v) Instruction Register

16.11.2 On-Chip Memory

The TMS320C5X processor has the following on-chip memories:

(i) Program read-only memory (ROM)
(ii) Data/program dual-access RAM (DARAM)
(iii) Data/program single-access RAM (SARAM)

The TMS320C5X processor has a total address range of 224K words × 16-bits. The memory space is divided as follows:

(i) 64K Word program memory space
(ii) 64K Word local data memory space
(iii) 64K Word I/O ports
(iv) 32K Word global data memory space

Program ROM The TMS320C5X processor has 16-bit on-chip maskable programmable ROM. The ′C50 - ′C57S processors have boot loader code within the on-chip ROM, all other ′C5X processors offer the boot loader code as an option. This memory is used for booting program code from slower external ROM or EPROM to fast on-chip or external RAM. If the on-chip ROM is not selected, the ′C5X devices start execution from off-chip memory.

Data/Program Dual-Access RAM The TMS320C5X processor has 1056 word × 16-bit on-chip DARAM. The DARAM is primarily intended to store data values, but when needed, can be used to store programs as well. It improves the operational speed of the TMS320C5X processor. The DARAM is divided into three individual selectable memory blocks as given below:

(i) 512 word data or program DARAM block B0
(ii) 512 word data DARAM block B1
(iii) 32 word data DARAM block B2

DARAM blocks B1 and B2 are always configured as data memory where block B0 is configured by software as data or program memory.

Data/Program Single-Access RAM All TMS320C5X processors except the ′C52 carry a 16-bit on-chip SARAM of various sizes. Code can be booted from an off-chip ROM and then executed at full speed, once it is loaded into the on-chip SARAM. The entire range of SARAM can be configured as

(i) SARAM Data memory
(ii) SARAM Program memory
(iii) SARAM Data/Program memory

Though the TMS320C5X processor supports parallel accesses to SARAM blocks, the CPU can read from or write to one SARAM block while accessing another SARAM block. During multiple request from CPU, the SARAM schedules the accesses by providing a not ready condition to the CPU and executing the multiple accesses one cycle at a time. SARAM supports more flexible address mapping than DARAM as it can be mapped to both program and data memory space simultaneously. The on-chip contents are protected by its maskable option.

16.11.3 On-Chip Peripherals

The family of TMS320C5X processor has the same CPU structure but different on-chip peripherals connected to their CPUs. The following peripherals are available in the family of TMS320C5X processor:

(i) Clock Generator
(ii) Hardware Timer
(iii) Software Programmable Wait-State Generator
(iv) Parallel I/O Ports
(v) Host Port Interface (HPI)
(vi) Serial Port
(vii) Buffered Serial Port (BSP)
(viii) Time Division Multiplexed (TDM) Serial Port
(ix) User Maskable Interrupts

The clock generator consists of a crystal resonator circuit for clock source and a phase-locked loop (PLL) to generate an internal CPU clock by multiplying the clock source by a specific factor. The TMS320C5X processor has a 16-bit programmable hardware timer with a 4-bit prescaler. It clocks at a rate 1/2 and 1/32 of the machine cycle rate, depending upon the divide-down ratio of the timer. The timer can be stopped, restarted, reset or disabled by specific status bits.

The TMS320C5X processor has 64K I/O ports. Sixteen of these ports are memory-mapped in data memory space. These I/O ports are addressed by IN and the OUT instructions.

The TMS320C5X processor consists of three types of serial ports:

(i) A general purpose serial port
(ii) A time-division multiplexed (TDM) serial port
(iii) A buffered serial port (BSP)

The TMS320C5X processor family contains atleast one general-purpose serial port. The general purpose serial port is a high-speed synchronous, full-duplexed serial port interface that provides direct communication with serial devices such as codecs, serial analog-to-digital converter (ADC) etc. controlled by maskable external interrupt signal. It is capable of operating at up to 1/4 the machine cycle rate.

The TMS320C5X processor has the following interrupts:

(i) Four external interrupt lines ($\overline{INT}1$ – $\overline{INT}4$)
(ii) Five internal interrupts, a timer interrupt
(iii) Four serial port interrupts (user maskable)

When the interrupt service routine (ISR) is executed, the contents of the program counter are saved on an 8 level hardware stack, and the contents of the eleven specific CPU registers are automatically saved on a one-level-deep stack. When a return from interrupt instruction is executed, the contents of the CPU registers are restored.

16.11.4 Addressing Modes

TMS320C5X processors can address 64K words of program memory and 96K words of data memory. The following addressing modes are supported by TMS320C5X processors.

(i) Direct addressing
(ii) Memory-mapped register addressing
(iii) Immediate addressing
(iv) Indirect addressing
(v) Dedicated-register addressing
(vi) Circular addressing

Direct Addressing

The data memory used with C5X processors is split into 512 pages each of 128 words long. The data memory page pointer (DP) in ST0 holds the address of the current data memory page. In the direct addressing mode of C5X, only lower-order 7 bits of the address are specified in the instruction. The upper 9 bits are taken from the DP as shown in Fig. 16.5.

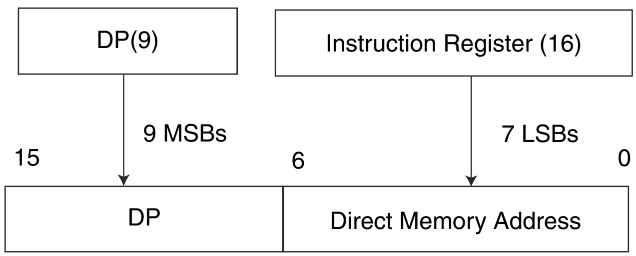


Fig. 16.5 *16-bit Data Memory Address Bus (DAB)*

Memory Mapped Register Addressing

With memory-mapped register addressing, the MMRs can be modified without affecting the current data page pointer value. In addition, any scratch pad RAM (DARAM B2) location or data page 0 can also be modified. The memory-mapped register addressing mode is similar to direct addressing mode, except that the 9 MSBs of the address are forced to 0 instead of being loaded with the contents of the DP. The following instructions operate in the memory-mapped register addressing mode. Using these instructions does not affect the contents of memory-mapped register addressing mode and the contents of the DP.

LAMM-Load accumulator with memory-mapped register

LMMR-Load memory-mapped register

SAMM-Store accumulator in memory-mapped register
SMMR-Store memory-mapped register

Immediate Addressing

The immediate addressing mode can be used to load either a 16-bit constant or a constant of length 13, 9 or 7. Accordingly, it is referred to as long immediate or short immediate addressing mode. This mode is indicated by the symbol #.

Example: ADD # 56h adds 56h to ACC. Similarly, ADD# 4567h adds 4567h to ACC.

Indirect Addressing

The ARs AR0 – AR7 are used for accessing data using indirect addressing mode. In the indirect addressing mode, out of eight ARs, the one which is currently used for accessing data is denoted by the register ARP. The contents of ARP can be temporarily stored in the ARB register. The indirect addressing mode of C5X permits the AR used for the addressing to be updated automatically either after or before the operand is fetched. Hence a separate instruction is not required to update the AR. However, if required, the contents of an AR can be incremented or decremented by any 8-bit constant using SBRK and ADRK instructions. For example, SBRK #k, ADRK #k subtracts, adds the constant k from/to the AR pointed by ARP.

In the indirect addressing mode, the manner in which the memory address is computed and the manner in which the AR is altered after the instruction depends on the instruction. This is indicated to the assembler by the symbols * , *+ , *− , *0+, *0−, *BRO+, *BRO–. The symbol used to indicate the indirect addressing mode and the action taken after executing the instruction are given below.

*	AR unaltered
*+	AR incremented by 1
*−	AR decremented by 1
*0+	AR incremented by the content of INDX
*0−	AR decremented by the content of INDX
*BR0+	AR incremented by the content of INDX with reverse carry propagation
*BR0−	AR decremented by the content of INDX with reverse carry propagation

Dedicated Register Addressing

The dedicated-registered addressing mode operates like the long immediate addressing mode, except that the address comes from one of two special-purpose memory-mapped registers in the CPU: the block move address register (BMAR) and the dynamic bit manipulation register (DBMR). The advantage of this addressing mode is that the address of the block of memory to be acted upon can be changed during execution of the program.

Example: BLDD BMAR, DAT100; DP = 0

If BMAR contains the value 200h, then the content of data memory location 200h is copied to data memory location 100 on the current data page.

Circular Addressing

Algorithms such as convolution, correlation and finite impulse response (FIR) filter use circular buffers in memory to implement a sliding window, which contains the most recent data to be processed. The C5X supports two concurrent circular buffers operating via the ARs. The following are the five memory-mapped registers that control the circular buffer operation:

CBSR1 – Circular buffer 1 start register
CBSR2 – Circular buffer 2 start register
CBER1 – Circular buffer 1 end register

CBER2 – Circular buffer 2 end register

CBCR – Circular buffer control register

The 8-bit CBCR enables and disables the circular buffer operation. To define circular buffers, the start and end addresses are loaded into the corresponding buffer registers first; next, a value between the start and end registers for the circular buffer is loaded into AR. The corresponding circular buffer enables bit in the CBCR should be set.

16.11.5 Instruction Set Summary

Accumulator Memory Reference Instructions

Mnemonic	*Description*
ABS	Absolute value of ACC; zero carry bit
ADCB	Add ACCB and carry bit to ACC
ADD	Add data memory value, with left shift, to ACC
	Add data memory value, with left shift of 16, to ACC
	Add short immediate to ACC
	Add long immediate, with left shift, to ACC
ADDB	Add ACCB to ACC
ADDC	Add data memory value and carry bit to ACC with sign extension suppressed
ADDS	Add data memory value to ACC with sign extension suppressed
ADDT	Add data memory value, with left shift specified by TREG 1, to ACC
AND	AND data memory value with ACCL; zero ACCH
	AND long immediate, with left shift, with ACC
	AND long immediate, with left shift of 16, with ACC
ADNB	AND ACCB with ACC
BSAR	Barrel-shift ACC right
	CMPL 1 s complement ACC
CRGT	Store ACC in ACCB if ACC > ACCB
CRLT	Store ACC in ACCB if ACC < ACCB
EXAR	Exchange ACCB with ACC
LACB	Load ACC to ACCB
LACC	Load data memory value, with left shift, to ACC
	Load long immediate, with left shift, to ACC
	Load data memory value, with left shift of 16, to ACC
LACL	Load data memory value to ACCL; zero ACCH
	Load short immediate to ACCL; zero ACCH
LACT	Load data memory value, with left shift specified by TREG 1, to ACC
LAMM	Load contents of memory-mapped register to ACCL; zero ACCH
NEG	Negate (2s complement) ACC
NORM	Normalise ACC
OR	OR data memory value with ACCL
	OR long immediate, with left shift, with ACC
	OR long immediate, with left shift of 16, with ACC
ORB	OR ACCB with ACC
ROL	Rotate ACC left 1 bit
ROLB	Rotate ACCB and ACC left 1 bit

ROR Rotate acc right 1 bit
RORB Rotate ACCB and ACC right I bit
SACB Store ACC in ACCB
SACH Store ACCH, with left shift, in data memory location
SACL Store ACCL, with left shift, in data memory location
SAMM Store ACCL in memory-mapped register
SATH Barrel-shift ACC right 0 or 16 bits as specified by TREG I
SATL Barrel-shift ACC right as specified by TREG 1
SBB Subtract ACCB from ACC
SBBB Subtract ACCB and logical inversion of carry bit from ACC
SFL Shift ACC left 1 bit
SFLB Shift ACCB and ACC left 1 bit
SFR Shift ACC right 1 bit
SFRB Shift ACCB and ACC right 1 bit
SUB Subtract data memory value, with left shift, from ACC
SUBB Subtract data memory value and logical inversion of carry bit from ACC with sign extension suppressed
SUBS Subtract data memory value from ACC with sign extension suppressed
SUBT Subtract data memory value, with left shift specified by TREG 1, from ACC
XOR Exclusive-OR data memory value with ACCL
XORB Exclusive-OR ACCB with ACC
ZALR Zero ACCL and load ACCH with rounding
ZAP Zero ACC and PREG

Auxiliary Registers and Data Memory Page Pointer Instructions

Mnemonic Description

ADRK Add short immediate to AR
CMPR Compare AR with ARCR as specified by CM bits
LAR Load data memory value to ARx
LDP Load data memory value to DP bits
MAR Modify AR
SAR Store ARx in data memory location
SBRK Subtract short immediate from AR

Parallel Logic Unit (PLU) Instructions

Mnemonic Description

APL AND data memory value with DBMR, and store result in data memory location
AND data memory value with long immediate and store result in data memory location
CPL Compare data memory value with DBMR
Compare data memory value with long immediate
OPL OR data memory value with DBMR and store result in data memory location
OR Data memory value with long immediate and store result in data memory location
SPLK Store long immediate in data memory location
XPL Exclusive-OR data memory value with DBMR and store result in data memory location
Exclusive-OR data memory value with long immediate and store result in data memory location
LPH Load data memory value to PREG high byte
LT Load data memory value to TREG0

TREG0, PREG and Multiply Instructions

Mnemonic	Description
LTA	Load data memory value to TREG0; add PREG, with shift specified by PM bits, to ACC
LTD	Load data memory value to TREG0; add PREG, with shift specified by PM bits, to ACC; and move data
LTP	Load data memory value to TREG0; store PREG, with shift specified by PM bits, in ACC
LTS	Load data memory value to TREG0; subtract PREG, with shift specified by PM bits, from ACC
MAC	Add PREG, with shift specified by PM bits, to ACC; load data memory value to TREG0; multiply data memory value by program memory value and store result in PREG
MACD	Add PREG, with shift specified by PM bits, to ACC; load data memory value to TREG0; multiply data memory value by program memory value and store result in PREG; and move data
MADD	Add PREG, with shift specified by PM bits, to ACC; load data memory value to TREG0; multiply data memory value by value specified in BMAR and store result in PREG; and move data
MADS	Add PREG, with shift specified by PM bits, to ACC; load data memory value to TREG0; multiply data memory value by value specified in BMAR and store result in PREG
MPY	Multiply data memory value by TREG0 and store result in PREG Multiply short immediate by TREG0 and store result in PREG Multiply long immediate by TREG0 and store result in PREG
MPYA	Add PREG, with shift specified by PM bits, to ACC; multiply data memory value by TREG0 and store result in PREG

Mnemonic	Description
MPYS	Subtract PREG, with shift specified by PM bits, from ACC; multiply data memory value by TREG0 and store result in PREG
MPYU	Multiply unsigned data memory value by TREG0 and store result in PREG
PAC	Load PREG, with shift specified by PM bits, to ACC
SPAC	Subtract PREG, with shift specified by PM bits, from ACC
SPH	Store PREG high byte, with shift specified by PM bits, in data memory location
SPL	Store PREG low byte, with shift specified by PM bits, in data memory location
SPM	Set product shift mode (PM) bits
SQRA	Add PREG, with shift specified by PM bits, to ACC; load data memory value to TREG0; square value and store result in PREG
SQRS	Subtract PREG, with shift specified by PM bits, from ACC; load data memory value to TREG0; square value and store result in PREG
ZPR	Zero PREG

Branch and Call Instructions

Mnemonic	Description
B	Branch unconditionally to program memory location
BACC	Branch to program memory location specified by ACCL
BACCD	Delayed branch to program memory location specified by ACCL
BANZ	Branch to program memory location if AR not zero
BANZD	Delayed branch to program memory location if AR not zero
BCND	Branch conditionally to program memory location
BCNDD	Delayed branch conditionally to program memory location
BD	Delayed branch unconditionally to program memory location

CALA Call to subroutine addressed by ACCL
CALAD Delayed call to subroutine addressed by ACCL
CALL Call to subroutine unconditionally
CALLD Delayed call to subroutine unconditionally
CC Call to subroutine conditionally
CCD Delayed call to subroutine conditionally
INTR Software interrupt that branches program control to program memory location
NMI Nonmaskable interrupt and globally disable interrupts (INTM = 1)
RET Return from subroutine
RETC Return from subroutine conditionally
RETCD Delayed return from subroutine conditionally
RETD Delayed return from subroutine
RETE Return from interrupt with context switch and globally enable interrupts (INTM = 0) RET1
Return from interrupt with context switch
TRAP Software interrupt that branches program control to program memory location 22h
XC Execute next instruction(s) conditionally

I/O and Data Memory Instructions

Mnemonic Description

BLDD Block move from data to data memory
Block move from data to data memory with destination address long immediate
Block move from data to data memory with source address in BMAR
Block move from data to data memory with destination address in BMAR
BLDP Block move from data to program memory with destination address in BMAR
BLPD Block move from program to data memory with source address in BMAR
Block move from program to data memory with source address long immediate
DMOV Move data in data memory
IN Input data from I/O port to data memory location
LMMR Load data memory value to memory-mapped register
OUT Output data from data memory location to I/O port
SMMR Store memory-mapped register in data memory location
TBLR Transfer data from program to data memory with source address in ACCL
TBLW Transfer data from data to program memory with destination address in ACCL

Control Instructions

Mnemonic Description

BIT Test bit BITT Test bit specified by TREG2
CLRC Clear overflow mode (OVM) bit
Clear sign extension mode (SXM) bit
Clear hold mode (hM) bit
Clear test/control (TC) bit
Clear carry (C) bit
Clear configuration control (CNF) bit
Clear interrupt mode (INTM) bit
Clear external flag (XF) pin
IDLE Idle until non-maskable interrupt or reset
IDLE2 Idle until non-maskable interrupt or reset, low-power mode
LST Load data memory value to STO
Load data memory value to ST1

NOP	No operation
POP	Pop top of stack to ACCL; zero ACCH
POPD	Pop top of stack to data memory location
PSHD	Push data memory value to top of stack
PUSH	Push ACCL to top of stack
RPT	Repeat next instruction specified by data memory value
	Repeat next instruction specified by short immediate
	Repeat next instruction specified by long immediate
RPTB	Repeat block of instructions specified by BRCR
RPTZ	Clear ACC and PREG; repeat next instruction specified by long immediate
SETC	Set overflow mode (OVM) bit
	Set sign extension mode (SXM) bit
	Set hold mode (hM) bit
	Set test/control (TC) bit
	Set carry (C) bit
	Set external flag (XF) pin high
	Set configuration control (CNF) bit
	Set interrupt mode (INTM) bit
SST	Store ST0 in data memory location
	Store ST1 in data memory location

16.11.6 Pipelining in TMS320C5X

The pipeline is invisible to the user except in some cases, such as AR updates, memory-mapped accesses of the CPU registers, and NORM instruction and memory configuration commands. Furthermore, the pipeline operation is not protected. The user has to understand the pipeline operation to avoid the pipeline conflict by arranging the code. The following example shows the normal pipeline operation with four phases of instruction execution.

Consider the following program involving only single-word single-cycle instructions:

```
ADD * +
SAMM TREG0
MPY* +
SQRA * + , AR2
```

The instruction pipeline for the above program is shown in Table 16.10.

Table 16.10 *Pipeline operation of 1-word instruction*

Cycle	*PC*	*Fetch*	*Decode*	*Read*	*Execute*
1	[SAMM]	ADD			
2	[MPY]	SAMM	ADD		
3	[SQRA]	MPY	SAMM	ADD	
4		SQRA	MPY	SAMM	ADD
5			SQRA	MPY	SAMM
6				SQRA	MPY
7					SQRA

16.11.7 Architecture of TMS320C54X

The internal hardware architecture of TMS320C54X is shown in Fig 16.6. The following describes the various units of the processor in detail.

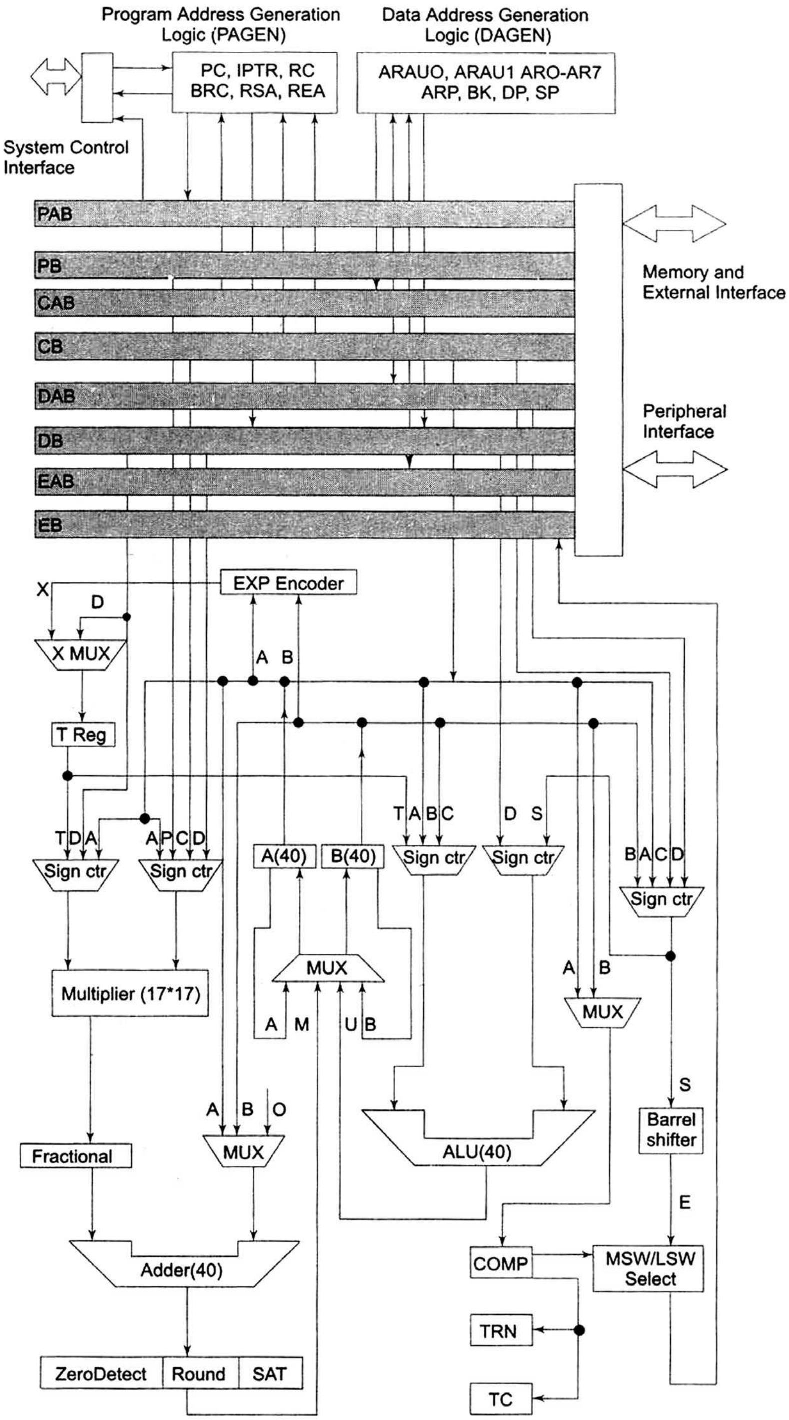


Fig. 16.6 *Block Diagram of 54X Internal Hardware*

'54X Buses The 54X processor has four 16 bit program/data buses and four 16 bit address buses. The program bus (PB) carries the instruction code and immediate operands from program memory. Three data buses (CB, DB and EB) interconnect to various elements, such as the CPU, data address generation logic, program address generation logic, on-chip peripherals and data memory. The CB and DB carry the operands that are read from data memory. The EB carries the data to be written to memory. Four address buses (PAB, CAB, DAB and EAB) carry the addresses needed for instruction execution.

The '54X also has an on-chip bidirectional bus for accessing on-chip peripherals; this bus is connected to DB and EB through the bus exchanger in the CPU interface. Accesses that use this bus can require two or more cycles for reads and writes, depending on the peripheral's structure.

Internal Memory Organisation The '54X memory is organised into program memory, data memory and I/O space. All '54X devices contain both random access memory (RAM) and read only memory (ROM). There are two types of RAM namely dual access RAM (DARAM) and single-access RAM (SARAM). They can be configured as either program memory or data memory.

On-Chip ROM The on-chip ROM is a part of the program memory space and a part of the data memory space in some cases. On devices with a small amount of ROM (2K words), the ROM contains a boot loader, which is useful for booting to faster on-chip or external RAM. On devices with larger amounts of ROM, a portion of the ROM may be mapped into both data and program space.

On-Chip Dual-Access RAM (DARAM) The DARAM is composed of several blocks. Each DARAM block can be accessed twice per machine cycle. Thus the CPU can read from and write to a single block of DARAM in the same cycle. The DARAM is always mapped in data space, and is primarily intended to store data values. It can also be mapped into program space and used to store program code.

On-Chip Single-Access RAM (SARAM) The SARAM is also composed of several blocks. Each block is accessible once per machine cycle for either a read or a write. The SARAM is always mapped in data space and is primarily intended to store data values. It can also be mapped into program space and used to store program code.

On-Chip Memory Security The '54X has a maskable memory security option that protects the contents of on-chip memories. When this option is chosen, no externally originating instruction can access the on-chip memory spaces.

Memory-Mapped Registers The data memory space contains memory-mapped registers for the CPU and the on-chip peripherals. These registers are located on data page 0, simplifying access to them. The memory-mapped access provides a convenient way to save and restore the registers for context switches and to transfer information between the accumulators and the other registers.

Central Processing Unit (CPU) The internal hardware of 54X CPU contains the following units.

40-Bit Arithmetic Logic Unit (ALU)
Two 40-Bit Accumulator Registers
Barrel Shifter Supporting A -16 To 31 Shift Range
Multiply/Accumulate Block
16-Bit Temporary Register (T)
16-Bit Transition Register (TRN)
Compare, Select and Store Unit (CSSU)
Exponent Encoder

The CPU registers are memory-mapped, enabling quick saves and restores. The following are the CPU registers and are used for various purposes.

(i) Interrupt Registers (IMR, IFR)
(ii) Status Registers (ST0 and ST1)
(iii) Processor Mode Status Register (PMST)
(iv) Accumulators (A and B)
(v) Temporary (1) Register
(vi) Transition (TRN) Register
(vii) Auxiliary Registers (AR0 – AR7)
(viii) Stack-Pointer (SP) Register
(ix) Circular-Buffer Size Register (BK)
(x) Block-Repeat Registers (BRC, RSA and REA)
(xi) Program Counter Extension Register (XPC)

Arithmetic Logic Unit (ALU) The 40-bit ALU shown in Fig. 16.7, implements a wide range of arithmetic and logical functions, most of which execute in a single clock cycle. After an operation is performed in the ALU, the result is usually transferred to a destination accumulator (accumulator A or B). The exceptions are the instructions that perform memory-to-memory operations (ADDM, ANDM, ORM and XORM).

Barrel Shifter The barrel shifter is used for scaling operations such as prescaling an input data-memory operand or the accumulator value before an ALU operation; performing a logical or arithmetic shift of the accumulator value; normalising the accumulator; postscaling the accumulator before storing the accumulator value into data memory.

Multiplier / Adder Unit The '54X CPU has three 17-bit hardware multiplier coupled to a 40-bit dedicated adder. This multiplier/adder unit provides multiply and accumulate (MAC) capability in one pipeline phase cycle. The multiplier/adder unit is shown in Fig. 16.8.

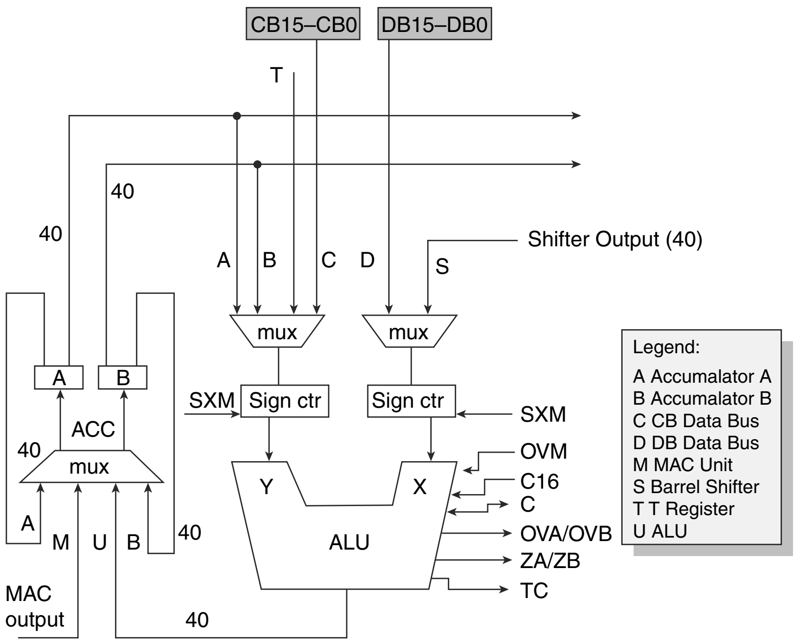


Fig. 16.7 *'C54X Arithmetic Logic Unit Functional Diagram*

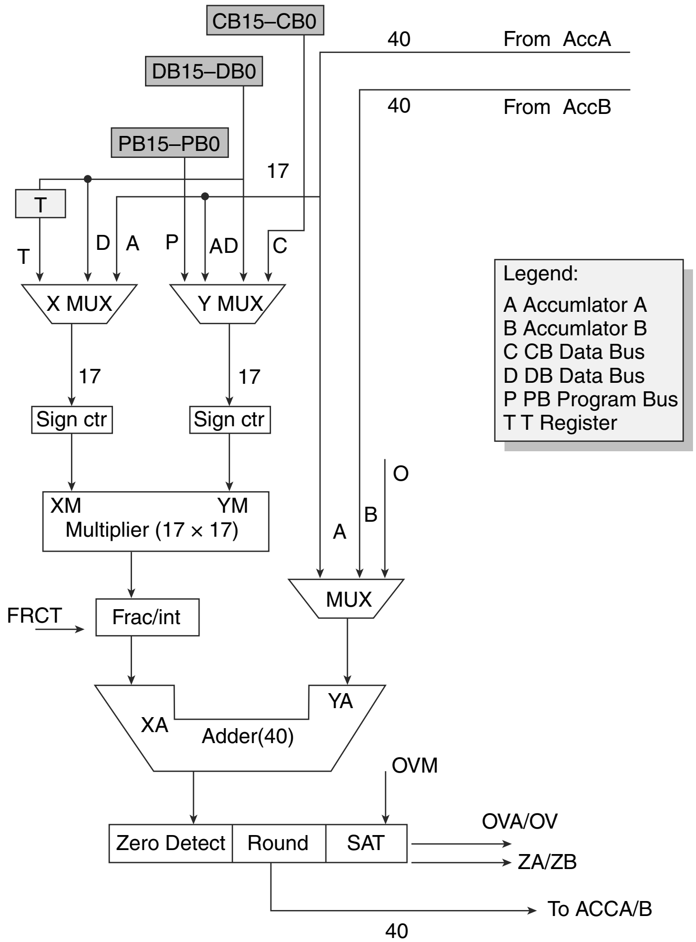


Fig. 16.8 *Multiplier and Adder Functional Diagram*

Compare, Select and Store Unit (CSSU) The compare, select and store unit (CSSU) is shown in Fig. 16.9.

The compare, select and store unit (CSSU) is an application-specific hardware unit dedicated to add/compare/select (ACS) operations of the Viterbi operator. Fig. 16.10 shows the CSSU, which is used with the ALU to perform fast ACS operations. The CSSU allows the '54X to support various Viterbi butterfly algorithms that are used in equalisers and channel decoders.

16.12 SIXTH-GENERATION TMS320C6X PROCESSOR

The TMS320C6X processor is built around Texas Instruments (TIs) VelociTI Very Long Instruction Word (VLIW) architecture.

16.12.1 VLIW Architecture

TMS320C6X, another architecture used for P-DSPs is the very long instruction word (VLIW) architecture. These P-DSPs have a number of processing units (data paths). In other words they have a number

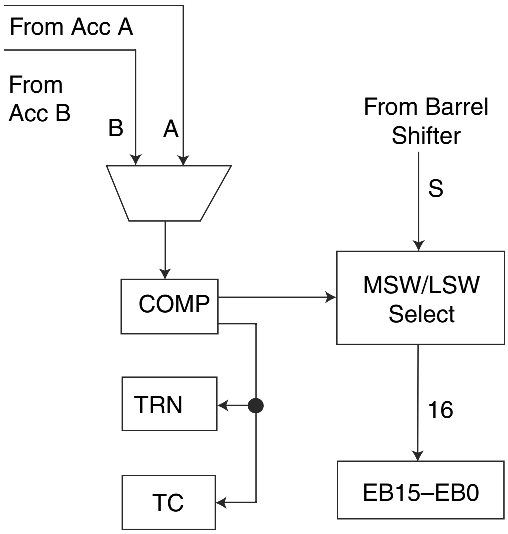


Fig. 16.9 *Compare, Select and Store Unit (CSSU)*

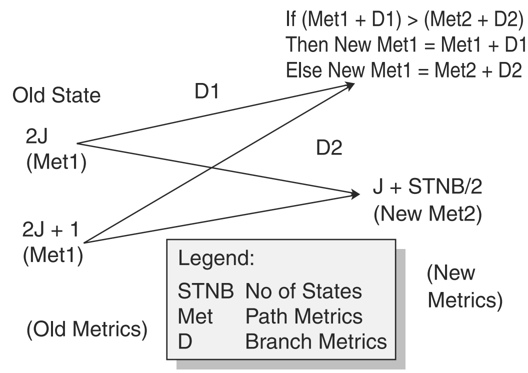


Fig. 16.10 *Viterbi Operator*

of ALUs, MAC Units, Shifters, etc. Parallel random access by the functional units to the register file is facilitated by the read/write cross bar.

The performance gains that can be achieved with VLIW architecture depends on the degree of parallelism in the algorithm selected for a DSP application and the number of functional units. The throughput will be higher only if the algorithm involves execution of independent operations.

However, it may not always be possible to have independent stream of data for processing. Further the number of functional units is also limited by the hardware cost for the multiported register file and cross bar switch.

VelociTIs advanced features include

(i) Instruction packing: reduced code size
(ii) All instructions can operate conditionally: flexibility of code
(iii) Variability-width instructions: flexibility of data types

The TMS320C6X processor operates on a 256-bit (very large) instruction, which is a combination of eight 32-bit instructions per cycle, over two data paths. The TMS320C6X processor is created with 0.18μ CMOS technology, it achieves 6000 MIPS in TIs testing, at speeds up to 1G flop. The TMS320C6X processor CPU consists of 64 general purpose 32-bit register and eight functional units. These eight functional units contain

(i) Two multipliers
(ii) Six ALUs

Features of C6000 Devices

(i) Each multiplier can perform two 16×16-bit or four 8×8-bit multiplies every clock cycle. Each multiplier can perform 32×32-bit multiplies
(ii) The CPU executes up to eight instructions per cycle for up to ten times the performance of typical processors
(iii) It allows a designer to develop highly effective RISC-like code for fast development time

(iv) It gives code size equivalence for eight instructions executed serially or in parallel. It reduces code size, program fetches and power consumption
(v) Special communication-specific instructions have been added to address common operations in error-correcting codes
(vi) Bit count and rotate hardware extends support for bit-level algorithms

The C6000 generation has a complete set of optimised development tools, including an efficient C compiler, an assembly optimiser for simplified assembly language programming and scheduling, and a window based debugger interface for visibility into source code execution characteristics.

The major applications of the TMS320C6X processor are
(i) The design of the embedded system
(ii) Real-time image processing and virtual reality 3 D graphics
(iii) Speech recognition system
(iv) Atmospheric modeling and finite element analysis

16.12.2 TMS320C6X Processor

The TMS320C6X processor is the family member of fixed-point device platform. The block diagram of TMS320C6X processor is given in Fig. 16.11.

The program fetch, instruction dispatch, and instruction decode units can deliver up to eight 32-bit instructions to the functional units in every CPU clock cycle. The processing of instructions occurs in each of the two data paths (A and B), each of which contains four functional units (.L, .S, .M and .D) and thirty two, 32-bit general-purpose registers.

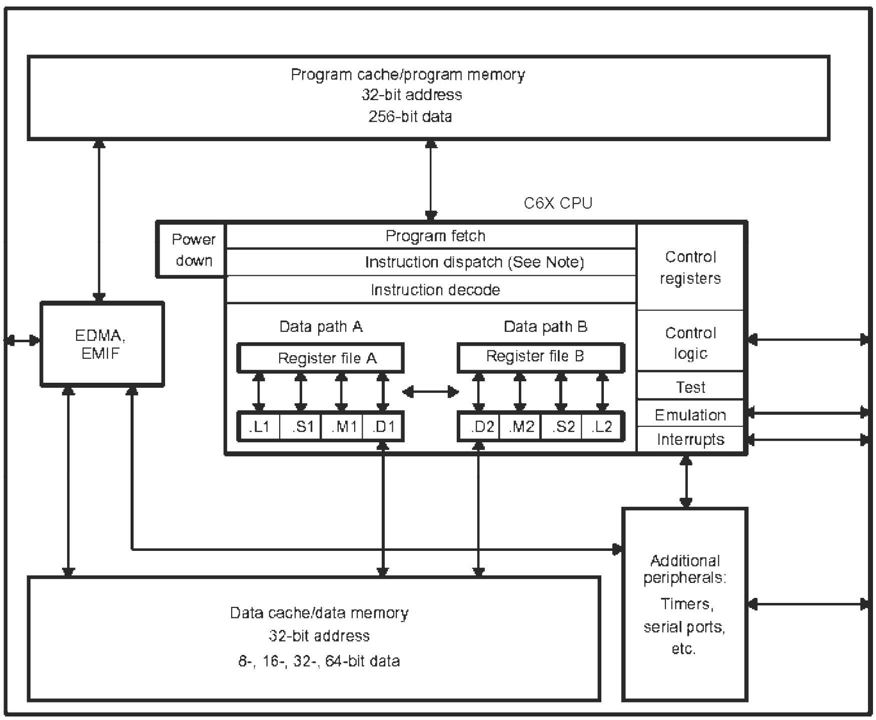


Fig. 16.11 *Architecture of TMS320C6X Processor*

The C64X processors have a 32-bit, byte-addressable address space. On-chip memory is organised in separate data and program spaces. The C64X processor has two 64-bit internal data memory.

A variety of memory and peripheral options are available as follows:

(i) Large on-chip RAM (up to 7 Mbits)
(ii) Two level program cache
(iii) 32-bit external memory interface supports SDRAM, SBSRAM, SRAM and other asynchronous memories
(iv) Enhanced DMA controller
(v) The Ethernet Media Access Controller (EMAC) and physical layer (PHY) device management data I/O (MDIO) module interfaces to the processor through a custom interface for efficient data transmission and reception
(vi) Host processor can directly access the CPU memory space through the host port interface (HPI)
(vii) The multichannel audio serial port (McASP) for multichannel audio applications
(viii) The multichannel buffered serial port (McBSP)
(ix) The peripheral component inter-connect (PCI) port supports connections of C64X processor to a PCI host

16.12.3 Central Processing Unit and Data Paths

The CPU of TMS320C6X consists of eight functional units (.L1,.L2,.S1,.S2,.M1,.M2,.D1 and .D2) in which two are multipliers and the remaining six are ALUs. There are two data paths for the CPU. The eight functional units in the data path are divided into two groups of four each. Each functional unit in one data path is identical to the corresponding unit in the other data path. There are two general-purpose register files (A and B), one for each data path. Each of these files contains sixteen 32-bit registers (A_0–A_{15} for file A and B_0–B_{15} for file B). These register files can be used for data, data address pointers, or condition registers. The TMS320C6X register files support data size ranging from packed 16-bit data to 40-bit fixed point.

Each functional unit present in the processor reads directly from and writes directly to the register files within its own data path, i.e. the .L1,.S1,.M1 and .D1 units write to the register file A and the .L2,.S2,.M2 and .D2 units write to the register file B. There are two register-file data-cross paths (1X and 2X), through which the register files are connected to opposite-side registers. These cross paths allow functional units from one data path to access a 32-bit operand from the opposite-side register file. Out of eight units, only six units in TMS320C6X can access the register file in the opposite-side (M,S and L units, not D units), via a cross path. The M units and S units can access only the src2 operand through cross path, whereas in L units, both src1 and scr2 operands can be accessed through the cross path.

The TMS320C6X has two 32-bit paths, LD1 for register file A and LD2 for register file B, for loading data from memory to the register file, and two 32-bit paths, ST1 and ST2, for storing register values to memory from each register file. There are two data-address paths DA1 and DA2, each of them connected to .D unit in both data paths. This allows data address generated by any one path to access to or from any register. Figure 16.12 shows the CPU unit of the TMS320C6X processor.

16.12.4 Functional Units and their Operations

The .L unit can perform 32/40-bit arithmetic, logic and compare operations. It is also capable of performing arithmetic operations on 8/16-bit operands. The S units can perform 32-bit arithmetic, logic

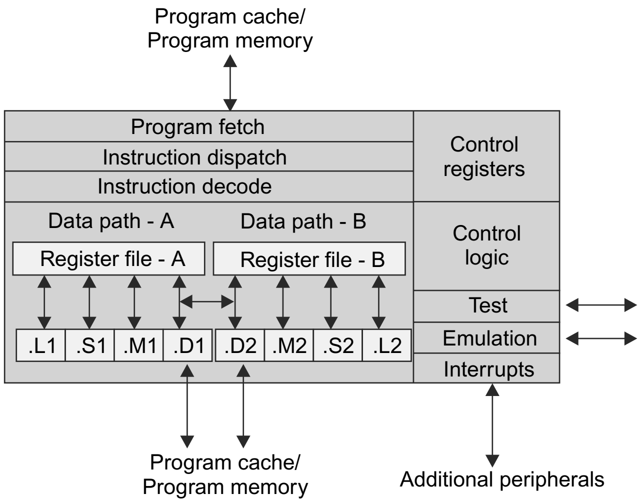


Fig. 16.12 *CPU unit of TMS320C6X processor*

and shift operations. The M units are specially meant to multiply and rotate operations. The .D units are used for load, store and address-generation operations. The operations performed by the functional units are given in Table 16.11 and 16.12. The L unit and .S unit are capable of doing data packing, unpacking and byte-shift operations. Only the .S2 unit can access the control register file.

16.12.5 Addressing Modes in TMS320C6X

The TMS320C6X family processors support linear and circular addressing mode. In the linear addressing mode, the address of the operand is specified in any one of the registers in the register file. The operands are shifted by the offset value given in the offset field and access of byte, half word or word is performed. As far as the circular addressing is concerned, there are two registers, BK0 and BK1, which specify the block size for the circular addressing.

16.12.6 Memory Architecture

The TMS320C6X processor provides two-level memory architecture for the internal program and data buses. The first-level memory for both the internal program and data bus is 4K byte cache, designated L1P for the program cache and L1D for the data cache. The second-level memory is a 64K byte memory block that is shared by the program memory bus and data memory bus, denoted as L2. The bus connections between internal memories and the CPU are shown in Fig.16.14. These cache spaces are not included in the memory map and are enabled at all times. The CPU can access level one caches. The program cache controller provides interface between the CPU and L1P cache. The CPU provides a 256-bit wide path to permit a continuous stream of eight 32-bit instructions for maximum performance. The 4K L1P cache is organized as a 64-line direct mapped cache with a 64-byte line size.

The data cache controller provides interface between the CPU and L1D cache. The L1D is a dual ported memory. This allows simultaneous access to both sides of the CPU. The L1D, 4K cache is organized as a 64-set 2-way set associative cache with a 32-byte line size. Initially, the L1P and L1D caches are accessed, on a miss to either L1D or L1P, and the request is passed to L2 controller. The L2

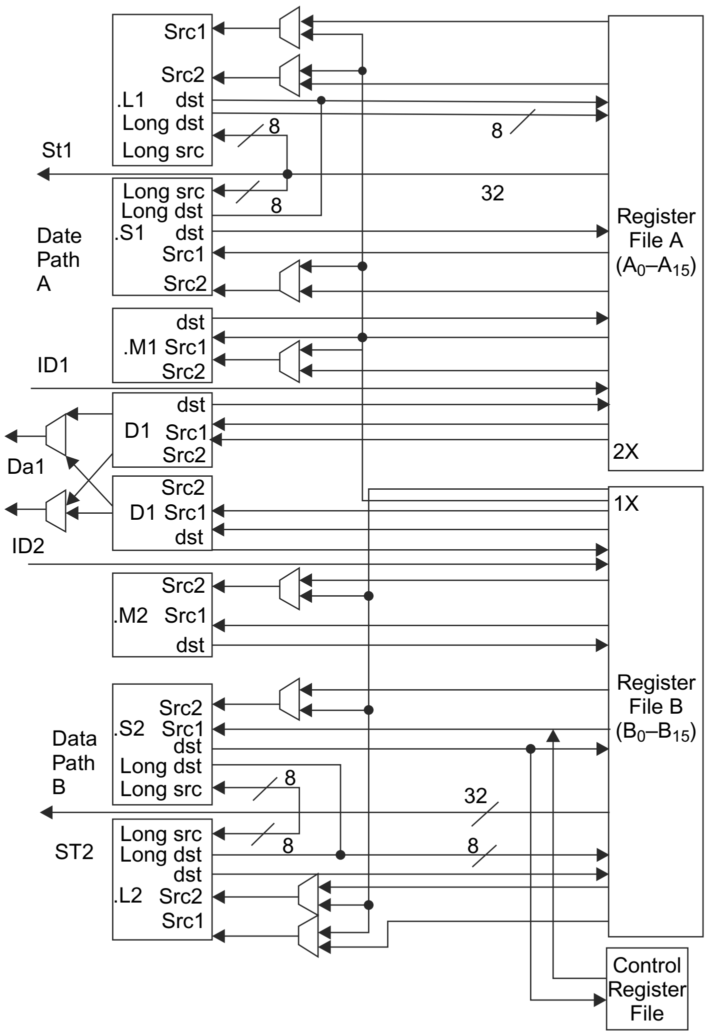


Fig. 16.13 *TMS320C6X CPU data paths*

Table 16.11 *Functional units of TMS320C6X processor*

Name of the Unit / *Type of operation*	*.L unit*	*.S unit*	*.M unit*	*.D unit*
Arithmetic operation	32/40 bit operation Dual 16-bit, Quad 8-bit arithmetic and min/max operations*	32-bit operation Dual 16-bit, Quad 8-bit saturated arithmetic operations*	–	32-bit add and subtract operations only
Logical operation	32-bit operations	32-bit operations	–	32-bit logical operations

Contd.

Name of the Unit / *Type of operation*	*.L unit*	*.S unit*	*.M unit*	*.D unit*
Multiply operations	–	–	16 × 16 multiply operations 16 × 32, Quad 8 × 8, Dual 16 × 16 multiply operations	–
Shift operations	Byte shifts*	Byte shifts, Dual 16-bit shift operation*		
Compare operations	32/40 bit operations	Dual 16-bit, Quad 8-bit compare operations*	–	–
Branch operations	–	Yes	–	–
Load and Store operations	–	–	–	Load and stores with 5-bit constant offset (15-bit constant offset in .D2 only)
Linear and circular address calculation	–	–	–	Yes
Constant generation	5-bit constant generation*	Yes	–	5-bit constant generation*
Count operations	32/40-bit count operations	–	–	–
Move operations	Register to register only 16-bit move operations*	16-bit move operations	Register to register*	Register to register 16-bit move operations*

* Additional operations performed by the functional units in C64X processor.

Table 16.12 *Functional units and operations*

Functional Unit	*Operations*
.L Unit (.L1 and .L2)	32/40-bit arithmetic and compare operations, left 1 or 0 bit counting for 32 bits, Normalization count for 32 and 40 bits, 32-bit logical operations, byte shifts, data packing/unpacking, 5-bit constant generation
.S Unit (.S1 and .S2)	32-bit arithmetic operations, 32/40-bit shifts and 32-bit bit-field operations, 32-bit logical operations, branches,constant generation, register transfer to/from control registers (.S2 only), byte shifts, data packing/unpacking
.M Unit (.M1 and .M2)	16 × 16 multiply operations, bit expansion, bit interleaving/deinterleaving, variables shift operations, rotation
.D Unit (.D1 and .D2)	32-bit add, subtract, linear and circular address calculation, load and stores operations, 5-bit constant generation, 32-bit logical operations

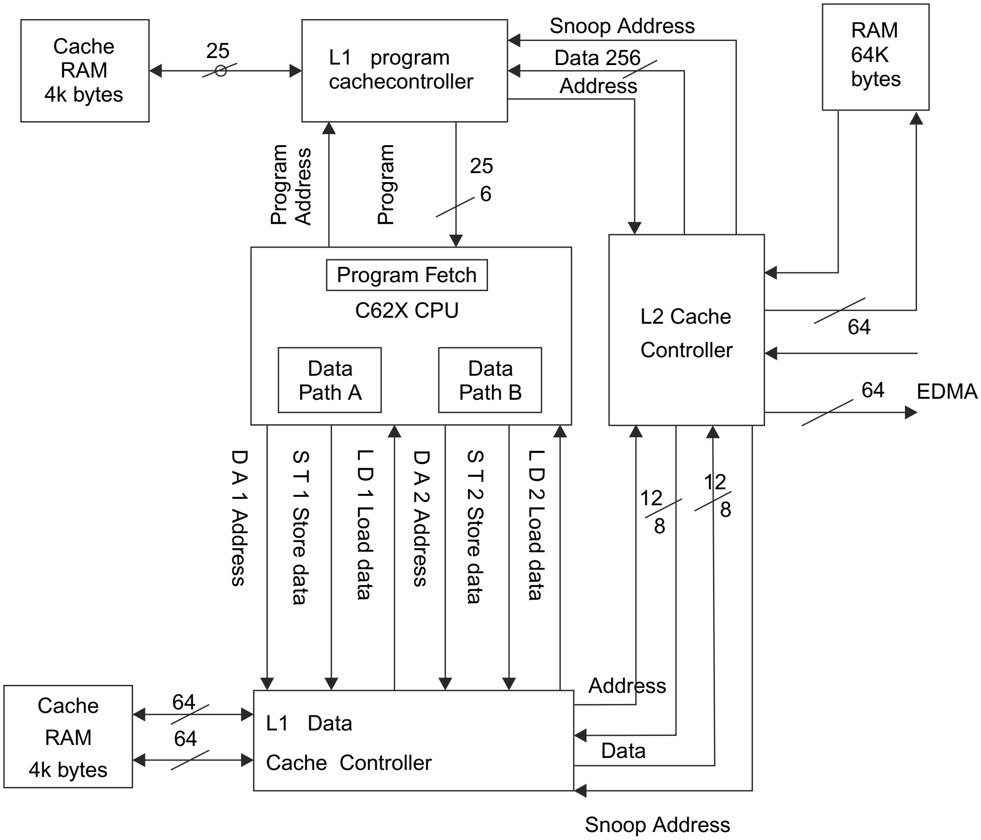


Fig 16.14 *Internal memory block diagram of TMS320C6X*

controller provides the following:

(i) Necessary arbitration is provided by the CPU and the Enhanced Direct Memory Access (EDMA) controller to access the internal memory
(ii) The CPU data access to the EMIF
(iii) The CPU access to on-chip peripheral
(iv) Sends a request to EMIF for an L2 data miss

On request to L2 service, the services depend on the operating mode of L2, which is set in the Cache Configuration Register Fields (CCFG). This is a memory-mapped register, whose memory-map address is 0184 0000h. The format of the CCFG is shown in Fig.16.15 and the various L2 modes are shown in Table 16.13. The L2 memories are organized as four 64-bit-wide banks.

313010987320

P	Reserved	IP	ID	Reserved	L2MODE

RW, + 0 R, +× W, + 0 W, + 0 R + 0 0000 RW + 000

Fig 16.15 *Format of cache configuration register (CCFG)*

16.12.7 External Memory Access

When L2 data is missed, the L2 controller sends a request to EMIF. The Memory Attribute Register (MAR) can be programmed to turn on caching of each of the external Chip Enable (CE) spaces. Thus,

Table 16.13 *Cache configuration register field description*

Functional Unit	Operations
L2 MODE	L2 Operation modes 000b-64K bytes RAM 001b-16K bytes 1-way cache/48K bytes mapped RAM 010b-32K bytes 2-way cache/32K bytes mapped RAM 011b-48K bytes 3-way cache/16K bytes mapped RAM 111b-64K bytes 4-way cache
ID	Invalidate L1D:ID = 0-normal L1D operation ID = 1-A || L1D lines invalidated
IP	Invalidate L1P:IP = 0-normal L1P operation IP = 1-A || L1P lines invalidated
P	L2 Requestor Priority :P = 0, CPU accesses prioritized over enhanced DMA access P = 1.Ehanced DMA accesses prioritized over CPU accesses

single-word read operations to external mapped devices are performed. Without this feature, any external read would always read an entire L2 line of data. The four CE spaces are divided into four ranges, each of which maps the least significant bit of the MAR register.L2 caches have a corresponding address range if MAR register is set.Upon reset, MAR registers are set to 0. To begin caching data in the L2, it is necessary to set the corresponding MAR register. The MAR defines the cacheability for the EMIF. Table 16.14 shows the various CE spaces and the corresponding MAR registers.

All the memory space-base address registers, word count registers and the 15 memory attribute register are memory-mapped registers starting from the location 0184 0000h to 1084 82CCh. Before the memory access, corresponding registers have to be initialized.

Table 16.14 *The CE space address and its MAR registers*

MAR	*Address Range Enabled*	*CE Space*
15	B300 0000h-B3FF FFFFh	CE3
14	B300 0000h-B2FF FFFFh	CE3
13	B100 0000h-B1FF FFFFh	CE3
12	B000 0000h-B0FF FFFFh	CE3
11	A300 0000h-A3FF FFFFh	CE2
10	A200 0000h-A2FF FFFFh	CE2
9	A100 0000h-A1FF FFFFh	CE2
8	A000 0000h-A0FF FFFFh	CE2
7	9300 0000h-93FF FFFFh	CE1
6	9200 0000h-92FF FFFFh	CE1
5	9100 0000h-91FF FFFFh	CE1
4	9000 0000h-90FF FFFFh	CE1
3	8300 0000h-83FF FFFFh	CE0
2	8200 0000h-82FF FFFFh	CE0
1	8100 0000h-81FF FFFFh	CE0
0	8000 0000h-80FF FFFFh	CE0

16.12.8 Pipelining in the TMS320C6X Processor

The pipelining is used to provide flexibility in programming and also to improve the performance of the processor. The control of the pipeline is simplified by eliminating pipeline interlocks. The bottlenecks in program fetch, data access, and multiply operations can be eliminated by increased pipelining. The single-cycle throughput is achieved by pipelining. There are three stages of pipeline phase and they are (i) Fetch, (ii) Decode, and (iii) Execute.

All the instructions require the same number of pipeline phase for fetch and decode, but require a varying number of execution phases. The fetch stage of the pipeline has four phases for all the instructions as given below.

(i) Program address Generate (PG)
(ii) Program address Send (PS)
(iii) Program access ready Wait (PW)
(iv) Program fetch packet Receive (PR)

The fetch packet of the processor uses eight instructions and all the eight instructions proceed through fetch processing together with the PG, PS, PW and PR phases.

The decode stage of the pipeline has two phases. They are instructions dispatch (DP) and instructions decode (DC). In the DP phase of the pipeline, the fetch packets are split into execute packets and in the DC phase, the execution of the instructions in the functional units is done by decoding the source registers and destination registers, and associated data paths.

The execute stage of the pipeline is subdivided into five phases (E1-E5). Different types of instructions require different numbers of these phases to compute their execution.

16.12.9 Peripherals

The user accessible peripherals of TMS320C6X processor are listed below.

(i) Enhanced Direct Memory Access (EDMA) controller
(ii) Host-port Interface (HPI)
(iii) External Memory Interface (EMIF)
(iv) Boot configuration
(v) Two multichannel buffered serial ports (McBSP's)
(vi) Interrupt selector
(vii) Two 32-bit timers
(viii) Power down logic

These peripherals are configured through a set of memory-mapped control registers. The arbitration for accesses of on-chip peripherals is performed by the peripheral bus controller. The boot configuration is interfaced through the external signals only, and the power-down logic is accessed directly by the CPU. Figure 16.16 shows the peripherals in the block diagram for the TMS320C6X processor and it is described below.

EDMA Controller The data between address ranges in the memory gap is transferred by the EDMA controller without intervention by the CPU. The EDMA controller consists of 16 programmable channels and a RAM space to hold multiple configurations for future transfers.

Host Port Interface The HPI is a parallel port by which the CPU's memory space can be accessed-directly by a host processor. The host device has ease of access because it is the master of the interface. The host and the CPU can exchange information via internal memory. The memory-mapped peripherals can also be accessed directly by the host interface.

External Memory Interface The EMIF supports a glueless interface to several external devices such as synchronous burst SRAM (SBSRAM), synchronous DRAM (SDRAM), asynchronous devices, external shared memory device, etc.

Boot Configuration TMS320C6X provides a variety of boot configurations to determine the performance of DSP actions after device reset to prepare for initialization. These actions include loading in code from an external ROM space on the EMIF and loading code through the HPI/expansion bus from an external host.

McBSP The multichannel buffered serial port is based on the standard serial port interface. In addition, the port can also buffer serial samples in memory automatically with the aid of the DMA/EDMA controller. It also has multi channel capability compatible with various networking standards.

Interrupt Selector The TMS320C6X peripheral set produces 16 interrupt sources and the CPU has 12 interrupts available. The RESET and NMI are the non-maskable interrupts whereas the CPU interrupts are maskable. The Global Interrupt Enable bit (GIE) in the Control Status Register (CSR) is set to 1 in order to mask the interrupts. The respective bit in the Interrupt Enable (IE) register is set to 1 in order to enable an interrupt. When the corresponding interrupt occurs, the bit in the Interrupt Flag Register (IFR) is set. The interrupt selector allows the user to choose among the 12 interrupts based on system need and to change the polarity of external interrupt inputs.

Timer There are two 32-bit general-purpose timers, Timer 1 and Timer 0 as shown in Fig. 16.16. They are used to time events, count events, general pulses, interrupt the CPU and to send synchronization events to the DMA/EDMA controller.

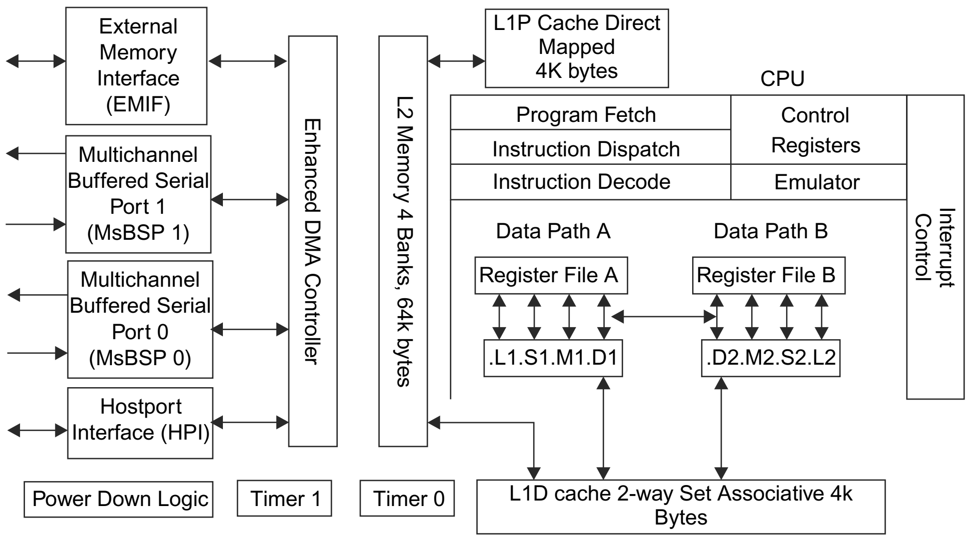


Fig 16.16 *Block diagram of C6X with peripherals*

Power-down Power saving is done through power-down logic and it allows reduced clocking to reduce power consumption. Most of the operating power of CMOS logic dissipates during circuit switching from one logic state to another. Significant power savings can be obtained without losing any data or operational context by preventing some or all of the chip's logic from switching.

ADSP PROCESSORS 16.13

The ADSP 21xx is the first digital signal processor from Analog Devices. The family consists of a large number of processors based on a common 16-bit fixed-point architecture with a 24-bit instruction word. The ADSP 2184 is a single-chip microcomputer optimized for Digital Signal Processing (DSP) and other high-speed numeric processing applications. The major application of ADSP 21xx processors are modem, audio processing, PC multimedia and digital cellular applications.

The Analog Devices ADSP 210x and ADSP 219x are 16-bit fixed-point processors with 24-bit instructions. The ADSP 219x is based on the ADSP 218x architecture andthe enhancements include the addition of new addressing modes, an expanded address space, addition of an instruction cache, and deeper pipeline (six stages, compared to three on the ADSP 218x) to enable faster clock speeds.

The ADSP 2184 combines the ADSP 2100 family base architecture (three computational units, data-address generators and a program sequencer) with two serial ports, a 16-bit internal DMA port, a byte DMA port, a programmable timer, Flag I/O, extensive interrupt capabilities and on-chip program and data memory.

The ADSP 2184 integrates 20K bytes of on-chip memory configured as 4K words (24-bit) of program RAM and 4K words (16-bit) of data RAM. Power-down circuitry is also provided to meet the low power needs of battery-operated portable equipment. The ADSP 2184 is available in a 100-lead LQFP package.

In addition, the ADSP 2184 supports instructions that include bit manipulations—bit set, bit clear, bit toggle, bit test—ALU constants, multiplication instruction (x squared), biased rounding, result-free ALU operations, I/O memory transfers, and global interrupt masking for increased flexibility. Fabricated in a high-speed, double-metal, low-power CMOS process, the ADSP 2184 operates with a 25 ns instruction cycle time. Every instruction can execute in a single processor cycle. Figure 16.17 shows the functional block diagram of the ADSP 2184 processor.

The ADSP 21xx family of DSPs contain a shadow bank register that is useful for single-cycle context switching of the processor. The ADSP 219x are used in low-cost applications, particularly on control applications. There are three members in the ADSP 219x family.

The ADSP 2184 flexible architecture and comprehensive instruction set allow the processor to perform multiple operations in parallel. In one cycle, the ADSP 2184 can

(i) generate the next program address,
(ii) fetch the next instruction,
(iii) perform one or two data moves,
(iv) update one or two data address pointers, and
(v) perform a computational operation.

The above operations take place when the processor continues to

(i) receive and transmit data through the two serial ports,
(ii) receive or transmit data through the internal DMA port,
(iii) receive or transmit data through the byte DMA port, and
(iv) decrement the timer.

16.13.1 Architecture Overview

The ADSP 2184 instruction set provides flexible data moves and multifunction (one or two data moves with a computation) instructions. Every instruction can be executed in a single processor cycle. The ADSP 2184 assembly language uses an algebraic syntax for ease of coding and readability. A comprehensive set of development tools supports program development.

The architectural block diagram of the ADSP 2I84 is shown in Fig. 16.17. The processor contains three independent functional units, namely ALU, multiplier/accumulator (MAC) and shifter. The functional units process 16-bit data directly and have provisions to support multiprecision computations. The ALU performs a standard set of arithmetic and logic operations; division primitives are also supported. The MAC performs single-cycle multiply, multiply/add and multiply/subtract operations with 40 bits of accumulation. The shifter performs logical and arithmetic shifts, normalization, denormalization and derives exponent operations.

The shifter can be used to efficiently implement numeric format control including multi word and block floating-point representations. The internal result (R) bus connects the functional units so the output of any unit may be the input of any unit on the next cycle.

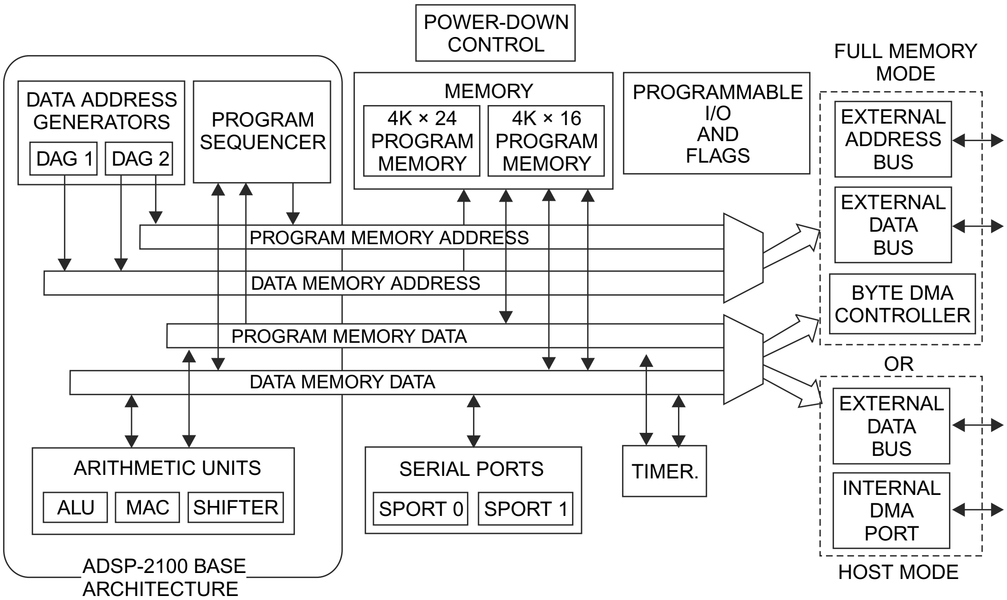


Fig. 16.17 *Functional block diagram of the ADSP 2184 processor*

A powerful program sequencer and two dedicated data-address generators ensure efficient delivery of operands to these functional units. The sequencer supports conditional jumps, sub-routine calls and returns in a single cycle. With internal loop counters and loop stacks, the ADSP 2184 executes looped code with zero overhead; no explicit jump instructions are required to maintain loops.

Two Data Address Generators (DAGs) provide addresses for simultaneous dual operand fetches from data memory and program memory. Each DAG maintains and updates four address pointers. Whenever the pointer is used to access data (indirect addressing), it is post-modified by the value of one of four possible modify registers. A length value may be associated with each pointer to implement automatic modulo addressing for circular buffers.

Efficient data transfer is achieved by using the following five internal buses

(i) Program Memory Address (PMA) Bus
(ii) Program Memory Data (PMD) Bus
(iii) Data Memory Address (DMA) Bus
(iv) Data Memory Data (DMD) Bus
(v) Result (R) Bus

The two address buses (PMA and DMA) share a single external address bus, allowing memory to be expanded off-chip, and the two data buses (PMD and DMD) share a single external data bus. Byte memory space and I/O memory space also share the external buses.

Program memory can store both instructions and data, allowing the ADSP 2184 to fetch two operands in a single cycle, one from program memory and the other from data memory. The ADSP2184 can fetch an operand from program memory and the next instruction in the same cycle.

When configured in host mode, the ADSP 2184 has a 16-bit Internal DMA port (IDMA port) for connection to external systems. The IDMA port is made up of 16 data/address pins and five control pins. The IDMA port provides transparent, direct access to the DSPs on-chip program and data RAM.

An interface to low-cost byte-wide memory is provided by the Byte DMA port (BDMA port). The BDMA port is bidirectional and can directly address up to four megabytes of external RAM or ROM for off-chip storage of program overlays or data tables.

The byte memory and I/O memory space interface supports slow memories and I/O memory-mapped peripherals with programmable wait state generation. External devices can gain control of external buses with bus request/grant signals ($\overline{BR}$, $\overline{BGH}$ and $\overline{BG}$). One execution mode (Go Mode) allows the ADSP 2184 to continue running from on-chip memory. Normal execution mode requires the processor to halt while buses are granted.

The ADSP 2184 can respond to eleven interrupts. There are up to six external interrupts (one edge-sensitive, two level-sensitive and three configurable) and seven internal interrupts generated by the timer, the serial ports (SPORTs), the Byte DMA port and the power-down circuitry. There is also a master $\overline{RESET}$ signal. The two serial ports provide a complete synchronous serial interface with optional companding in hardware and a wide variety of framed or frameless data transmit and receive modes of operation.

Each port can generate an internal programmable serial clock or accept an external serial clock. The ADSP 2184 provides up to 13 general-purpose flag pins. The data input and output pins on SPORT1 can be alternatively configured as an input flag and an output flag. In addition, eight flags are programmable as inputs or outputs, and three flags are always outputs.

A programmable interval timer generates periodic interrupts. A 16-bit count register (TCOUNT) decrements every n processor cycle, where n is a scaling value stored in an 8-bit register (TSCALE). When the value of the count register reaches zero, an interrupt is generated and the count register is reloaded from a 16-bit period register (TPERIOD).

16.13.2 Serial Ports

The ADSP 2184 incorporates two complete synchronous serial ports (SPORT0 and SPORT1) for serial communications and multiprocessor communication.

(i) SPORTs are bidirectional and have a separate, double-buffered transmit and receive section.
(ii) SPORTs can use an external serial clock or generate their own serial clock internally.
(iii) SPORTs have independent framing for the receive and transmit sections. Sections run in a frameless mode or with frame synchronization signals internally or externally generated. Frame sync signals are active high or inverted, with either of two pulse widths and timings.
(iv) SPORTs support serial data-word lengths from 3 to 16 bits and provide optional A-law (European) and μ-law (US and Japan) companding according to CCITT recommendation G.711.
(v) SPORT receive and transmit sections can generate unique interrupts on completing a data-word transfer.
(vi) SPORTs can receive and transmit an entire circular buffer of data with only one overhead cycle per data word. An interrupt is generated after a data buffer transfer.
(vii) SPORT0 has a multichannel interface to selectively receive and transmit a 24- or 32-word, time-division multiplexed, serial bitstream.
(viii) SPORT1 can be configured to have two external interrupts ($\overline{IRQ0}$ and $\overline{IRQ1}$) and the Flag-In and Flag-Out signals. The internally generated serial clock may still be used in this configuration.

16.13.3 PIN Descriptions

The ADSP 2184 is a l00-lead LQFP package. Some serial ports, program flags, interrupt and external bus pins have dual multiplexed functionality. Hence, there is a reduction in package size and pin count, and improvement on performance. The external bus pins are configured during RESET, while serial port pins are software configurable during program execution. Flag and interrupt functionality is retained concurrently on multiplexed pins. In cases where pin functionality is reconfigurable, the default state is shown in plain text; alternate functionality is shown in italics.Table 16.15 shows the common mode pin details of 2184 processor.

Table 16.15 *Common mode pins*

Pin Name(s)	*# of Pins*	*Input/Output*	*Function*
$\overline{\text{RESET}}$	1	I	Processor Reset Input
$\overline{\text{BR}}$	1	I	Bus Request Input
$\overline{\text{BG}}$	1	O	Bus Grant Output
$\overline{\text{BGH}}$	1	O	Bus Grant Hung Output
$\overline{\text{DMS}}$	1	O	Data Memory Select Output
$\overline{\text{PMS}}$	1	O	Program Memory Select Output
$\overline{\text{IOMS}}$	1	O	I/O Memory Select Output
$\overline{\text{BMS}}$	1	O	Byte Memory select Output
$\overline{\text{CMS}}$	1	O	Combined Memory Select output
$\overline{\text{RD}}$	1	O	Memory Read Enabled output
$\overline{WR}$	1	O	Memory Write Enabled output
$\overline{WR}$ IRQ@/	1	I	Edge- or Level-Sensitive Interrupt Request
PF7		I/O	Programmable I/O Pin
$\overline{IRQL}$0/	1	I	Level Sensitive Interrupt Requests*
PF5		I/O	Programmable I/O pin
IRQL1/	1	I	Level-Sensitive Interrupt Requests*
PF6		I/O	Programmable I/O Pin
$\overline{IRQE}$/	1	I	Edge-Sensitive Interrupt Requests*
PF4		I/O	Programmable I/O Pin
PF3	1	I/O	Programmable I/O Pin
Mode C/	1	I	Mode Select Input—Checked only during RESET
PF2		I/O	Programmable I/O Pin During normal operation
Mode B/	1	I	Mode Select Input–Checked only during RESET
PF1		I/O	Programmable I/O Pin during normal Operation
CLKIN, xtal	2	I	Clock or Quartz Crystal Input
CLKOUT	1	O	Processor Clock Output
SPORT 0	5	I/O	Serial Port I/O Pins
SPORT1/	5	I/O	Serial Port I/O Pins
$\overline{IRQ}$1:$\overline{0}$			Edge-or Level Sensitive Interrupts
F1,FO			Flag-In, Flag-Out**
$\overline{PWD}$ PWD	1	I	Power-Down Control Input
PWDACK	1	O	Power Down Control Output
Flo, FL1, FL2,	3	O	Output Flags
VDD and GND	16	I	Power and Ground
EZ-Port	9	I/O	For Emulation Use

Note: * Interrupt/Flag pins retain both functions concurrently. If IMASK is set to enable the corresponding interrupts, the DSP will vector to the appropriate interrupt vector address when the pin is asserted either by external devices or set as a programmable flag.

** SPORT configuration determined by the DSP system Control register Software configurable.

Memory Interface Pins

The ADSP2184 processor can be operatedeither in Full Memory Mode, which allows BDMA operation with full external overlay memory or in I/O capability (Host Mode)that allows IDMA operation with limited external addressing capabilities. The operating mode is determined by the state of the Mode C pin during RESET and cannot be changed while the processor is running. Table 16.16 shows the details of the full memory mode pins and Table 16.17 shows the host mode pins.

In host mode, external peripheral addresses can be decoded using the A0 $\overline{BMS}$, $\overline{CMS}$, $\overline{PMS}$, $\overline{DMS}$ and $\overline{IOMS}$ signals.

Table 16.16 *Full memory mode pins (Mode C = 0)*

Pin Name	*# of Pins*	*Input/Output*	*Function*
A13:0	14	O	Address Output Pins for Program, Data, Byte and I/O Spaces
D23:0	24	I/O	Data I/O Pins for Program, Data Byte and I/O Spaces (8MSBs Are Also Used as Byte Memory Addresses)

Table 16.17 *Host mode pins (Mode C = 1)*

Pin Name	*# of Pins*	*Input/Output*	*Function*
IAD 15:0	16	I/O	IDMA port Address/Data Bus
A0	1	O	Address Pin for External I/O, Program, Data or Byte Access
D23:8	16	I/O	Data I/O Pins for Program, Data Byte and I/O Spaces
IWR	1	1	IDMA Write Enable
IRD	1	I	IDMA Read Enable
IAL	1	I	IDMA Address Latch Pin
IS	1	I	IDMA Select
IACK	1	O	IDMA Port Acknowledge

16.13.4 Interrupts

The interrupt controller handles eleven possible interrupts and reset with minimum overhead. The ADSP 2184 supports four dedicated external interrupt input pins, $\overline{IRQ2}$, $\overline{IRQL0}$, $\overline{IRQL1}$ and $\overline{IRQE}$ (shared with the PF7:4 pins). In addition, SPORT1 may be reconfigured for $\overline{IRQ0}$, $\overline{IRQ1}$, FLAG_IN and FLAG_OUT, for a total of six external interrupts. The ADSP′-2184 also supports internal interrupts from the timer, the byte DMA port, the two serial ports, software and the power-down control circuit. The interrupt levels are internally prioritized and individually maskable(except power-down and $\overline{RESET}$). The $\overline{IRQ2}$, $\overline{IRQ0}$ and $\overline{IRQ1}$ input pins can be programmed to be either level- or edge-sensitive. $\overline{IRQL0}$ and $\overline{IRQL1}$ are level-sensitive and IRQE is edge-sensitive. The priorities and vector addresses of all interrupts are shown in Table 16.18.

Table 16.18 *Interrupt Priority & Interrupt Vector Addresses*

Source of Interrupt	***Interrupt Vector Address (Hex)***
Reset (or Power-Up with PUCR = 1)	0000 (Highest Priority)
Power-Down (Nonmaskable)	002C
$\overline{IRQ2}$	0004
$\overline{IRQL1}$	0008
$\overline{IRQL0}$	000C
SPORT0 Transmit	0010
SPORT0 Receive	0014
$\overline{IRQE}$	0018
BDMA Interrupt	001C
SPORT1 Transmit or $\overline{IRQ1}$	0020
SPORT1 Receive or $\overline{IRQ0}$	0024
Timer	0028 (Lowest Priority)

Interrupt routines can either be nested, with higher priority interrupts taking precedence, or processed sequentially. Interrupts can be masked or unmasked with the IMASK register. Individual interrupt requests are logically ANDed with the bits in IMASK; the highest priority unmasked interrupt is then selected. The power-down interrupt is nonmaskable.

The ADSP 2184 masks all interrupts for one instruction cycle following the execution of an instruction that modifies the IMASK register. This does not affect serial port autobuffering or DMA transfers.

The interrupt control register, ICNTL, controls interrupt nesting and defines the $\overline{IRQ0}$, $\overline{IRQ1}$ and $\overline{IRQ2}$ external interrupts to be either edge- or level-sensitive. The $\overline{IRQE}$ pin is an external edge-sensitive interrupt and can be forced and cleared. The $\overline{IRQL0}$ and $\overline{IRQL1}$ pins are external level-sensitive interrupts. The IFC register is a write-only register used to force and clear interrupts. On-chip stacks preserve the processor status and are automatically maintained during interrupt handling. The stacks are twelve levels deep to allow interrupt, loop and subroutine nesting.

The following instructions allow global enable or disable servicing of the interrupts (including power-down), regardless of the state of IMASK. Disabling the interrupts does not affect serial port auto-buffering or DMA.

```
ENA INTS;
DIS INTS;
```

When the processor is reset, interrupt servicing is enabled.

16.13.5 Low-Power Operation

ADSP 2184 has three low-power modes that can reduce the power dissipation when the device operates under standby conditions. These modes are Power-down, Idle and Slow Idle. The CLKOUT pin may also be disabled to reduce external power dissipation.

Power-Down

ADSP 2184 processor has a low power feature that lets the processor enter a very low-power dormant state through hardware or software control. Following is a brief list of power-down features.

(i) Quick recovery from power-down. The processor executes instructions in 200 CLKIN cycles.
(ii) Support for an externally generated TTL or CMOS processor clock. The external clock can continue running during power-down without affecting the lowest power rating and 200 CLKIN cycle recovery.
(iii) Support for crystal operation includes disabling the oscillator to save power (the processor automatically waits approximately 4096 CLKIN cycles for the crystal oscillator to start or stabilize), and letting the oscillator run to allow 200 CLKIN cycle start-up.
(iv) Power-down is initiated by either the power-down pin ($\overline{PWD}$) or the software power-down force bit.
(v) Interrupt support allows an unlimited number of instructions to be executed before optionally powering down. The power-down interrupt also can be used as a nonmaskable, edge-sensitive interrupt.
(vi) Context clear/save control allows the processor to continue where it left off or start with a clean context when leaving the power-down state.
(vii) The $\overline{RESET}$ pin is used to terminate power-down.
(viii) Power-down acknowledge pin indicates when the processor has entered power-down.

Idle

When the ADSP 2184 is in the idle mode, the processor waits indefinitely in a low-power state until an interrupt occurs. When an unmasked interrupt occurs, it is serviced; execution then continues with the instruction following the IDLE instruction. In idle mode, IDMA, BDMA and autobuffer cycle steals still occur.

Slow Idle

The IDLE instruction is enhanced on the ADSP 2184 to let the processor's internal clock signal be slowed, further reducing power consumption. The reduced clock frequency, a programmable fraction of the normal clock rate, is specified by a selectable divisor given in the IDLE instruction. The format of the Instruction is *IDLE (n)*; where n = 16, 32, 64 or 128. This instruction keeps the processor fully functional, but operating at the slower clock rate. While it is in this state, the processor's other internal clock signals, such as SCLK, CLKOUT and timer clock, are reduced by the same ratio. The default form of the instruction, when no clock divisor is given, is the standard IDLE instruction.

When the *IDLE (n)* instruction is used, it effectively slows down the processor's internal clock and thus its response time to incoming interrupts. The one-cycle response time of the standard idle state is increased by n, the clock divisor. When an enabled interrupt is received, the ADSP 2184 will remain in the idle state for up to a maximum of n processor cycles (n = 16, 32, 64 or 128) before resuming normal operation.

When the *IDLE (n)* instruction is used in systems that have an externally generated serial clock (SCLK), the serial clock rate may be faster than the processor's reduced internal clock rate. Under these conditions, interrupts must not be generated at a faster rate than can be serviced, due to the additional time the processor takes to come out of the idle state (a maximum of n processor cycles).

16.13.6 Memory Architecture

The ADSP 2184 provides a variety of memory and peripheral interface options. The main functional units are Program Memory, Data Memory, Byte Memory and I/O.

Program Memory (Full Memory Mode) is a 24-bit-wide space for storing both instruction opcodes and data. The ADSP 2184 has 4K words of Program Memory RAM on chip. It has the capability of accessing up to two 8K external memory overlay spaces using the external data bus. Both an instruction opcode and a data value can be read from on-chip program memory in a single cycle.

Data Memory (Full Memory Mode) is a 16-bit-wide space used for the storage of data variables and for memory-mapped control registers. The ADSP 2184 has 4K words on Data Memory RAM on chip. It can support up to two 8K external memory overlay spaces through the external data bus.

Byte Memory (Full Memory Mode) provides access to an 8-bit-wide memory space through the Byte DMA (BDMA) port. The Byte Memory interface provides access to 4 MBytes of memory by utilizing eight data lines as additional address lines. This gives the BDMA Port an effective 22-bit address range. On power-up, the DSP can automatically load bootstrap code from byte memory.

I/O Space (Full Memory Mode) allows access to 2048 locations of 16-bit-wide data. It is intended to be used to communicate with parallel peripheral devices such as data converters and external registers or latches.

Program Memory

The ADSP 2184 contains 4K × 24 of on-chip program RAM. The on-chip program memory is designed to allow up to two accesses of each cycle so that all operations can complete in a single cycle. In addition, the ADSP 2184 allows the use of 8K external memory overlays.

The program memory space organization is controlled by the Mode B pin and the PMOVLAY registers. Normally, the ADSP 2184 is configured with Mode B = 0 and program memory organized as shown in Table16.19.

When PMOVLAY is set to 1 or 2, external accesses occur at addresses 0x2000 through 0x3FFF. The external address is generated as shown in Table 16.20.

Table 16.19 *Program memory (Mode B = 0)*

Program Memory	*Address*
EXTERNAL 8K (PMOVLAY = 1 or 2, MODE B = 0)	0x3FFF
RESERVED MEMORY RANGE	0x1FFF 0x1000
4k INTERNAL	0x1FFF 0x0000

Table 16.20 *External address of PMOVLAY*

PMOVLAY	*Memory*	*A13*	*A12:0*
0	Internal	Not Applicable	Not Applicable
1	External Overlay 1	0	13 LSBs of Address Between 0×2000 and 0×3FFF
2	External Overlay 2	1	13 LSBs of Address Between 0×2000 and 0×3FFF

Note: Addresses 0×2000 through 0×3FFF should not be accessed when PMOVLAY = 0.

This organization provides for two external 8K overlay segments using only the normal 14 address bits, which allows for simple program overlays using one of the two external segments in place of the on-chip memory. Care must be taken in using this overlay space in that the processor core (i.e., the sequencer) does not take into account the PMOVLAY register value. For example, if a loop operation is occurring on one of the external overlays and the program changes to another external overlay or internal memory, an incorrect loop operation could occur. In addition, care must be taken in interrupt service routines as the overlay registers are not automatically saved and restored on the processor mode stack.

When Mode B = 1, booting is disabled and overlay memory is disabled the 4K internal PM cannot be accessed with MODE B = 1. Table 16.21 shows the

Table 16.21 *Program memory (Mode B= 1)*

Program Memory	*Address*
Reserved	0×3FFF 0×2000
8K External	0×1FFF 0×0000

memory map (B = 1) configuration.

Data Memory

The ADSP 2184 has 4K IO-bit words of internal data memory. In addition, the ADSP 2184 allows the use of 8K external memory overlays.Table 16.22 shows the organization of the data memory.

There are 4K words of memory accessible internally when the DMOVLAY register is set to 0. When DMOVLAY is set to 1 or 2, external accesses occur at addresses 0x0000 through 0xIFFF. The external address is generated as shown in Table 16.23.

Table 16.22 *Organisation of the data memory*

Data Memory	*Address*
32 Memory- Mapped Registers	0x3FFF 0x3FEO
4064 Reserved Words	0x3FDF 0x3000
Internal 4K words	0x2FFF 0x2000
External 8K (DMOVLAY = 1, 2)	0x1FFF 0x0000

Table 16.23 *External address of DMOVLAY*

DMOVLAY	*Memory*	*A13*	*A12:0*
0	Internal	Not Applicable	Not Applicable
1	External Overlay 1	0	13 LSBs of Address between 0x0000 and 0xlFFF
2	External Overlay 2	1	13 LSBs of Address between 0x0000 and 0xlFFF

This organization allows for two external 8K overlays using only the normal 14 address bits. All internal accesses complete in one cycle. Accesses to external memory are timed using the wait states specified by the DWAIT register.

16.13.7 I/O Space (Full Memory Mode)

The ADSP 2184 supports an additional external memory space called I/O space. This space is designed to support simple connections to peripherals or to bus interface ASIC data registers. I/O space supports 2048 locations. The lower eleven bits of the external address bus are used; the upper three bits are undefined. Two instructions were added to the core ADSP 2100 Family instruction set to read from and write to I/O memory space. The I/O space also has four dedicated three-bit wait state registers, IOWAITO-3, that specify up to seven wait states to be automatically generated for each of four regions. The wait states act on address ranges as shown in Table 16.24.

Table 16.24 *Wait states*

Address Range	*Wait State Register*
0x000-0x1FF	IOWAIT0
0x200-0x3FF	IOWAIT1
0x400-0x5FF	IOWAIT2
0x600-0x7FF	IOWAIT3

Composite Memory Select ($\overline{CMS}$)

The ADSP 2184 has a programmable memory select signal that is useful for generating memory select signals for memories mapped to more than one space. The $\overline{CMS}$ signal is generated to have the same timing as each of the individual memory select signals($\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$, $\overline{IOMS}$), but can combine their functionality.

Each bit in the CMSSEL register, when set, causes the $\overline{CMS}$ signal to be asserted when the selected memory select is asserted. For example, to use a 32K word memory to act as both program and data memory, set the $\overline{PMS}$ and $\overline{DMS}$ bits in the CMSSEL register and use the $\overline{CMS}$ pin to drive the chip select of the memory and use either $\overline{DMS}$ or $\overline{PMS}$ as the additional address bit.

The $\overline{CMS}$ pin functions as the other memory select signals, with the same timing and bus request logic. A logic '1' in the enable bit causes the assertion of the $\overline{CMS}$ signal at the same time as the selected memory select signal. All enable bits, except the $\overline{BMS}$ bit, default to 1 at reset.

Byte Memory

The byte memory space is a bidirectional, 8-bit-wide, external memory space used to store programs and data. Byte memory is accessed using the BDMA feature. The byte memory space consists of 256 pages, each of which is 16K × 8.

The byte memory space on the ADSP 2184 supports read and write operations as well as four different data formats. The byte memory uses data bits 15:8 for data. The byte memory uses data bits 23: 16 and address bits 13:0 to create a 22-bit address. This allows up to a 4 M × 8 (32 megabit) ROM or RAM to be used without glue logic. All byte memory accesses are timed by the BMW AIT register.

Byte Memory DMA (BDMA, Full Memory Mode)

The Byte Memory DMA controller allows loading and storing of program instructions and data using the byte memory space. The BDMA circuit is able to access the byte memory space while the processor is operating normally and steals only one DSP cycle per 8-, 16- or 24-bit word transferred.

The BDMA circuit supports four different data formats that are selected by the BTYPE register field. The appropriate number of 8-bit accesses is done from the byte memory space to build the word size selected. Table 16.25 shows the data formats supported by the BDMA circuit.

Table 16.25 *Data format of BDMA*

BTYPE	*Internal Memory Space*	*Word Size*	*Alignment*
00	Program Memory	24	Full Word
01	Data Memory	16	Full Word
10	Data Memory	8	MSBs
11	Data Memory	8	LSBs

Unused bits in the 8-bit data memory formats are filled with 0s. The BIAD register field is used to specify the starting address for the on-chip memory involved with the transfer. The 14-bit BEAD register specifies the starting address for the external byte memory space. The 8-bit BMPAGE register specifies the starting page for the external byte memory space. The BDIR register field selects the direction of the transfer. The 14-bit BWCOUNT register specifies the number of DSP words to transfer and initiates the BDMA circuit transfers.

BDMA accesses can cross page boundaries during sequential addressing. A BDMA interrupt is generated on the completion of the number of transfers specified by the BWCOUNT register. The BWCOUNT register is updated after each transfer so it can be used to check the status of the transfers. When it reaches zero, the transfers have finished and a BDMA interrupt is generated. The BMPAGE and BEAD registers must not be accessed by the DSP during BDMA operations.

The source or destination of a BDMA transfer will always be on-chip program or data memory, regardless of the values of Mode B, PMOVLAY or DMOVLAY.

When the BWCOUNT register is written with a nonzero value, the BDMA circuit starts executing byte memory accesses with wait states set by BMWAIT. These accesses 'continue until the count reaches zero. When enough accesses have occurred to create a destination word, it is transferred to or from on-chip memory. The transfer takes one DSP cycle. DSP accesses to external memory have priority over BDMA byte memory accesses.

The BDMA Context Reset bit (BCR) controls whether the processor is held off while the BDMA accesses are occurring. Setting the BCR bit to 0 allows the processor to continue operations. Setting the BCR bit to 1 causes the processor to stop execution while the BDMA accesses are occurring, to clear the context of the processor and start execution at address 0 when the BDMA accesses have completed.

Internal Memory DMA Port (IDMA Port; Host Memory Mode) The IDMA Port provides an efficient means of communication between a host system and the ADSP 2184. The port is used to access the on-chip program memory and data memory of the DSP with only one DSP cycle per word overhead. The IDMA port cannot, however, be used to write to the DSP's memory-mapped control registers.

The IDMA port has a 16-bit multiplexed address and data bus that supports 24-bit program memory. The IDMA port is completely asynchronous and can be written to while the ADSP 2184 is operating at full speed.

The DSP memory address is latched and then automatically incremented after each IDMA transaction. An external device can, therefore, access a block of sequentially addressed memory by specifying only the starting address of the block. This increases throughput as the address does not have to be sent for each memory access.

IDMA Port access occurs in two phases. The first is the IDMA Address Latch cycle. When the acknowledge is asserted, a 14-bit address and l-bit destination type can be driven onto the bus by an external device. The address specifies an on-chip memory location, the destination type specifies whether it is a DM or PM access. The falling edge of the IDMA address hitch signal ($\overline{IAL}$) or the missing edge of the IDMA select signal ($\overline{IS}$) latches this value into the IDMAA register.

Once the address is stored, data can then either be read from or written to the ADSP 2184's on-chip memory. Asserting the select line ($\overline{IS}$) and the appropriate read or write line ($\overline{IRD}$) and ($\overline{IWR}$) respectively) signals the ADSP 2184 that a particular transaction is required. In either case, there is a one-processor cycle delay for synchronization. The memory access consumes one additional processor cycle.

Once an access has occurred, the latched address is automatically incremented and another access can occur.

Through the IDMAA register, the DSP can also specify the starting address and data format for DMA operation.

MOTOROLA PROCESSORS 16.14

The Motorola DSP560xx family has several kinds of 24-bit fixed-point digital signal processors based on common core architecture. This processorfamily is popularly used in digital audio applications, where its 24-bit word improves the dynamic range and reduces quantization noise compared to 16-bit fixed-point DSPs.

The DSP56000 and DSP56001 are the first members of the DSP560xx family which were introduced in 1987. Motorola introduced the DSP56002 and DSP56011 in 1992 and 1996 respectively, which were used for audio decoding of the DVD (Digital Versatile Disc) players.

The DSP560xx has a 24-bit, fixed-point data path which features an integrated MAC/ALU with a $24 \times 24 - 48$ bit multiplier, a 56-bit ALU, and two 56-bit accumulators which provide eight guard bits each. Every operation of DSP560xx data path uses fractional arithmetic. As DSP560xx has no integer multiply instruction for integer multiplication to be performed by the programmers, it isrequired to convert the result of a fractional multiply to integer format by shifting a sign bit into the accumulator

MSB. The data path can shift values one bit left or right and it provides 48-bit double-precision arithmetic. A carry bit can be updated by shifting operation and ALU operation.

The Motorola DSP561xx family is based on Motorola's DSP56100 16-bit fixed-point DSP core.Its architecture, instruction set and development environment are similar to that of Motorola's DSP560xx 24-bit fixed-point processor family.DSP561xx processors execute at speed up to 30 MIPS and its applications include cell phones and pagers.

The two processors in the DSP561xx family are the DSP56156 and the DSP56166. Both the processors provide on-chip voice band A/D and D/A converters (codes).

The DSP561xx family's data path is based on a 16–32-bit multiplier, integrated with a 40-bit accumulator, which is provided with eight guard bits. Multiply-accumulate operations are executed in a single clock cycle. The multiply-accumulate unit supports signed/unsigned multiplication. Though the data path does not have a barrel shifter, a shifting unit provides accumulator which shifts one or four bits left, or one, four, or 16 bits right. Convergent rounding, biased rounding, saturation shifting and output shifting are supported by the data paths.

The Motorola DSP96002 is a 32-bit IEEE standard 754 floating-point processor with 32-bit integer support. The overall architecture is similar to that of the Motorola DSP560xx family. The execution speeds of the fastest versions of the processors are at 20 MIPS.

The DSP96002 are popularly used in scientific and military applications (especially those involving the Fast Fourier transform).

SELECTION OF DIGITAL SIGNAL PROCESSORS 16.15

The factors which influence the selection of a DSP processor for a specific application are described as follows.

Architectural Features The key features of DSP architecture are on-chip memory size, special instructions and I/O capability. On-chip memory is essential in applications where large memory is required. It helps in access of data at high speed and fast execution of the program. For applications requiring high memory(e.g. digital audio-Dolby AC-2, FAX/Modem, MPEG coding/decoding), the size of internal RAM should be high. For applications which require fast and efficient communication or data flow with the outside world, I/O features such interface to ADC and DACs, DMA capability and support for multiprocessing is of importance. Depending on the application, a rich set of special instructions are required to support DSP operations, e.g. zero-overhead looping capability, dedicated DSP instructions and circular addressing.

Type of Arithmetic Fixed and floating-point arithmetic are the two important types of arithmetic used in modern digital signal processors. Fixed-point processors are used in low-cost and high-volume applications (e.g. cell phones and computer disk drives). Floating-point arithmetic processors are used for application with wide and variable dynamic range requirements which are expensive.

Execution Speed The execution speed of digital signal processors plays an important role in the selection of a processor. The execution speed is measured in terms of the clock speed of the processor (in MHz) and the number of instructions performed [in Millions of Instructions Per Seconds (MIPS)]. For floating-point digital signal processors, execution speed is measured in Millions of Floating Point Operations Per Second (MFLOPS). For example, the TMS320C6X family of processors can execute as many as eight instructions in a cycle. The number of operations performed in each cycle differs from processor to processor.

Word Length Processor data-word length is an important parameter in DSP as it has a significant impact on the signal quality. In general, if the data word is long, less errors are introduced by digital signal processing. Fixed-point digital signal processors used in telecommunications markets employ 16-bit word length (e.g. TMS320C54x), whereas the processors used in high-quality audio applications use 24-bits (e.g. DSP56300).For example, fixed-point audio processing requires a processor word length of at least 24-bits. In floating point DSP processors, a 32-bit data size (24-bit mantissa and 8-bit exponents) is used for single-precision arithmetic. Most floating-point DSP processors have fixed point arithmetic capability and often support variable data size.

REVIEW QUESTIONS

16.1 What are the relative merits and demerits of RISC and CISC processors?
16.2 What are the advantages of DSP processors in relation to general-purpose processors?
16.3 Explain *Von Neumann* and *Harvard* architectures and explain why the *Von Neumann* architecture is not suitable for DSP operations.
16.4 Write down the applications of each of the families of TIs DSPs.
16.5 Explain how a higher throughput is obtained using the VLIW architecture. Give an example of a DSP that has VLIW architecture
16.6 What is a digital signal processor?
16.7 Mention various generations of digital signal processors.
16.8 Explain the broad applications of digital signal processors.
16.9 What are the differences between fixed-type processors and floating type processors?
16.10 List the basic characteristics of a digital signal processor.
16.11 List the family members of the first-generation TMS processor and note down the distinguish features.
16.12 Explain arithmetic-logic unit.
16.13 Explain the basic operation that takes place while implementing a typical ALU operation.
16.14 Explain Barrel shifter.
16.15 Explain the major updates of TMS320C2X over the TMS320C1X processor.
16.16 Draw the schematic block diagram of the TMS320C2X processor and explain the major block diagram of the same.
16.17 What is the major advantage of having on-chip memory?
16.18 Explain MAC operation.
16.19 Name the six registers used in the TMS320C2X processor.
16.20 Explain the major updates of TMS320C3X over the TMS320C2X processor.
16.21 Distinguish various features of the TMS320C3X processor family members.
16.22 List the various key features of the TMS320C4X processor.
16.23 List the enhanced features of the TMS320C5X processor.
16.24 Draw the schematic block diagram of the TMS320C5X processor and explain the major block diagram of the same.
16.25 List the various addressing modes used in the TMS320C5X processor.
16.26 List the enhanced features of the TMS320C6X processor.
16.27 Draw the schematic block diagram of the TMS320C6X processor and explain.
16.28 List the various addressing modes used in the TMS320C6X processor.
16.29 Explain the memory architecture of the TMS320C6X processor.
16.30 Draw the architectural block diagram of the ADSP 2184 processor and explain.
16.31 Explain the memory architecture of ADSP 2184 processor.
16.32 Explain about low-power-mode operation of ADSP 2184 processor.
16.33 Write short notes on the Motorola DSP processor.
16.34 Discuss the characteristics for selection of digital signal processors.

Chapter 17 MATLAB Programs

INTRODUCTION 17.1

MATLAB stands for MATrix LABoratory. It is a technical computing environment for high performance numeric computation and visualisation. It integrates numerical analysis, matrix computation, signal processing and graphics in an easy-to-use environment, where problems and solutions are expressed just as they are written mathematically, without traditional programming. MATLAB allows us to express the entire algorithm in a few dozen lines, to compute the solution with great accuracy in a few minutes on a computer, and to readily manipulate a three-dimensional display of the result in colour.

MATLAB is an interactive system whose basic data element is a matrix that does not require dimensioning. It enables us to solve many numerical problems in a fraction of the time that it would take to write a program and execute in a language such as FORTRAN, BASIC, or C. It also features a family of application specific solutions, called ***toolboxes***. Areas in which toolboxes are available include signal processing, image processing, control systems design, dynamic systems simulation, systems identification, neural networks, wavelength communication and others. It can handle linear, non-linear, continuous-time, discrete-time, multivariable and multirate systems. This chapter gives simple programs to solve specific problems that are included in the previous chapters. All these programs have been tested under version 10 of MATLAB.

REPRESENTATION OF BASIC SIGNALS 17.2

MATLAB programs for the generation of unit impulse, unit step, ramp, exponential, sinusoidal and cosine sequences are as follows.

17.2.1 Program for Generation of Unit Impulse Signal

```
clc
clear all
close all
n=[-4:4];
z=[zeros(1,4),1,zeros(1,4)];
subplot(1,1,1);
stem(n,z);
axis tight
xlabel('n ---> ');
ylabel('Amplitude --- >');
title('Unit Impulse');
```

17.2.2 Program for Generation of Unit Step Sequence [$u(n) - u(n-1)$]

```
clc
clear all
close all
n=input('Enter the N Value');
y=[ones(1,n)];
subplot(1,1,1);
stem(y);
axis tight
xlabel('n --- >');
ylabel('Amplitude --- >');
title('Unit Step');
```

17.2.3 Program for Generation of Sine Wave

```
clc
clear all
close all
n=0:0.1:4;
x=sin(2*pi*n);
figure,
subplot(1,1,1)
plot(n,x)
axis tight
xlabel('n --- >');
ylabel('Amplitude --- >');
title('sine Wave');
```

17.2.4 Program for Generation of Cosine Wave

```
clc
clear all
close all
f=4;
t=0:0.1:4;
x=cos(2*pi*f*t);
figure,
subplot(1,1,1)
plot(n,x)
axis tight
xlabel('frequency (Hz)');
ylabel('amplitude (v)');
title('cosine');
```

17.2.5 Program for Generation of Square Wave

```
clc
clear all
close all
```

```
f=8;
t=0:0.01:1;
x=square(2*pi*f*t);
subplot(1,1,1);
plot(t,x);
axis tight
xlabel('n ---- >');
ylabel('Amplitude --- >');
title('Square wave');
```

17.2.6 Program for Generation of Ramp Sequence

```
clc
clear all
close all
t=0:0.1:1;
r=t;
subplot(1,1,1);
stem(r);
axis tight
xlabel('n ---- >');
ylabel('Amplitude --- >');
title('Ramp');
```

17.2.7 Program for Generation of Exponential Sequence

```
clc
clear all
close all
t=0:0.01:1;
a=5;
z=exp(-a*t);
subplot(1,1,1);
stem(z);
axis tight
xlabel('n ---- >');
ylabel('Amplitude --- >');
title('Exponential');
```

17.2.8 Program for Generation of Damped Sequence

```
clc
clear all
close all
a=6;
t=0:0.001:3;
x=a*sin(2*pi*a*t);
z=exp(-t);
q=x.*z;
```

```
subplot(1,1,1);
plot(t,q);
axis tight
xlabel('n ---- >');
ylabel('Amplitude --- >');
title('Damped');
```

17.2.9 Program for Generation of Sinc Wave

```
clc
clear all
close all
b=linspace(-10,10);
y=sinc(b);
subplot(1,1,1);
plot(b,y);
axis tight
xlabel('n ---- >');
ylabel('Amplitude --- >');
title('sinc');
```

CONVOLUTION 17.3

17.3.1 Program to Obtain Linear Convolution of Two Finite Length Sequences

```
clc; clear all; close all;
x=input('Enter the 1st sequence ');
h=input('Enter the 2nd sequence ');
n1=length(x);
n2=length(h);
N=n1+n2-1;
x1=[x zeros(1,N-n1)];
h1=[h zeros(1,N-n2)];
for n=1:N
for m=1:N
if n>m
H(m,n)=0;
else
H(m,n)=h1(m-(n-1));
end
end
end
y=(H*x1');
disp(y')
subplot(3,1,1);
stem(x);
title('Sequnce 1');
```

```
xlabel('Samples --- >');
ylabel('Amplitude');
subplot(3,1,2);
stem(h);
title('Sequnce 2');
xlabel('Samples --- >');
ylabel('Amplitude');
subplot(3,1,3);
stem(y);
title('Linear Convlution Output');
xlabel('Samples --- >');
ylabel('Amplitude');
```

17.3.2 Program for Linear Convolution

```
clc; clear all; close all;
x=input('enter the 1st sequence');
h=input('enter the 2nd sequence');
y=conv(x,h);
figure;subplot(3,1,1);
stem(x);ylabel('Amplitude -->');
xlabel('(a) n -->');
subplot(3,1,2);
stem(h);ylabel('Amplitude -->');
xlabel('(b) n -->');
subplot(3,1,3);
stem(y);ylabel('Amplitude -->');
xlabel('(c) n -->');
disp('The resultant signal is');y
```

17.3.3 Program for Computing Circular Convolution

```
clc; clear all; close all;
x=input('Enter the sequence 1 ');
h=input('Enter the sequence 2 ');
m=length(x);
n=length(h);
N=max(m,n);
s=m-n;
j=1;
z=[];
if(s==0)
h=[h,zeros(1,s)];
else
x=[x,zeros(1,-s)];
h=[h,zeros(1,s)];
end
```

```
for n=1:N
y=0;
for i=1:N
j=(n-i)+1;
if(j<=0)
j=N+j;
end
y=y+(x(i)*h(j));
end
z=[z y];
end
subplot(3,1,1)
stem(x)
title('Signal 1')
xlabel('Samples --- >')
ylabel('Amplitude')
subplot(3,1,2)
stem(h)
title('Signal 2')
xlabel('Samples --- >')
ylabel('Amplitude')
subplot(3,1,3)
stem(z)
title('Circular Convulution Output')
xlabel('Samples --- >')
ylabel('Amplitude')
```

17.3.4 Program for Computing Block Convolution using Overlap Save Method

```
clc; clear all; close all;
x=input('Enter the sequence x(n) = ');
h=input('Enter the sequence h(n) = ');
n1=length(x);
n2=length(h);
N=n1+n2-1;
h1=[h zeros(1,N-n1)];
n3=length(h1);
y=zeros(1,N);
x1=[zeros(1,n3-n2) x zeros(1,n3)];
H=fft(h1);
for i=1:n2:N
  y1=x1(i:i+(2*(n3-n2)));
  y2=fft(y1);
  y3=y2.*H;
  y4=round(ifft(y3));
  y(i:(i+n3-n2))=y4(n2:n3);
```

```
end
disp('The output sequence y(n)=');
disp(y(1:N));
stem(y(1:N));
title('Overlap Save Method');
xlabel('n');
ylabel('y(n)');
```

17.3.5 Program for Computing Block Convolution using Overlap Add Method

```
clc; clear all; close all;
x=input('Enter the sequence x(n) = ');
h=input('Enter the sequence h(n) = ');
n1=length(x);
n2=length(h);
N=n1+n2-1;
y=zeros(1,N);
h1=[h zeros(1,n2-1)];
n3=length(h1);
y=zeros(1,N+n3-n2);
H=fft(h1);
for i=1:n2:n1
  if i<=(n1+n2-1)
    x1=[x(i:i+n3-n2) zeros(1,n3-n2)];
  else
    x1=[x(i:n1) zeros(1,n3-n2)];
  end
  x2=fft(x1);
  x3=x2.*H;
  x4=round(ifft(x3));
  if (i==1)
    y(1:n3)=x4(1:n3);
  else
    y(i:i+n3-1)=y(i:i+n3-1)+x4(1:n3);
  end
end
disp('The output sequence y(n)=')
disp(y(1:N))
stem((y(1:N))
title('Overlap Add Method')
xlabel('n')
ylabel('y(n)')
```

DISCRETE CORRELATION 17.4

17.4.1 Program for Computing Cross-correlation

```
clc; clear all; close all;
x=input('enter the first sequence');
```

```
y=input('enter the second sequence');
yr=fliplr(y);
disp(yr);
x1=length(x);
y1=length(yr);
n=x1+y1-1;
x=[x,zeros(1,n-x1)];
yr=[yr,zeros(1,n-y1)];
z=x;
h=zeros(n,n);
k=0;
for j=1:n
    for k=0:n-1
        if(j+k<=n)
            h(j+k,j)=x(k+1);
        else
            h(j+k-n,j)=x(k+1);
        end
    end
end
c=(h*yr')';
disp('correlated output=')
disp(c)
subplot(3,1,1)
stem(z)
title('signal 1')
xlabel('samples')
ylabel('amplitude')
subplot(3,1,2)
stem(yr)
title('signal 2')
xlabel('samples')
ylabel('amplitude')
subplot(3,1,3)
stem(c)
title('cross-correlation output')
xlabel('samples')
ylabel('amplitude')
```

17.4.2 Program for Computing Cross-correlation

```
clc;clear all; close all;
x=input('enter the 1st sequence');
h=input('enter the 2nd sequence');
y=xcorr(x,h);
figure;subplot(3,1,1);
stem(x);ylabel('Amplitude -->');
xlabel('(a) n -->');
```

```
subplot(3,1,2);
stem(h);ylabel('Amplitude -->');
xlabel('(b) n -->');
subplot(3,1,3);
stem(fliplr(y));ylabel('Amplitude -->');
xlabel('(c) n -->');
disp('The resultant signal is'); fliplr(y)
```

17.4.3 Program for Computing Auto-correlation

```
clc; clear all; close all;
x=input('enter the first sequence');
yr=fliplr(x);
disp(yr);
x1=length(x);
y1=length(yr);
n=x1+y1-1;
x=[x,zeros(1,n-x1)];
yr=[yr,zeros(1,n-y1)];
z=x;
h=zeros(n,n);
k=0;
for j=1:n
    for k=0:n-1
        if(j+k<=n)
            h(j+k,j)=x(k+1);
        else
            h(j+k-n,j)=x(k+1);
        end
    end
end
c=(h*yr')';
disp('correlated output=')
disp(c)
subplot(2,1,1)
stem(z)
title('signal 1')
xlabel('samples')
ylabel('amplitude')
subplot(2,1,2)
stem(c)
title('auto-correlation output')
xlabel('samples')
ylabel('amplitude')
```

17.4.4 Program for Computing Autocorrelation Function

```
clc; clear all; close all;
x=input('enter the sequence');
```

```
y=xcorr(x,x);
figure;subplot(2,1,1);
stem(x);ylabel('Amplitude -->');
xlabel('(a) n -->');
subplot(2,1,2);
stem(fliplr(y));ylabel('Amplitude -->');
xlabel('(a) n -->');
disp('The resultant signal is');fliplr(y)
```

STABILITY TEST 17.5

% Program for stability test

```
clc;clear all;close all;
b=input('enter the denominator coefficients of the filter');
k=poly2rc(b);
knew=fliplr(k);
s=all(abs(knew)1);
if(s== 1)
  disp('"Stable system"');
else
  disp('"Non-stable system"');
end
```

SAMPLING THEOREM 17.6

```
clc; clear all; close all;
f=input('Enter the Sine wave frequency');
f1=f;f2=2*f;f3=8*f;
n=0:0.5:20;
x1=sin(2*pi*(f/f1)*n);
x2=sin(2*pi*(f/f2)*n);
x3=sin(2*pi*(f/f3)*n);
figure,
subplot(3,1,1)
plot(x1)
title('Under Sampled Signal Fsamp < 2*Fsig')
xlabel('Samples --->')
ylabel('Amplitude --->')
subplot(3,1,2)
plot(x2)
title('Critical Sampled Signal Fsamp = 2*Fsig')
xlabel('Samples --->')
ylabel('Amplitude --->')
subplot(3,1,3)
plot(x3)
title('Over Sampled Signal Fsamp > 2*Fsig')
xlabel('Samples --->')
ylabel('Amplitude --->')
```

DISCRETE FOURIER TRANSFORM 17.7

17.7.1 Program for Computing DFT

```
clc; clear all; close all;
% fs=10000;
%t=0:1/fs:0.01;
% x=sin(2*pi*(4000/fs)*t);
x=input('Enter the sequence:  ');
l=length(x);
n=input('Enter the number of points:  ');
x1=[x zeros(1,n-l)];
y=zeros(1,n);
for k=1:1:n
    for j=1:1:n
       y(k)=y(k)+x1(j)*exp(-2*pi*1i*(k-1)*(j-1)/n);
    end
end
display(y)
subplot(211)
stem(x)
xlabel('Samples')
ylabel('Amplitude')
title('Input Sequence')
subplot(212)
stem(abs(y))
xlabel('Samples')
ylabel('Amplitude')
title('DFT Output-Magnitude')
```

FAST FOURIER TRANSFORM 17.8

17.8.1 Program for Computing Decimation-In-Time FFT

```
clc; clear all; close all;
x=input('Enter the sequence: ');
l=length(x);
k=0;
while((2^k)<l)
    k=k+1;
end
x1=[x zeros(1,(2^k)-l)];
n=length(x1);
y=zeros(1,n);
j=1;
for k=1:2:n
    xe(j)=x1(k);
    j=j+1;
```

```
end
j=1;
for k=2:2:n
    xo(j)=x1(k);
    j=j+1;
end
for k=1:1:n
    for j=1:1:(n/2)
        y(k)=y(k)+xe(j)*exp(-4*pi*1i*(j-1)*(k-1)/n)+exp(-2*pi*1i*(k-
1)/n)*xo(j)*exp(-4*pi*1i*(j-1)*(k-1)/n);
    end
end
display(y)
figure,
subplot(211)
stem(x)
xlabel('Samples')
ylabel('Amplitude')
title('Input Sequence')
subplot(212)
stem(abs(y))
xlabel('Samples')
ylabel('Amplitude')
title('DITFFT Output-Magnitude')
```

17.8.2 Program for Computing Decimation-In-Frequency FFT

```
clc; clear all; close all;
x=input('Enter the sequence: ');
l=length(x);
k=0;
while((2^k)<l)
    k=k+1;
end
x11=[x zeros(1,(2^k)-l)];
n=length(x11);
y=zeros(1,n);
j=1;
for j=1:1:(n/2)
    x1(j)=x11(j);
end
 for j=((n/2)+1):1:n
     x2(j-(n/2))=x11(j);
 end
 k=1;
 j=1;
 for k=1:1:n
```

```
     for j=1:1:(n/2)
         y(k)=y(k)+(x1(j)+exp(-1i*pi*(k-1))*x2(j))*exp(-2*pi*1i*(j-
1)*(k-1)/n);
     end
 end
 display(y)
 subplot(211)
stem(x)
xlabel('Samples')
ylabel('Amplitude')
title('Input Sequence')
subplot(212)
stem(abs(y))
xlabel('Samples')
ylabel('Amplitude')
title('DIFFFT Output-Magnitude')
```

17.8.3 Program for Computing Fast Fourier Transform

```
clc;close all;clear all;
x=input('enter the sequence');
n=input('enter the length of fft');
X(k)=fft(x,n);
stem(y);ylabel('Imaginary axis -->');
xlabel('Real axis -->');
X(k)
```

17.8.4 Program for Computing Spectrum of Signals Using Fast Fourier Transform

% Spectrum of Single Sinusoid

```
clc; clear all; close all;
f1=60;
Fs=960;
N=128;
n=0:N-1;
x1=2*sin(2*pi*(f1/Fs)*n);
subplot(121);
stem(n,x1,'k');
title('Single Sinusoid')
xlabel('n')
ylabel('X(n)')
axis tight
specx=abs(fft(x1))
f=(0:N-1)*Fs/N
subplot(122)
stem(f(1:round(N/2)),specx(1:round(N/2)),'k')
```

```
title('Spectrum')
xlabel('frequency')
ylabel('X(f)'),axis tight;
```

% Sum of Two Sinusoids

```
clc; clear all; close all;
f1=60;
f2=120;
Fs=480;
N=64;
n=0:N-1;
x1=2*sin(2*pi*(f1/Fs)*n)+ 2*sin(2*pi*(f2/Fs)*n);
subplot(121);
stem(n,x1,'k');
title('Sum of two Sinusoids'),xlabel('n'),ylabel('x(n)'),
axis tight;
specx=abs(fft(x1));
f=(0:N-1)*Fs/N;
subplot(122);
stem(f(1:round(N/2)),specx(1:round(N/2)),'k');
title('Spectrum');xlabel('frequency');ylabel('X(f)')
axis tight;
```

% Resolution

```
clc; clear all; close all;
N=64;
f1=60;
f2=100;
fs=500;
n=0:N-1;
x=sin(2*pi*f1*n/fs)+sin(2*pi*f2*n/fs);
spec_x=abs(fft(x));
f=([0:N-1]/N)*fs;
subplot(2,2,1);
plot(f(1:round(N/2)),spec_x(1:round(N/2)));
xlabel('frequency')
ylabel('amplitude')
title('RESOLUTION')
axis tight;
N=16;
n=[0:N-1];
x=sin(2*pi*f1*n/fs)+sin(2*pi*f2*n/fs);
spec_y=abs(fft(x));
subplot(2,2,2);
plot(f(1:round(N/2)),spec_y(1:round(N/2)));
axis tight;
xlabel('frequency')
ylabel('amplitude')
```

% Zero Padding

```
clc; clear all; close all;
N=64;
f1=60;
f2=100;
fs=500;
n=0:N-1;
x=sin(2*pi*f1*n/fs)+sin(2*pi*f2*n/fs);
spec_x=abs(fft(x));
f=([0:N-1]/N)*fs;
subplot(2,2,1);
plot(f(1:round(N/2)),spec_x(1:round(N/2)));
xlabel('frequency');
ylabel('amplitude');
title('without ZERO PADDING');
axis tight;
x1=[x,zeros(1,200)];
N1=length(x1);
n=[0:N1-1];
spec_y=abs(fft(x1));
f=([0:N1-1]/N1)*fs;
subplot(2,2,2);
plot(f(1:round(N1/2)),spec_y(1:round(N1/2)));
xlabel('frequency')
ylabel('amplitude')
title('with ZERO PADDING')
axis tight;
```

% Picket Fence Effect

```
clc; clear all; close all;
f1=20;
f2=100;
f3=300;
fs=5000;
N=128;
n=0:N-1;
x=(sin(2*pi*f1*n/fs)+sin(2*pi*f2*n/fs)+sin(2*pi*f3*n/fs));
spec_x=abs(fft(x));
f=((0:N-1)/N)*fs;
stem(f(1:round(N/2)),spec_x(1:round(N/2)));
```

% Spectral Leakage

```
clc; clear all; close all;
f1=100;
f2=200;
fs1=1000;
```

```
N=32;
n=0:N-1;
x1=2*sin(2*pi*f1*n/f1)+2*sin(2*pi*f2*n/fs1);
specx1=abs(fft(x1));
f=[(0:N-1)/N]*fs1;
subplot(321);
stem(f,specx1);
xlabel('frequency');
ylabel('amplitude');
title('SPECTRAL LEAKAGE');
fs2=600;
x2=2*sin(2*pi*f1*n/fs2)+2*sin(2*pi*f2*n/fs2);
specx2=abs(fft(x1));
f3=[(0:N-1)/N]*fs2;
subplot(322)
stem(f3,specx2)
xlabel('frequency')
ylabel('amplitude')
fs4=500;
x4=2*sin(2*pi*f1*n/fs4)+2*sin(2*pi*f2*n/fs4);
specx4=abs(fft(x4));
f5=[(0:N-1)/N]*fs4;
subplot(323)
stem(f5,specx4)
xlabel('frequency')
ylabel('amplitude')
fs5=250;
x5=2*sin(2*pi*f1*n/fs5)+2*sin(2*pi*f2*n/fs5);
specx5=abs(fft(x5));
f6=[(0:N-1)/N]*fs5;
subplot(324)
stem(f5,specx5)
xlabel('frequency')
ylabel('amplitude')
axis tight;
```

BUTTERWORTH ANALOG FILTERS 17.9

17.9.1 Program for Design of Butterworth Analog Lowpass Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
wp=input('enter the passband freq');
ws=input('enter the stopband freq');
fs=input('enter the sampling freq');
w1=2*wp/fs;w2=2*ws/fs;
```

```
[n,wn]=buttord(w1,w2,rp,rs,'s');
[z,p,k]=butter(n,wn);
[b,a]=zp2tf(z,p,k);
[b,a]=butter(n,wn,'s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.9.2 Program for Design of Butterworth Analog High-pass Filter

```
clc; clear all; close all;
format long
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
wp=input('enter the passband freq');
ws=input('enter the stopband freq');
fs=input('enter the sampling freq');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=buttord(w1,w2,rp,rs,'s');
[b,a]=butter(n,wn,'high','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.9.3 Program for Design of Butterworth Analog Bandpass Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=buttord(w1,w2,rp,rs);
wn=[w1 w2];
```

```
[b,a]=butter(n,wn,'bandpass','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.9.4 Program for Design of Butterworth Analog Bandstop Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=buttord(w1,w2,rp,rs,'s');
wn=[w1 w2];
[b,a]=butter(n,wn,'stop','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

CHEBYSHEV TYPE-1 ANALOG FILTERS 17.10

17.10.1 Program for Design of Chebyshev Type-1 Low-pass Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=cheb1ord(w1,w2,rp,rs,'s');
[b,a]=cheby1(n,rp,wn,'s');
```

```
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.10.2 Program for Design of Chebyshev Type-1 High-pass Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=cheb1ord(w1,w2,rp,rs,'s');
[b,a]=cheby1(n,rp,wn,'high','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.10.3 Program for Design of Chebyshev Type-1 Bandpass Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=cheb1ord(w1,w2,rp,rs,'s');
wn=[w1 w2];
[b,a]=cheby1(n,rp,wn,'bandpass','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
```

```
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.10.4 Program for Design of Chebyshev Type-1 Bandstop Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=cheb1ord(w1,w2,rp,rs,'s');
wn=[w1 w2];
[b,a]=cheby1(n,rp,wn,'stop','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

CHEBYSHEV TYPE-2 ANALOG FILTERS 17.11

17.11.1 Program for Design of Chebyshev Type-2 Low Pass Analog Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=cheb2ord(w1,w2,rp,rs,'s');
[b,a]=cheby2(n,rs,wn,'s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
```

```
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.11.2 Program for Design of Chebyshev Type-2 High-Pass Analog Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=cheb2ord(w1,w2,rp,rs,'s');
[b,a]=cheby2(n,rs,wn,'high','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.11.3 Program for Design of Chebyshev Type-2 Bandpass Analog Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=cheb2ord(w1,w2,rp,rs,'s');
wn=[w1 w2];
[b,a]=cheby2(n,rs,wn,'bandpass','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
```

```
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.11.4 Program for Design of Chebyshev Type-2 Bandstop Analog Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=cheb2ord(w1,w2,rp,rs,'s');
wn=[w1 w2];
[b,a]=cheby2(n,rs,wn,'stop','s');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

BUTTERWORTH DIGITAL IIR FILTERS 17.12

17.12.1 Program for Design of Butterworth Low-pass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
wp=input('enter the passband freq');
ws=input('enter the stopband freq');
fs=input('enter the sampling freq');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]= buttord(w1,w2,rp,rs);
[b,a]= butter(n,wn);
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
```

```
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.12.2 Program for Design of Butterworth High-Pass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
wp=input('enter the passband freq');
ws=input('enter the stopband freq');
fs=input('enter the sampling freq');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=buttord(w1,w2,rp,rs);
[b,a]=butter(n,wn,'high');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.12.3 Program for Design of Butterworth Bandpass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
wp=input('enter the passband freq');
ws=input('enter the stopband freq');
fs=input('enter the sampling freq');
w1=2*wp/fs;w2=2*ws/fs;
[n]=buttord(w1,w2,rp,rs);
wn=[w1 w2];
[b,a]=butter(n,wn,'bandpass');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
```

```
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.12.4 Program for Design of Butterworth Bandstop Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
wp=input('enter the passband freq');
ws=input('enter the stopband freq');
fs=input('enter the sampling freq');
w1=2*wp/fs;w2=2*ws/fs;
[n]=buttord(w1,w2,rp,rs);
wn=[w1 w2];
[b,a]=butter(n,wn,'stop');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

CHEBYSHEV TYPE-1 DIGITAL FILTERS 17.13

17.13.1 Program for Design of Chebyshev Type-1 Low-pass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=cheb1ord(w1,w2,rp,rs);
[b,a]=cheby1(n,rp,wn);
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
```

```
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.13.2 Program for Design of Chebyshev Type-1 High-pass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=cheb1ord(w1,w2,rp,rs);
[b,a]=cheby1(n,rp,wn,'high');
w=0:.01/pi:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.13.3 Program for Design of Chebyshev Type-1 Bandpass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=cheb1ord(w1,w2,rp,rs);
wn=[w1 w2];
[b,a]=cheby1(n,rp,wn,'bandpass');
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
```

```
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.13.4 Program for Design of Chebyshev Type-1 Bandstop Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=cheb1ord(w1,w2,rp,rs);
wn=[w1 w2];
[b,a]=cheby1(n,rp,wn,'stop');
w=0:.1/pi:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

CHEBYSHEV TYPE-2 DIGITAL FILTERS 17.14

17.14.1 Program for Design of Chebyshev Type-2 Low-pass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=cheb2ord(w1,w2,rp,rs);
[b,a]=cheby2(n,rs,wn);
w=0:.01:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
```

```
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.14.2 Program for the Design of Chebyshev Type-2 High-pass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n,wn]=cheb2ord(w1,w2,rp,rs);
[b,a]=cheby2(n,rs,wn,'high');
w=0:.01/pi:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.14.3 Program for Design of Chebyshev Type-2 Bandpass Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=cheb2ord(w1,w2,rp,rs);
wn=[w1 w2];
[b,a]=cheby2(n,rs,wn,'bandpass');
w=0:.01/pi:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
```

```
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

17.14.4 Program for Design of Chebyshev Type-2 Bandstop Digital Filter

```
clc;close all;clear all;
format long
rp=input('enter the passband ripple...');
rs=input('enter the stopband ripple...');
wp=input('enter the passband freq...');
ws=input('enter the stopband freq...');
fs=input('enter the sampling freq...');
w1=2*wp/fs;w2=2*ws/fs;
[n]=cheb2ord(w1,w2,rp,rs);
wn=[w1 w2];
[b,a]=cheby2(n,rs,wn,'stop');
w=0:.1/pi:pi;
[h,om]=freqz(b,a,w);
m=20*log10(abs(h));
an=angle(h);
subplot(2,1,1);plot(om/pi,m);
ylabel('Gain in dB -->');xlabel('(a) Normalised frequency -->');
subplot(2,1,2);plot(om/pi,an);
xlabel('(b) Normalised frequency -->');
ylabel('Phase in radians -->');
```

FIR FILTER DESIGN USING WINDOW TECHNIQUES 17.15

17.15.1 Program for Design of FIR Low-pass, High-pass, Bandpass and Bandstop Filters using Rectangular Window

```
clc;clear all;close all;
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
fp=input('enter the passband freq');
fs=input('enter the stopband freq');
f=input('enter the sampling freq');
wp=2*fp/f;ws=2*fs/f;
num=-20*log10(sqrt(rp*rs))-13;
dem=14.6*(fs-fp)/f;
n=ceil(num/dem);
n1=n+1;
if (rem(n,2)~=0)
n1=n;
n=n-1;
end
y=boxcar(n1);
```

% LOW-PASS FILTER

```
b=fir1(n,wp,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,1);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(a) Normalised frequency -->');
```

% HIGH-PASS FILTER

```
b=fir1(n,wp,'high',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,2);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(b) Normalised frequency -->');
```

% BAND PASS FILTER

```
wn=[wp ws];
b=fir1(n,wn,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,3);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(c) Normalised frequency -->');
```

% BAND STOP FILTER

```
b=fir1(n,wn,'stop',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,4);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(d) Normalised frequency -->');
```

17.15.2 Program for Design of FIR Low-pass, High-pass, Bandpass and Bandstop Filters using Bartlett Window

```
clc;clear all;close all;
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
fp=input('enter the passband freq');
fs=input('enter the stopband freq');
f=input('enter the sampling freq');
wp=2*fp/f;ws=2*fs/f;
num=-20*log10(sqrt(rp*rs))-13;
dem=14.6*(fs-fp)/f;
n=ceil(num/dem);
n1=n+1;
if (rem(n,2)~=0)
n1=n;
n=n-1;
end
y=bartlett(n1);
```

% LOW-PASS FILTER

```
b=fir1(n,wp,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,1);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(a) Normalised frequency -->');
```

% HIGH-PASS FILTER

```
b=fir1(n,wp,'high',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,2);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(b) Normalised frequency -->');
```

% BAND PASS FILTER

```
wn=[wp ws];
b=fir1(n,wn,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,3);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(c) Normalised frequency -->');
```

% BAND STOP FILTER

```
b=fir1(n,wn,'stop',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,4);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(d) Normalised frequency -->');
```

17.15.3 Program for Design of FIR Low-pass, High-pass, Bandpass and Bandstop Digital Filters using Blackman Window

```
clc;clear all;close all;
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
fp=input('enter the passband freq');
fs=input('enter the stopband freq');
f=input('enter the sampling freq');
wp=2*fp/f;ws=2*fs/f;
num=-20*log10(sqrt(rp*rs))-13;
dem=14.6*(fs-fp)/f;
n=ceil(num/dem);
n1=n+1;
if (rem(n,2)~=0)
  n1=n;
  n=n-1;
end
y=blackman(n1);
```

% LOW-PASS FILTER

```
b=fir1(n,wp,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,1);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(a) Normalised frequency -->');
```

% HIGH-PASS FILTER

```
b=fir1(n,wp,'high',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,2);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(b) Normalised frequency -->');
```

% BAND PASS FILTER

```
wn=[wp ws];
b=fir1(n,wn,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,3);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(c) Normalised frequency -->');
```

% BAND STOP FILTER

```
b=fir1(n,wn,'stop',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,4);plot(o/pi,m);;ylabel('Gain in dB -->');
xlabel('(d) Normalised frequency -->');
```

17.15.4 Program for Design of FIR Low-pass, High-pass, Bandpass and Bandstop Filters using Chebyshev Window

```
clc;clear all;close all;
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
fp=input('enter the passband freq');
fs=input('enter the stopband freq');
f=input('enter the sampling freq');
r=input('enter the ripple value (in dBs)');
wp=2*fp/f;ws=2*fs/f;
num=-20*log10(sqrt(rp*rs))-13;
dem=14.6*(fs-fp)/f;
n=ceil(num/dem);
if(rem(n,2)~=0)
  n=n+1;
end
y=chebwin(n,r);
```

% LOW-PASS FILTER

```
b=fir1(n-1,wp,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,1);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(a) Normalised frequency -->');
```

% HIGH-PASS FILTER

```
b=fir1(n−1,wp,'high',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,2);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(b) Normalised frequency -->');
```

% BAND-PASS FILTER

```
wn=[wp ws];
b=fir1(n−1,wn,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,3);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(c) Normalised frequency -->');
```

% BAND-STOP FILTER

```
b=fir1(n−1,wn,'stop',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,4);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(d) Normalised frequency -->');
```

17.15.5 Program for Design of FIR Low-pass, High-pass, Bandpass and Bandstop Filters using Hamming Window

```
clc;clear all;close all;
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
fp=input('enter the passband freq');
fs=input('enter the stopband freq');
f=input('enter the sampling freq');
wp=2*fp/f;ws=2*fs/f;
num=−20*log10(sqrt(rp*rs))−13;
dem=14.6*(fs−fp)/f;
n=ceil(num/dem);
n1=n+1;
if (rem(n,2)~=0)
  n1=n;
  n=n−1;
```

```
end
y=hamming(n1);
```

% LOW-PASS FILTER

```
b=fir1(n,wp,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,1);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(a) Normalised frequency -->');
```

% HIGH-PASS FILTER

```
b=fir1(n,wp,'high',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,2);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(b) Normalised frequency -->');
```

% BAND PASS FILTER

```
wn=[wp ws];
b=fir1(n,wn,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,3);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(c) Normalised frequency -->');
```

% BAND STOP FILTER

```
b=fir1(n,wn,'stop',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,4);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(d) Normalised frequency -->');
```

17.15.6 Program for Design of FIR Low-pass, High-pass, Bandpass and Bandstop Filters using Hanning Window

```
clc;clear all;close all;
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
fp=input('enter the passband freq');
fs=input('enter the stopband freq');
f=input('enter the sampling freq');
wp=2*fp/f;ws=2*fs/f;
num=-20*log10(sqrt(rp*rs))-13;
dem=14.6*(fs-fp)/f;
n=ceil(num/dem);
n1=n+1;
if (rem(n,2)~=0)
n1=n;
```

```
n=n-1;
end
y=hamming(n1);
```

% LOW-PASS FILTER

```
b=fir1(n,wp,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,1);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(a) Normalised frequency -->');
```

% HIGH-PASS FILTER

```
b=fir1(n,wp,'high',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,2);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(b) Normalised frequency -->');
```

% BAND PASS FILTER

```
wn=[wp ws];
b=fir1(n,wn,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,3);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(c) Normalised frequency -->');
```

% BAND STOP FILTER

```
b=fir1(n,wn,'stop',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,4);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(d) Normalised frequency -->');
```

17.15.7 Program for Design of FIR Low-pass, High-pass, Bandpass and Bandstop Filters using Kaiser Window

```
clc;clear all;close all;
rp=input('enter the passband ripple');
rs=input('enter the stopband ripple');
fp=input('enter the passband freq');
fs=input('enter the stopband freq');
f=input('enter the sampling freq');
beta=input('enter the beta value');
wp=2*fp/f;ws=2*fs/f;
num=-20*log10(sqrt(rp*rs))-13;
dem=14.6*(fs-fp)/f;
n=ceil(num/dem);
n1=n+1;
```

```
if (rem(n,2)~=0)
  n1=n;
  n=n-1;
end
y=kaiser(n1,beta);
```

% LOW-PASS FILTER

```
b=fir1(n,wp,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,1);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(a) Normalised frequency -->');
```

% HIGH-PASS FILTER

```
b=fir1(n,wp,'high',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,2);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(b) Normalised frequency -->');
```

% BAND PASS FILTER

```
wn=[wp ws];
b=fir1(n,wn,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,3);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(c) Normalised frequency -->');
```

% BAND STOP FILTER

```
b=fir1(n,wn,'stop',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,4);plot(o/pi,m);ylabel('Gain in dB -->');
xlabel('(d) Normalised frequency -->');
```

UPSAMPLING A SINUSOIDAL SIGNAL 17.16

% Program for upsampling a sinusoidal signal by factor L

```
N=input('Input length of the sinusoidal sequence=');
L=input('Up Samping factor=');
fi=input('Input signal frequency=');
```

% Generate the sinusoidal sequence for the specified length *N*

```
n=0:N-1;
x=sin(2*pi*fi*n);
```

% Generate the upsampled signal

```
y=zeros (1,L*length(x));
y([1:L:length(y)])=x;
```

```
%Plot the input sequence
subplot (2,1,1);
stem (n,x);
title('Input Sequence');
xlabel('Time n');
ylabel('Amplitude');
%Plot the output sequence
subplot (2,1,2);
stem (n,y(1:length(x)));
title('[output sequence,upsampling factor=',num2str(L)]);
xlabel('Time n');
ylabel('Amplitude');
```

UPSAMPLING AN EXPONENTIAL SEQUENCE 17.17

% Program for upsampling an exponential sequence by a factor *M*

```
n=input('enter length of input sequence …');
l=input('enter up sampling factor …');
```

% Generate the exponential sequence

```
m=0:n−1;
a=input('enter the value of a …');
x=a.^m;
```

% Generate the upsampled signal

```
y=zeros(1,l*length(x));
y([1:l:length(y)])=x;
figure(1)
stem(m,x);
xlabel({'Time n';'(a)'});
ylabel('Amplitude');
figure(2)
stem(m,y(1:length(x)));
xlabel({'Time n';'(b)'});
ylabel('Amplitude');
```

DOWNSAMPLING A SINUSOIDAL SEQUENCE 17.18

% Program for down sampling a sinusoidal sequence by a factor M

```
N=input('Input length of the sinusoidal signal=');
M=input('Down samping factor=');
fi=input('Input signal frequency=');
```

%Generate the sinusoidal sequence

```
n=0:N−1;
m=0:N*M−1;
x=sin(2*pi*fi*m);
```

%Generate the down sampled signal

```
y=x([1:M:length(x)]);
```

```
%Plot the input sequence
subplot (2,1,1);
stem(n,x(1:N));
title('Input Sequence');
xlabel('Time n');
ylabel('Amplitude');
%Plot the down sampled signal sequence
subplot(2,1,2);
stem(n,y);
title(['Output sequence down sampling factor',num2str(M)]);
xlabel('Time n');
ylabel('Amplitude');
```

DOWNSAMPLING AN EXPONENTIAL SEQUENCE 17.19

% Program for downsampling an exponential sequence by a factor M

```
N=input('enter the length of the output sequence …');
M=input('enter the down sampling factor …');
```

% Generate the exponential sequence

```
n=0:N−1;
m=0:N*M−1;
a=input('enter the value of a …');
x=a.^m;
```

% Generate the downsampled signal

```
y=x([1:M:length(x)]);
figure(1)
stem(n,x(1:N));
xlabel({'Time n';'(a)'});
ylabel('Amplitude');
figure(2)
stem(n,y);
xlabel({'Time n';'(b)'});
ylabel('Amplitude');
```

DECIMATOR 17.20

% Program for downsampling the sum of two sinusoids using MATLAB's inbuilt decimation function by a factor M

```
N=input('Length of the input signal=');
M=input('Down sampling factor=');
f1=input('Frequency of first sinusoid=');
f2=input('Frequency of second sinusoid=');
n=0:N−1;
```

% Generate the input sequence

```
x=2*sin(2*pi*f1*n)+3*sin(2*pi*f2*n);
```

%Generate the decimated signal

```
% FIR low pass decimation is used
y=decimate(x,M,'fir');
%Plot the input sequence
subplot (2,1,1);
stem (n,x(1:N));
title('Input Sequence');
xlabel('Time n');
ylabel('Amplitude');
%Plot the output sequence
subplot (2,1,2);
m=0:N/M−1;
stem (m,y(1:N/M));
title(['Output sequence down sampling factor',num2str(M)]);
xlabel('Time n');
ylabel('Amplitude');
```

DECIMATOR AND INTERPOLATOR 17.21

% Program for downsampling and upsampling the sum of two sinusoids using MATLAB's inbuilt decimation and interpolation function by a factor of 20.

```
clc;clear all;close all;
Fs=200;f1=50;f2=100;
t=0:1/200:10;
y=3*cos(2*pi*(f1/Fs)*t)+cos(2*pi*(f2/Fs)*t);
figure,
stem(y)
xlabel({'Time in Seconds';'(a)})
ylabel('Amplitude')
figure,
```

Fig. 17.17 *(Contd.)*

```
stem(decimate(y,20))
xlabel({'Time in Seconds';'(b)})
ylabel('Amplitude')
figure,
stem(interp(decimate(y,20),2))
xlabel({'Time in Seconds';'(c)})
ylabel('Amplitude')
```

ESTIMATION OF POWER SPECTRAL DENSITY (PSD) 17.22

% Program for estimating PSD of two sinusoids plus noise

```
clc; close all; clear all;
f1=input('Enter the frequency of first sinusoid');
```

```
f2=input('Enter the frequency of second sinusoid');
fs=input('Enter the sampling frequency');
N=input("Enter the length of the input sequence');
N1=input("Enter the input FFT length 1');
N2=input("Enter the input FFT length 2');
```

%Generation of input sequence

```
t=0:1/fs:1;
x=2*sin(2*pi*f1*1)+3*sin(2*pi*f2*t)-randn(size(t));
```

%Generation of PSD for two different FFT lengths

```
Pxx1=abs(fft(x,N1)).^2/(N+1);
Pxx2=abs(fft(x,N2)).^2/(N+1);
%Plot the psd;
subplot(2,1,1);
plot ((0:(N1-1))/N1*fs,10*log10(Pxx1));
xlabel('Frequency in Hz');
ylabel('Power spectrum in dB');
title('[PSD with FFT length,num2str(N1)]');
subplot (2,1,2);
plot ((0:(N2-1))/N2*fs,10*log10(Pxx2));
xlabel('Frequency in Hz');
ylabel('Power spectrum in dB');
title('[PSD with FFT length,num2str(N2)]');
```

PSD ESTIMATOR 17.23

% Program for estimating PSD of a two sinusoids plus noise

```
%(i)non-overlapping sections
%(ii)overlapping sections and averaging the periodograms
clc; close all; clear all;
f1=input('Enter the frequency of first sinusoid');
f2=input('Enter the frequency of second sinusoid');
fs=input('Enter the sampling frequency');
N=input("Enter the length of the input sequence');
N1=input("Enter the input FFT length 1');
N2=input("Enter the input FFT length 2');
```

%Generation of input sequence

```
t=0:1/fs:1;
x=2*sin(2*pi*f1*1)+3*sin(2*pi*f2*t)-randn(size(t));
```

%Generation of PSD for two different FFT lengths

```
Pxx1=(abs(fft(x(1:256))).^2+abs(fft(x(257:512))).^2+abs
(fft(x(513:768))).^2/(256*3); %using nonoverlapping sections
```

```
Pxx2=(abs(fft(x(1:256))).^2+abs(fft(x(129:384))).^2+abs(fft(x(257:
512))).^2+abs(fft(x(385:640))).^2+abs(fft(x(513:768))).^2+abs
(fft(x(641:896))).^2/(256*6); %using overlapping sections
% Plot the psd;
subplot (2,1,1);
plot ((0:255)/256*fs,10*log10(Pxx1));
xlabel('Frequency in Hz');
ylabel('Power spectrum in dB');
title('[PSD with FFT length,num2str(N1)]');
subplot (2,1,2);
plot ((0:255)/256*fs,10*log10(Pxx2));
xlabel('Frequency in Hz');
ylabel('Power spectrum in dB');
title('[PSD with FFT length,num2str(N2)]');
```

PERIODOGRAM ESTIMATION 17.24

```
clc;clear all;close all;
    %n1----> size of input sequences
    %m1----> number of rows
    %a----> uniformly distributed random sequences
    %x----> Input sequence
    %K1----> Fast Fourier transform of input sequence
    %v----> variance
    %m---->Mean
    %p----> Power spectral density
    %correlation method
v=0.1;m=0
%Generation of y
m1=80;
n1=input ('Enter the size of noise signal');
a=rand(m1,n1);
y=zeros(1,n1);
for j=1:n1
    b=0;
for i=1:m1
    b=b+a(I,j);
end
y(j)=b;
end
% Generation of Input sequence, x
for k=1:n1
    x(k)=(12*v/m1)*(y(k)-(m1/2))+m;
end
figure,
```

```
plot(x)
title ('Zero mean, 0.1 variance signal');
xlabel('--->time(sec) ');
ylabel('--->amp');
%DFT computation of input sequence, x
K1=fft(x,128);
K2=conj(K1);
M=K1.*K2;
%IFFT Computation
R=ifft(M)/n1;
W=HAMMING(128);
P=R.*(transpose(W));
%Periodogram estimation
P=fft(p,128);
figure,
subplot (3,1,1)
plot(K1)
title ('Spectral estimate')
subplot(3,2,1)
plot(W)
title ('Window function')
subplot (3,1,3)
plot(p)
title ('Power Spectral Density')
```

WELCH PSD ESTIMATOR 17.25

% Program for estimating the PSD of sum of two sinusoids plus noise using Welch method

```
n=0.01:0.001:.1;
x=sin(.25*pi*n)+3*sin(.45*pi*n)+rand(size(n));
pwelch(x)
xlabel('Normalised Frequency (x pi rad/sample)');
ylabel('Power Spectrum Density (dB/rad/sample)')
```

WELCH PSD ESTIMATOR USING WINDOWS (I) 17.26

% Program for estimating the PSD of sum of two sinusoids using Welch method with an overlap of 50 percent and with Hanning, Hamming, Bartlett, Blackman and rectangular windows

```
fs=1000;
t=0:1/fs:3;
x=sin(2*pi*200*t)+sin(2*pi*400*t);
figure(1)
subplot(2,1,1)
pwelch(x,[],[],[],fs);
title('Overlay plot of 50 Welch estimates')
```

```
subplot(2,1,2)
pwelch(x,hanning(512),0,512,fs)
title('N=512 Overlap=50% Hanning')
figure(2)
subplot(2,1,1)
pwelch(x,[],[],[],fs);
title('Overlay plot of 50 Welch estimates')
subplot(2,1,2)
pwelch(x,hamming(512),0,512,fs)
title('N=512 Overlap=50% Hamming')
figure(3)
subplot(2,1,1)
pwelch(x,[],[],[],fs);
title('Overlay plot of 50 Welch estimates')
subplot(2,1,2)
pwelch(x,bartlett(512),0,512,fs)
title('N=512 Overlap=50% Bartlett')
figure(4)
subplot(2,1,1)
pwelch(x,[],[],[],fs);
title('Overlay plot of 50 Welch estimates')
subplot(2,1,2)
pwelch(x,blackman(512),0,512,fs)
title('N=512 Overlap=50% Blackman')
figure(5)
subplot(2,1,1)
pwelch(x,[],[],[],fs);
title('Overlay plot of 50 Welch estimates')
subplot(2,1,2)
pwelch(x,boxcar(512),0,512,fs)
title('N=512 Overlap=50% Rectangular')
```

WELCH PSD ESTIMATOR USING WINDOWS (II) 17.27

% Program for estimating the PSD of sum of two sinusoids plus noise using Welch method with an overlap of 50 percent and with Hanning, Hamming, Bartlett, Blackman and rectangular windows

```
fs=1000;
t=0:1/fs:3;
x=2*sin(2*pi*200*t)+5*sin(2*pi*400*t);
y=x+randn(size(t));
figure(1)
subplot(2,1,1);
pwelch(y,[],[],[],fs);
title('Overlay plot of 50 Welch estimates');
```

```
subplot(2,1,2);
pwelch(y,hanning(512),0,512,fs);
title('N=512 Overlap=50% Hanning');
figure(2)
subplot(2,1,1);
pwelch(y,[],[],[],fs);
title('Overlay plot of 50 Welch estimates');
subplot(2,1,2);
pwelch(y,hamming(512),0,512,fs);
title('N=512 Overlap=50% Hamming');
figure(3)
subplot(2,1,1);
pwelch(y,[],[],[],fs);
title('Overlay plot of 50 Welch estimates');
subplot(2,1,2);
pwelch(y,bartlett(512),0,512,fs);
title('N=512 Overlap=50% Bartlett');
figure(4)
subplot(2,1,1);
pwelch(y,[],[],[],fs);
title('Overlay plot of 50 Welch estimates');
subplot(2,1,2);
pwelch(y,blackman(512),0,512,fs);
title('N=512 Overlap=50% Blackman');
figure(5)
subplot(2,1,1);
pwelch(y,[],[],[],fs);
title('Overlay plot of 50 Welch estimates');
subplot(2,1,2);
pwelch(y,boxcar(512),0,512,fs);
title('N=512 Overlap=50% Rectangular');
```

PROGRAM FOR ADAPTIVE FILTERING 17.28

```
clc;clear all;close all;
f=input('Enter the frequency of the signal, f=');
fs=input('Enter the sampling frequency(>=2f). fs=');
% Generation of input signal
x=sin(2*pi*(f/fs)*[0:99]);
% Generation of noise
n=randn(1,100);
nn=max(abs(n));
n=n/nn;
% output signal
```

```
y=x+n;
% Filter
h=fir1(99,0.25);
xs=y.*h;
e=x-xs;
pp=1;
ee=abs(e);
ee=max(ee);
while(ee>0.1)
for i=1:100
hp(i)=h(i)+2*0.025*y(i).*e(i);
h(i)=hp(i);
xs(i)=y(i).*h(i);
e(i)=x(i)-xs(i);
end
pp=pp+1;
ee=abs(e);
ee=max(ee);
end
pp;
figure,
subplot(2,2,1)
plot(x)
title('Input signal')
xlabel('--->time(sec)')
ylabel('--->amp')
subplot(2,2,2)
plot(n)
title ('Noise signal')
xlabel('--->time(sec)')
ylabel('--->amp')
subplot(2,2,3)
plot(y)
title ('Noise added signal')
xlabel('--->time(sec)')
ylabel('--->amp')
subplot(2,2,4)
plot(xs)
title ('Enhanced signal')
xlabel('--->time(sec)')
ylabel('--->amp')
```

17.29 PROGRAM FOR SPECTRUM ESTIMATION USING AR PROCESS

```
clc;clear all;close all;
rx = [1;0.75;0.5;0.25;0];
omg = [0:500]*pi/500;
```

```
P=2;
Rbar = toeplitz(rx(1:P),rx(1:P));
[ahat,VarP] = osigest(Rbar,1);
Hhat = freqz(VarP,[1;ahat]',omg);
RxAP = abs(Hhat).*abs(Hhat);
subplot(2,2,1); plot(omg/pi,RxAP,'r','linewidth',2); hold on;
title('AP(2) Spectral Estimates');
xlabel('normalized frequency');
ylabel('R_x(e^{j\omega})');
axis([0,1,0,ceil(max(RxAP)/1.0)*1.0]);
P = 3;
Rbar = toeplitz(rx(1:P),rx(1:P));
[ahat,VarP] = osigest(Rbar,1);
Hhat = freqz(VarP,[1;ahat]',omg);
RxAP = abs(Hhat).*abs(Hhat);
subplot(2,2,2); plot(omg/pi,RxAP,'r','linewidth',2); hold on;
title('AP(3) Spectral Estimates');
xlabel('normalized frequency');
ylabel('R_x(e^{j\omega})');
axis([0,1,0,ceil(max(RxAP)/1.0)*1.0]);
P=4;
Rbar = toeplitz(rx(1:P),rx(1:P));
[ahat,VarP] = osigest(Rbar,1);
Hhat = freqz(VarP,[1;ahat]',omg);
RxAP = abs(Hhat).*abs(Hhat);
subplot(2,2,3); plot(omg/pi,RxAP,'r','linewidth',2); hold on;
title('AP(4) Spectral Estimates');
xlabel('normalized frequency');
ylabel('R_x(e^{j\omega})');
axis([0,1,0,ceil(max(RxAP)/1.0)*1.0]);
P=5;
Rbar = toeplitz(rx(1:P),rx(1:P));
i=1;
Ri=Rbar;
Ri(:,i)=[];Ri(i,:)=[];
di=Rbar(:,i);
di(i)=[];
ci=-Ri\di;
Pi=Rbar(i,i)+conj(ci')*di;
ahat=ci;VarP=pi;
Hhat = freqz(VarP,[1;ahat]',omg);
RxAP = abs(Hhat).*abs(Hhat);
subplot(2,2,4); plot(omg/pi,RxAP,'r','linewidth',2); hold on;
title('AP(5) Spectral Estimates');
xlabel('normalized frequency');
ylabel('R_x(e^{j\omega})');
axis([0,1,0,ceil(max(RxAP)/1.0)*1.0]);
```

STATE-SPACE REPRESENTATION 17.30

% Program for computing the state-space matrices from the given transfer function

```
function [A,B,C,D]=tf2ss(b,a);
a=input ('enter the denominator polynomials=');
b=input ('enter the numerator polynomials=');
p=length(a)−1;q=length(b)−1;N=max(p,q);
if(Np),a=[a,zeros(1,N−p)];end
if(Nq),b=[b,zeros(1,N−q)];end
A=[−a(2:N+1);[eye(N−1),zeros(N−1,1)]];
B=[1;zeros(N−1,1)];
C=b(2:N+1)−b(1)*(2:N+1);
D=b(1);
```

PARTIAL FRACTION DECOMPOSITION 17.31

% Program for partial fraction decomposition of a rational transfer function

```
function[c,A,alpha]=tf2pf(b,a);
a=input ('enter the denominator polynomials=');
b=input ('enter the numerator polynomials=');
p=length(a)−1;
q=length(b)−1;
a=(1/a(1))*reshape(a,1,p+1);
b=(1/a(1))*reshape(b,1,q+1);
if(q=p),%case of nonempty c(z)
  temp=toeplitz([a,zeros(1,q−p)]',[a(1),zeros(1,q−p)]);
  temp=[temp,[eye(p);zeros(q−p+1,p)]);
  temp=temp/b';
  c=temp(1:;q−p+1);
  d=temp(q−p+2:q+1)';
else
  c=[];
  d=[b,zeros(1,p−q−1)];
end
alpha=cplxpair (roots(a));';
A=zeros(1,p);
for k=1 :p
  temp=prod(alpha(k)−alpha(find(1:p≈=k)));
  if(temp==0),error('repeated roots in TF2PF');
  else,A(k)=polyval(d,alpha(k))/temp;
  end
end
```

INVERSE z-TRANSFORM 17.32

% Program for computing inverse z-transform of a rational transfer function

```
function x=invz(b,a,N);
b=input ('enter the numerator polynomials=');
```

```
a=input ('enter the denominator polynomials=');
N=input ('enter the number of points to be computed=');
[c,A,alpha]=tf2pf(b,a);
x=zeros (1,N);
x(1:length(c))=c;
for k=1:length(A),
  x=x+A(k)*(alpha(k)).^(0:N-1);
end
x=real(x);
```

GROUP DELAY 17.33

% Program for computing group delay of a rational transfer function on a given frequency interval

```
function D=grpdly(b,a,K,theta);
b=input ('enter the numerator polynomials=');
a=input ('enter the denominator polynomials=');
K=input ('enter the number of frequency response points=');
theta=input ('enter the theta value=');
a=reshape(a,1,length(a));
b=reshape(b,1,length(b));
if (length(a)==1)%case of FIR
  bd=-j*(0:length(b)-1).*b;
  if(nargin==3),
    B=frqresp(b,1,K);
    Bd=frqresp(bd,1,K);
  else,
    B=frqresp(b,1,K,theta);
    Bd=frqresp(bd,1,K,theta);
  end
  D=(real(Bd).*imag(B)-real(B).*imag(Bd))./abs(B).^2;
else %case of IIR
  if(nargin==3),
    D=grpdly (b,1,K)-grpdly(a,1,K);
  else,
    D=grpdly(b,1,K,theta)-grpdly(a,1,K,theta);
  end
end
```

IIR FILTER DESIGN—IMPULSE INVARIANT METHOD 17.34

% Program for transforming an analog filter into a digital filter using impulse invariant technique

```
function [bout,aout]=impinv(bin,ain,T);
bin=input('enter the numerator polynomials=');
ain=input('enter the denominator polynomials=');
T=input('enter the sampling interval=');
```

```
if(length(bin)=length(ain)),
  error('Anlog filter in IMPINV is not strictly proper');
end
[r,p,k]=residue(bin,ain);
[bout,aout]=pf2tf([],T*r,exp(T*p));
```

IIR FILTER DESIGN—BILINEAR TRANSFORMATION 17.35

% Program for transforming an analog filter into a digial filter using bilinear transformation

```
function [b,a,vout,uout,Cout]=bilin(vin,uin,Cin,T);
pin=input('enter the poles=');
zin=input('enter the zero=');
T=input('enter the sampling interval=');
Cin=input('enter the gain of the analog filter=');
p=length(pin);
q=length(zin);
Cout=Cin*(0.5*T)^(p-q)*prod(1-0.5*T*zin)/prod(1-0.5*T*pin);
zout=[(1+0.5*T*zin)./(1-0.5*T*pin),-ones(1,p-q)];
pout=(1+0.5*T*pin)./(1-0.5*T*pin);
a=1;
b=1;
for k=1 :length(pout),a=conv(a,[1,-pout(k)]); end
for k=1 :length(zout),b=conv(b,[1,-zout(k)]); end
a=real(a);
b=real(Cout*b);
Cout=real(Cout);
```

DIRECT REALISATION OF IIR DIGITAL FILTERS 17.36

% Program for computing direct realisation values of IIR digital filter

```
function y=direct(typ,b,a,x);
x=input('enter the input sequence=');
b=input('enter the numerator polynomials=');
a=input('enter the denominator polynomials=');
typ=input('type of realisation=');
p=length(a)-1;
q=length(b)-1;
pq=max(p,q);
a=a(2:p+1);u=zeros(1,pq);%u is the internal state;
if(typ==1)
  for i=1:length(x),
    unew=x(i)-sum(u(1:p).*a);
    u=[unew,u];
    y(i)=sum(u(1:q+1).*b);
    u=u(1:pq);
  end
elseif(typ==2)
```

```
  for i=1:length(x)
    y(i)=b(1)*x(i)+u(1);
    u=u[(2:pq),0];
    u(1:q)=u(1:q)+b(2:q+1)*x(i);
    u(1:p)=u(1:p)−a*y(i);
  end
end
```

PARALLEL REALISATION OF IIR DIGITAL FILTERS 17.37

% Program for computing parallel realisation values of IIR digital filter

```
function y=parallel(c,nsec,dsec,x);
x=input('enter the input sequence=');
b=input('enter the numerator polynomials=');
a=input('enter the denominator polynomials=');
c=input('enter the gain of the filter=');
[n,m]=size(a);a=a(:,2:3);
u=zeros(n,2);
for i=1:length(x),
  y(i)=c*x(i);
  for k=1:n,
    unew=x(i)−sum(u(k,:).*a(k,:));u(k,:)=[unew,u(k,1)];
    y(i)=y(i)+sum(u(k,:).*b(k,:));
  end
end
```

CASCADE REALISATION OF IIR DIGITAL FILTERS 17.38

% Program for computing cascade realisation values of digital IIR filter

```
function y=cascade(c,nsec,dsec,x);
x=input('enter the input sequence=');
b=input('enter the numerator polynomials=');
a=input('enter the denomiator polynomials=');
c=input('enter the gain of the filter=');
[n,m]=size(b);
a=a(:,2:3);b=b(:,2,:3);
u=zeros(n,2);
for i=1 :length(x),
  for k=1 :n,
    unew=x(i)−sum(u(k,:).*a(k,:));
    x(i)=−unew+sum(u(k,:).*b(k,:))
    u(k,:)=[unew,u(k,1)];
  end
    y(i)=c*x(i);
  end
```

DECIMATION BY POLYPHASE DECOMPOSITION 17.39

% Program for computing convolution and m-fold decimation by polyphase decomposition

```
function y=ppdec(x,h,M);
x=input('enter the input sequence=');
h=input('enter the FIR filter coefficients=');
M=input('enter the decimation factor=');
1h=length(h); 1p=floor((1h−1)/M)+1;
p=reshape([reshape(h,1,1h),zeros(1,1p*M−1h)],M,1p);
lx=length(x); ly=floor ((1x+1h−2)/M)+1;
1u=foor((1x+M−2)/M)+1; %length of decimated sequences
u=[zeros(1,M−1),reshape(x,1,1x),zeros(1,M*lu−lx−M+1)];
y=zeros(1,1u+1p−1);
for m=1:M,y=y+conv(u(m,: ),p(m,: )); end
y=y(1:1y);
```

MULTIBAND FIR FILTER DESIGN 17.40

% Program for the design of multiband FIR filters

```
function h=firdes(N,spec,win);
N=input('enter the length of the filter=');
spec=input('enter the low,high cutoff frequencies and
gain=');
win=input('enter the window length=');
flag=rem(N,2);
[K,m]=size(spec);
n=(0:N)−N/2;
if (~flag),n(N/2+1)=1;
end,h=zeros(1,N+1);
for k=1:K
  temp=(spec (k,3)/pi)*(sin(spec(k,2)*n)−sin(spec(k,1)*n))./n;
  if (~flag);temp(N/2+1)=spec(k,3)*(spec(k,2)−
  spec(k,1))/pi;
  end
  h=h+temp;
end
if (nargin==3),
  h=h.*reshape(win,1,N+1);
end
```

ANALYSIS FILTER BANK 17.41

% Program for maximally decimated uniform DFT analysis filter bank

```
function u=dftanal(x,g,M);
g=input('enter the filter coefficient=');
```

```
x=input('enter the input sequence=');
M=input('enter the decimation factor=');
1g=length(g); 1p=floor((1g−1)/M)+1;
p=reshape([reshape(g,1,1g),zeros(1,1p*M−1g)],M,1p);
lx=length(x); lu=floor ((1x+M−2)/M)+1;
x=[zeros(1,M−1),reshape(x,1,1x),zeros(1,M*lu−lx−M+1)];
x=flipud(reshape(x,M,1u)); %the decimated sequences u=[];
for m=1:M,u=[u; cov(x(m,:),p(m,:))]; end
u=ifft(u);
```

SYNTHESIS FILTER BANK 17.42

% Program for maximally decimated uniform DFT synthesis filter bank

```
function y=udftsynth(v,h,M);
1h=length(h); '1q=floor((1h−1)/M)+1;
q=flipud(reshape([reshape(h,1,1h),zeros(1,1q*M−1h)],M,1q));
v=fft(v);
y=[ ];
for m=1:M,y=[conv(v(m,:),q(m,:));y]; end
y=y(:).';
```

LEVINSON–DURBIN ALGORITHM 17.43

% Program for the solution of normal equations using Levinson-Durbin algorithm

```
function [a,rho,s]=levdur(kappa);
% Input;
% kappa: covariance sequence values from 0 to p
% Output parameters:
% a: AR polynomial,with leading entry 1
% rho set of p reflection coefficients
% s: innovation variance
kappa=input('enter the covariance sequence=');
p=length(kappa)−1;
kappa=reshape(kappa,p+1,1);
a=1; s=kappa(1); rho=[];
for i=1:p,
rhoi=(a*kappa(i+1:−1:2))/s; rho=[rho,rhoi];
s=s*(1−rhoi^2);
a=[a,0]; a=a−rhoi*fliplr(a);
end
```

WIENER EQUATION'S SOLUTION 17.44

% Program

```
function b=wiener(kappax,kappayx);
kappax=input('enter the covariance sequence=');
kappyx=input('enter the joint covariance sequence=');
```

```
q=length(kappax)-1;
kappax=reshape(kappax,q+1,1);
kappayx=reshape(kappayx,q+1,1);
b=(toeplitz(kappax)/(kappayx)';
```

SHORT-TIME SPECTRAL ANALYSIS 17.45

% Program

```
function X=stsa(x,N,K,L,w,opt,M,theta0,dtheta);
x=input('enter the input signal='); L=input('enter the number con-
secutive DFTs to average=');
N=input('enter the segment length='); K=input('enter the number of
overlapping points='); w=input('enter the window coefficients=');
opt=input('opt='); M=input('enter the length of DFT=');
theta0=input('theta0='); dtheta=input('dtheta=');
1x=length(x); nsec=ceil((1x-N)/(N-K)+1;
x=[reshape(x,1,1x),zeros(1,N+(nsec-1)*(N-K))-lx)];
nout=N; if (nargin 5),nout=M; else,opt='n'; end
X=zeros(nsec,nout);
for n=1: nsec,
temp=w.*x((n-1)*(N-K)+1:(n-1)*(N-K)+N);
if (opt(1) == 'z'),temp=[temp,zeros(1,M-N)]; end
if (opt(1)=='c'),temp=chirpf (temp,theta0,dtheta,M);
else,temp=fftshift(fft(temp)); end
X(n,: )=abs(temp).^2;
end
if(L1);
nsecL=floor(nsec/L);
for n=1:nsecL,X(n,:)=mean (X((n-1)*L+1:n*L,:)); end
if (nsec==nsecL*L+1),
X(nsecL+1,:)=X(nsecL*L+1,:); X=X(1:nsecL+1),: );
elseif(nsec nsecL*L),
X(nsecL+1,:)=mean(x(nsecL*L+1:nsec,:));
X=X(1:nsecL+1,:);
else,X=X(1:nsecL,:); end
end
LKh
```

CANCELLATION OF ECHO PRODUCED ON THE TELEPHONE-BASE BAND CHANNEL 17.46

% Simulation program for baseband echo cancellation using LMS algorithm

```
clc; close all; clear all;
format short
T=input('Enter the symbol interval');
```

```
br=input('Enter the bit rate value');
rf=input('Enter the roll off factor');
n=[-10 10];
y=5000*rcosfir(rf,n,br,T); %Transmit filter pulse shape is assumed as
raised cosine
ds=[5 2 5 2 5 2 5 2 5 5 5 5 2 2 2 5 5 5 5]; % data sequence
m=length(ds);
nl=length(y);
i=1;
  z=conv(ds(i),y);
  while(i)
    z1=[z, zeros(1,1.75*br)];
    z=conv(ds(i+1),y);
    z2=[zeros(1,i*1.75*br),z];
    z=z1+z2;
    i=i+1;
  end
  %plot(z); %near end signal
  h=randn(1,length(ds)); %echo path impulse response
  rs1=filter(h,1,z);
  for i=1; length(ds);
    rs(i)=rs 1(i)/15;
  end
  for i=1: round(x3/3),
    rs(i)=randn(1); % rs-echo produced in the hybrid
  end
  fs=[5 5 2 2 2 2 2 5 2 2 2 5 5 5 2 5 2 5 2]; % Desired data signal
  m=length(ds);
  nl=length(y);
  i=1;
    z=conv(fs(i),y);
    while(i)
      z1=[z,zeros(1,1.75*br)];
      z=conv(fs(i+1),y);
      z2=[zeros(1,i*1.75*br),z];
      z=z1+z2;
      i=i+1;
  end
  fs1=rs+fs; % echo added with desired signal
  ar=xcorr(ds,ds);
  crd=xcorr(rs,ds);
  ll=length(ar); j=1;
  for i=round(11/2): 11,
    ar1(j)=ar(i);
    j=j+1;
  end
```

```
r=toeplitz(ar1);
l2=length(crd); j=1;
for i=round(l2/2):l2,
  crdl(j)=crd(i);
  j=j+1;
end
p=crd1';
lam=max(eig(r)); la=min(eig(r)); l=lam/la;
w=inv(r)*p; % Initial value of filter coefficients
e=rs-filter(w,1,ds);
s=1; mu=1.5/lam;
  ni=1;
while (s 1 e-10)
  w1=w-2*mu*(e.*ds)' ; % LMS algorithm adaptation
  rs
  y4=filter(w1,1,ds); % Estimated echo signal using LMS algorithm
  e=y4-rs; s=0; e1=xcorr(e);
  for i=1:length(e1),
    s=s+e1(i);
  end
    s=s/length(e1);
    if (y4==rs)
    break
  end
    ni=ni+1;
    w=w1;
  end
figure(1); subplot(2,2,1); plot(z); title('near end signal');
subplot(2,2,2); plot(rs); title('echo produced in the hybrid');
subplot(2,2,3); plot(fs); title('desired signal');
subplot(2,2,4); plot(fs1); title('echo added with desired signal');
figure(2); subplot(2,1,1); plot(y4); title('estimated echo signal
using LMS algorithm');
subplot(2,1,2); plot(fs1-y4); title('echo cancelled signal');
```

CANCELLATION OF ECHO PRODUCED ON THE TELEPHONE PASS BAND CHANNEL 17.47

% Simulation program for passband echo cancellation using LMS algorithm

```
clc; close all; clear all;
format long
fd=8000; fs=16000; fc=8000;
f=4000; t=0:.01:1; %d=sin(2*pi*f*t/fd);
% Near end signal
```

```
ns=[5 2 5 2 5 5 2 2 2 5 5 2 2 2 2 2 2 5 5 2 5 2 5 5 5 5 5 5 5 5 5 5 5 5 5 5];
% Near end input signal is digitally modulated and plotted
y=dmod(ns,fc,fd,fs,'psk');
subplot(2,2,1); plot(y); title('input signal');
xlabel('Time ——'); ylabel('Amplitude ——');
% Echo is generated due to mismatch of hybrid impedances
h=5*randn(1,length(ns));——
rsl=filter(h,1,y);
% for i=1; length(ns);
% rsl(i)=rs6(i);
% end
for i=1; length(ns);
  rs(i)=rs1(i);
end
subplot(2,2,2); plot(rs); title('noise signal');
xlabel('Time ——'); ylabel('Amplitude ——');
% Far end signal
fs1=[5 5 2 5 2 5 2 5 2 5 2 5 2 5 5 5 5 5 5 5 5 2 2 2 2 2 2 2 2 2 2 2 2 2 2 5];
% rs=sign(rs2);
% Far end signal is digitally modulated and plotted
z1=dmod(fs1,fc,fd,fs,'psk');
for i=1:length(ns),
  z(i)=z1(i);
end
subplot(2,2,3); plot(z); title('far-end signal');
xlabel('Time ——'); ylabel('Amplitude ——');
% Echo and the far end modulated signal is added in the hybrid
q1=z1+rs1;
for i=1; length(ns);
  q(i)=q1(i);;
end
subplot(2,2,4); plot(q); title('received signal');
xlabel('Time ——'); ylabel('Amplitude ——');
q2=xcorr(q);
% Auto correlation is taken for the near end signal
ar=xcorr(ns);
% cross correction is taken for the near end and far end signal
crd=xcorr(rs,ns);
l1=length(ar); j=1;
for i=round(l1/2): l1,
  ar1(j)=ar(i)
  j=j+1;
end
% Toeplitz matrix is taken for the auto correlated signal
```

```
r=toeplitz(ar1);
l2=length(crd); j=1;
for i=round(l2/2):l2,
  crd1(j)=crd(i);
  j=j+1;
end
p=crd1';
% Maximum and minimum eigen values are calculated from the toeplitz
% matrix
lam=max(eig(r)); la=min(eig(r)); l=lam/la;
% initial filter taps are found using the below relation
w=inv(r)*p;
% The step size factor is calculated
m=length(ns)−2.5;
a=(m−.95367)/.274274;
mu=a/lam;
% The initial error is calculated
s=1;
e=rs−filter(w,1,ns); ni=1; figure(2); subplot(2,2,1);
% Filter taps are iterated until the mean squared error becomes
E−25
while ('s25' ! s0')
  w1=w−2*mu*(e.*ns)';
  if (ni=100)
   break;
  end
  rs
  y4=filter(w1,1,ns)
  e=y4−rs; s=0; el=e.*e;
  for i=1: length(e1),
   s=s+e1(i);
  end
   s=s/length(e1);
     ni=ni+1;
     w=w1; plot (ni,e); hold on; title(' MSE vs no. of
iterations');
   end
  end
  subplot(2,2,2); plot(y4); title('estimated noise signal');
  xlabel('Time ——'); ylabel('Amplitude ——');
  subplot(2,2,3); plot(q-y4); title('noise cancelled signal');
  xlabel('Time ——'); ylabel('Amplitude ——');
```

REVIEW QUESTIONS

17.1 (a) Generate unit impulse function

(b) Generate signal $x(n) = u(n) - u(n - N)$

(c) Plot the sequence, $x(n) = A \cos((2\pi f n)/f_s)$, where $n = 0$ to 100

$f_s = 100$ Hz, $f = 1$ Hz, $A = 0.5$

(d) Generate $x(n) = \exp(-5n)$, where $n = 0$ to 10.

17.2 Consider a system with impulse response

$$h(n) = \begin{cases} (1/2)^n, & n = 0 \text{ to } 4 \\ 0, & \text{elsewhere.} \end{cases}$$

17.3 Consider the figure.

(a) Express the overall impulse response in terms of $h_1(n)$, $h_2(n)$, $h_3(n)$, $h_4(n)$

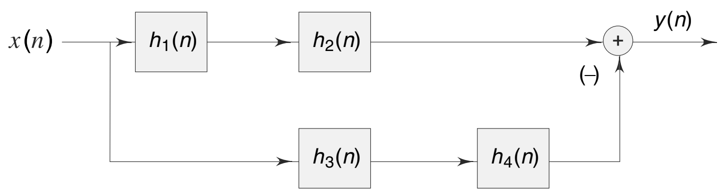


Fig. Q17.3

(b) Determine $h(n)$ when $h_1(n) = \{1/2, 1/4, 1/2\}$

$h_2(n) = h_3(n) = (n + 1)\, u(n)$

$h_4(n) = \delta(n - 2)$

(c) Determine the response of the system in part (b) if $x(n) = \delta(n + 2) + 3\delta(n-1) - 4\delta(n-3)$.

17.4 Compute the overall impulse response of the system

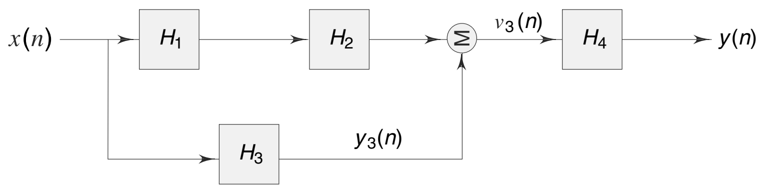


Fig. Q17.4

for $0 \le n \le 99$. The system H_1, H_2, H_3, H_4, are specified by

$H_1 : h_1[n] = \{1, 1/2, 1/4, 1/8, 1/16, 1/32\}$

$H_2 : h_2[n] = \{1, 1, 1, 1, 1\}$

$H_3 : y_3[n] = (1/4)\, x(n) + (1/2)\, x(n - 1) + (1/4)\, x(n - 2)$

$H_4 : y[n] = 0.9\, y(n - 1) - 0.81\, y(n - 2) + V(n) + V(n - 1)$

Plot $h(V)$ for $0 \le n \le 99$.

17.5 Consider the system with $h(n) = a^n u(n)$, $-1 < a < 1$. Determine the response.

$x(n) = u(n + 5) - u(n - 10)$

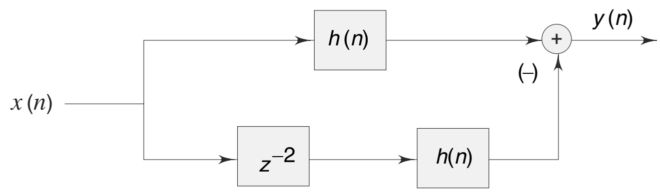


Fig. Q17.5

17.6 (i) Determine the range of values of the parameter '*a*' for which the linear time invariant system with impulse response

$$h(n) = \begin{cases} a^n, & n \geq 0, n \text{ even} \\ 0, & \text{otherwise} \end{cases}$$

is stable.

(ii) Determine the response of the system with impulse response $h(n)=a^n\,u(n)$ to the input signal $x(n)=u(n) - u(n-10)$

17.7 Consider the system described by the difference equation $y(n)=a\,(y+1)+bx\,(n)$. Determine '*b*' in terms of *a* so that $\Sigma\, h(n)=1$.

(a) Compute the zero-state step response $s(n)$ of the system and choose '*b*' so that $s(\infty)=1$.

(b) Compare the values of '*b*' obtained in parts (a) and (b). What did you observe?

17.8 Compute and sketch the convolution $y(n)$ and correlation $r_{xh}(n)$ sequences for the following pair of signals and comment on the results obtained.

$$x_1(n) = \left\{ \underset{\uparrow}{1}, 2, 4 \right\} \qquad h_1(n) = \left\{ \underset{\uparrow}{1}, 1, 1, 1, 1 \right\}$$

$$x_2(n) = \left\{ \underset{\uparrow}{0}, 1, -2, 3, -4 \right\} \qquad h_2(n) = \left\{ 1/2, 1, 2, \underset{\uparrow}{1/2} \right\}$$

$$x_3(n) = \left\{ \underset{\uparrow}{1}, 2, 3, 4 \right\} \qquad h_3(n) = \left\{ \underset{\uparrow}{4}, 3, 2, 1 \right\}$$

$$x_4(n) = \left\{ \underset{\uparrow}{1}, 2, 3, 4 \right\} \qquad h_4(n) = \left\{ \underset{\uparrow}{1}, 2, 3, 4 \right\}$$

17.9 Consider the recursive discrete-time system described by the difference equation.

$$y(n)=a_1 y(n-1) - a_2(n-2) + b_0\,x(n)$$

where $a_1=-0.8$, $a_2=0.64$, $b_0=0.866$

(a) Write a program to compute and plot the impulse response $h(n)$ of the system for $0 \leq n \leq 49$.

(b) Write a program to compute and plot the zero-state step response $s(n)$ of the system for $0 \leq n \leq 100$.

(c) Define an FIR system with impulse response $h_{FIR}(n)$ given by

$$h_{FIR}(n) = \begin{cases} h(n), & 0 \leq n \leq 19 \\ 0, & \text{elsewhere} \end{cases}$$

where $h(n)$ is the impulse response computed in part (a). Write a program to compute and plot its step response.

(d) Compare the results obtained in part (b) and part (c) and explain their similarities and differences.

17.10 Determine and plot the real and imaginary parts and the magnitude and phase spectra of the following DTFT for various values of r and θ

$$G(z)=1/(1-2r(\cos\theta)\,z^{-1}+r^2z^{-2}) \text{ for } 0<r<1.$$

17.11 Using MATLAB program compute the circular convolution of two length-N sequences via the DFT based approach. Using this problem determine the circular convolution of the following pairs of sequences:

(a) $x(n)=\{1, -3, 4, 2, 0, -2\}$ $\quad h(n)=\{3, 0, 1, -1, 2, 1\}$

(b) $x(n)=\{3+j2, -2+j, j3, 1+j4, -3+j3\}$, $\quad h(n)=\{1-j3, 4+j2, 2-j2, -3+j5, 2+j\}$

(c) $x(n)=\cos(\pi n/2)$ $\quad h(n)=3^n \quad 0\le n\le 5$

17.12 Determine the factored form of the following z-transforms

(a) $H_1(z)=(2z^4+16z^3+44z^2+56z+32)/(3z^3+3z^3-15z^2+18z-12)$

(b) $H_2(z)=(4z^4-8.68z^3-17.98z^2+26.74z-8.04)/(z^4-2z^3+10z^2+6z+65)$

and show their pole-zero plots. Determine all regions of convergence of each of the above z-transforms, and describe the type of their inverse z-transform (left-sided, right-sided, two-sided sequences) associated with each of the ROCs.

17.13 Determine the z-transform as a ratio of two polynomials in z^{-1} from each of the partial-fraction expansions listed below:

(a) $H_1(z)=3+\dfrac{12}{(2-z^{-1})}-\dfrac{16}{(4-z^{-1})}, \quad |z|>0.5$

(b) $H_2(z)=3+\dfrac{3}{(1+0.5-z^{-1})}-\dfrac{(4-z^{-1})}{(1+0.25z^{-2})}, \quad |z|>0.5$

(c) $H_3(z)=\dfrac{20}{(5+2-z^{-1})^2}-\dfrac{10}{(5+2z^{-1})}+\dfrac{4}{(1+0.9z^{-2})}, \quad |z|>0.4$

(d) $H_4(z)=8+\dfrac{10}{(5+2z^{-1})}+\dfrac{z^{-1}}{(6+5z^{-1}+z^{-2})}, \quad |z|>0.4$

17.14 Determine the inverse z-transform of each z-transform given in Q17.13.

17.15 Consider the system

$$H(z)=\frac{(1-2z^{-1}+2z^{-2}-z^{-3})}{(1-z^{-1})(1-0.5z^{-1})(1-0.2z^{-1})}, \; ROC\; 0.5<|z|<1$$

(a) Sketch the pole-zero pattern. Is the system stable?

(b) Determine the impulse response of the system.

17.16 Determine the impulse response and the step response of the following causal systems. Plot the pole-zero patterns and determine which of the systems are stable.

(a) $y(n)=\frac{3}{4}y(n-1)-\frac{1}{8}y(n-2)+x(n)$

(b) $y(n)=y(n-1)-0.5y(n-2)+x(n)+x(n-1)$

(c) $H(z)=\frac{z^{-1}(1+z^{-1})}{(1-z^{-3})}$

(d) $y(n)=0.6y(n-1)-0.08y(n-2)+x(n)$

(e) $y(n)=0.7y(n-1)-0.1y(n-2)+2x(n)-x(n-2)$

Ans: (a), (b), (d) and (e) are stable, (c) is unstable

17.17 The frequency analysis of an amplitude-modulated discrete-time signal

$x(n)=\sin 2\pi f_1 n+\sin 2\pi f_2 n$

where $f_1=\frac{1}{128}$ and $f_2=\frac{5}{128}$ modulates the amplitude-modulated signal is $x_c(n)=\sin 2\pi f_c n$ where $f_c=50/128$. The resulting amplitude-modulated signal is $x_{am}(n)=x(n)\sin 2\pi f_c n$

(a) Sketch the signals $x(n)$, $x_c(n)$ and $x_{am}(n)$, $0\le n\le 255$

(b) Compute and sketch the 128-point DFT of the signal $x_{am}(n)$, $0\le n\le 127$

(c) Compute and sketch the 128-point DFT of the signal $x_{am}(n)$, $0\le n\le 99$

(d) Compute and sketch the 256-point DFT of the signal $x_{am}(n)$, $0\le n\le 179$

(e) Explain the results obtained in parts (b) through (d) by deriving the spectrum of the amplitude modulated signal and comparing it with the experimental results.

17.18 A continuous time signal $x_a(t)$ consists of a linear combination of sinusoidal signals of frequencies 300 Hz, 400 Hz, 1.3 kHz, 3.6 kHz and 4.3 kHz. The $x_a(t)$ is sampled at 4 kHz rate and the sampled sequence is passed through an ideal low-pass filter with cut off frequency of 1 kHz, generating a continuous time signal $y_a(t)$. What are the frequency components present in the reconstructed signal $y_a(t)$?

17.19 Design an FIR linear phase, digital filter approximating the ideal frequency response

$$H_d(\omega)=\begin{cases}1, & \text{for } |\omega|\le\pi/6\\ 0, & \text{for } \pi/6<|\omega|\le\pi\end{cases}$$

(a) Determine the coefficient of a 25 tap filter based on the window method with a rectangular window.

(b) Determine and plot the magnitude and phase response of the filter.

(c) Repeat parts (a) and (b) using the Hamming window

(d) Repeat parts (a) and (b) using the Bartlett window.

17.20 Design an FIR Linear Phase, bandstop filter having the ideal frequency response

$$H_d(\omega)=\begin{cases}1, & \text{for } |\omega|\le\pi/6\\ 0, & \text{for } \pi/6<|\omega|\le\pi/3\\ 1, & \text{for } \pi/3\le|\omega|\le\pi\end{cases}$$

(a) Determine the coefficient of a 25 tap filter based on the window method with a rectangular window.

(b) Determine and plot the magnitude and phase response of the filter.

(c) Repeat parts (a) and (b) using the Hamming window
(d) Repeat parts (a) and (b) using the Bartlett window.

17.21 A digital low-pass filter is required to meet the following specifications
Passband ripple ≤ 1 dB
Passband edge 4 kHz
Stopband attenuation ≥ 40 dB
Stopband edge 6 kHz
Sample rate 24 kHz
The filter is to be designed by performing a bilinear transformation on an analog system function. Determine what order Butterworth, Chebyshev and elliptic analog design must be used to meet the specifications in the digital implementation.

17.22 An IIR digital low-pass filter is required to meet the following specifications
Passband ripple ≤ 0.5 dB
Passband edge 1.2 kHz
Stopband attenuation ≥ 40 dB
Stopband edge 2 kHz
Sample rate 8 kHz
Use the design formulas to determine the filter order for

(a) Digital Butterworth filter
(b) Digital Chebyshev filter
(c) Digital elliptic filter

17.23 An analog signal of the form $x_a(t) = a(t)\cos(2000\,\pi t)$ is bandlimited to the range $900 \leq F \leq 1100$ Hz. It is used as an input to the system shown in Fig. Q17.23.

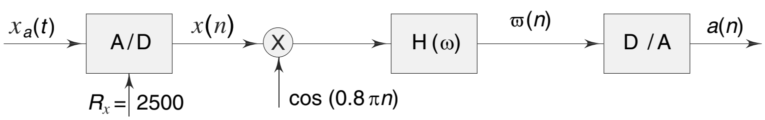


Fig. Q17.23

(a) Determine and sketch the spectra for the signal $x(n)$ and $\omega(n)$.
(b) Use Hamming window of length $M=31$ to design a low-pass linear phase FIR filter $H(\omega)$ that passes $\{a(n)\}$.
(c) Determine the sampling rate of A/D converter that would allow us to eliminate the frequency conversion in the above figure.

17.24 Consider the signal $x(n)=a^n\, u(m)$, $|a| < 1$

(a) Determine the spectrum $X(\omega)$
(b) The signal $x(n)$ is applied to a device (decimator) which reduces the rate by a factor of two. Determine the output spectrum.
(c) Show that the spectrum is simply the Fourier transform of $x(2n)$.

17.25 Design a digital type-I Chebyshev low-pass filter operating at a sampling rate of 44.1 kHz with a passband frequency at 2 kHz, a pass band ripple of 0.4dB, and a minimum stopband attenuation of 50 dB at 12 kHz using the impulse invariance method and the bilinear transformation method. Determine the order of analog filter prototype and design the analog prototype filter. Plot the gain and phase responses of the both designs using MATLAB. Compare the performances of the two filters. Show all steps used in the design.

Hint 1. The order of filter

$$N = \frac{\cosh^{-1}(\sqrt{(A^2-1)}/\varepsilon}{\cosh^{-1}(\Omega s/\Omega p)}$$

2. Use the function cheblap.

17.26 Design a linear phase FIR high-pass filter with following specifications:
Stopband edge at 0.5p, passband edge at 0.7p, maximum passband attenuation of 0.15 dB and a minimum stopband attenuation of 40 dB. Use each of the following windows for the design. Hamming, Hanning, Blackman and Kaiser. Show the impulse response coefficients and plot the gain response of the designed filters for each case.

17.27 Design using the windowed Fourier series approach a linear phase FIR low-pass filter with the following specifications: pass band edge at 1 rad/s, stopband edge at 2 rad/s, maximum passband attenuation of 0.2 dB, minimum stopband attenuation of 50 dB and a sampling frequency of 10 rad/s. Use each of the following windows for the design: Hamming, Hanning, Blackman, Kaiser and Chebyshev. Show the impulse response coefficients and plot the gain response of designed filters for each case.

17.28 Design a two-channel crossover FIR low-pass and high-pass filter pair for digital audio applications. The low-pass and high-pass filters are of length 31 and have a crossover frequency of 2 kHz operating at a sampling rate of 44.1 kHz. Use the function 'fir1' with a Hamming window to design the low-pass filter while the high-pass filter is derived from the low-pass filter using the delay complementary property. Plot the gain responses of both filters on the same figure. What is the minimum number of delays and multipliers needed to implement the crossover network?

17.29 Design a digital network butterworth low-pass filter operating at sampling rate of 44.1 kHz with a 0.5 dB cutoff frequency at 2 kHz and a minimum stopband attenuation of 45 dB at 10 kHz using the impulse invariance method and the bilinear transformation method. Assume the sampling interval for the impulse invariance design to be equal to 1. Determine the order of the analog filter prototype and then design the analog prototype filter. Plot the gain and phase responses of both designs. Compare the performances of the filters. Show all steps used in the design. Does the sampling interval have any effect on the design of the digital filter design based on the impulse invariance method?

Hint The order of filter is

$$N = \frac{\log_{10}(1/k_1)}{\log_{10}(1/k)}$$

and use the function '***buttap***'.

A Appendix

Table A1 *The sinc function*

x	sinc x	sinc^2 x	x	sinc x	sinc^2 x
0.0	1.0	1.0	1.6	−0.18921	0.03580
0.1	0.98363	0.96753	1.7	−0.15148	0.02295
0.2	0.93549	0.87514	1.8	−0.10394	0.01080
0.3	0.85839	0.73684	1.9	−0.05177	0.00268
0.4	0.75683	0.57279	2.0	0	0
0.5	0.63362	0.40528	2.1	0.04684	0.00219
0.6	0.50455	0.25457	2.2	0.08504	0.00723
0.7	0.36788	0.13534	2.3	0.11196	0.01254
0.8	0.23387	0.05470	2.4	0.12614	0.01591
0.9	0.10929	0.01194	2.5	0.12732	0.01621
1.0	0	0	2.6	0.11643	0.01356
1.1	−0.08942	0.00800	2.7	0.09538	0.00910
1.2	−0.15591	0.02431	2.8	0.06682	0.00447
1.3	−0.19809	0.03924	2.9	0.03392	0.00115
1.4	−0.21624	0.04676	3.0	0	0
1.5	−0.21221	0.04503			

B Appendix

IMPORTANT FORMULAE B.1

Trignometric Identities

1. $\varepsilon^{\pm jA} = \cos(A) \pm j \sin(A)$
2. $\left|e^{\pm jA}\right| = 1, \; \underline{|e^{\pm jA}} = \pm A$
3. $\cos(A) = \dfrac{\left(e^{jA} + e^{-jA}\right)}{2}$
4. $\sin(A) = \dfrac{\left(e^{jA} - e^{-jA}\right)}{2j}$
5. $\sin(-A) = -\sin(A)$
6. $\cos(-A) = \cos(A)$
7. $\cos(A \pm B) = \cos(A)\cos(B) \mp \sin(A)\sin(B)$
8. $\sin(A \pm B) = \sin(A)\cos(B) \pm \cos(A)\sin(B)$
9. $\tan(A \pm B) = \dfrac{\tan(A) \pm \tan(B)}{1 \mp \tan(A)\tan(B)}$
10. $\cos(A)\cos(B) = \dfrac{1}{2}[\cos(A - B) + \cos(A + B)]$
11. $\sin(A)\sin(B) = \dfrac{1}{2}[\cos(A - B) - \cos(A + B)]$
12. $\sin(A)\cos(B) = \dfrac{1}{2}[\sin(A + B) + \sin(A - B)]$
13. $\cos(A)\sin(B) = \dfrac{1}{2}[\sin(A + B) + \sin(A - B)]$
14. $\cos^3(A) = \dfrac{1}{4}[3\cos(A) + \cos(3A)]$
15. $\sin^3(A) = \dfrac{1}{4}[3\sin(A) - \sin(3A)]$

16. $\sin^4(A) = \frac{1}{8}[3 - 4\cos(2A) + \cos(4A)]$

17. $\cos^4(A) = \frac{1}{8}[3 + 4\cos(2A) + \cos(4A)]$

18. $\sin(2A) = 2\sin(A)\cos(A) = \dfrac{2\tan(A)}{1+\tan^2(A)}$

19. $\cos(2A) = 2\cos^2(A) - 1 = 1 - 2\sin^2(A) = \cos^2(A) - \sin^2(A) = \dfrac{1-\tan^2(A)}{1+\tan^2(A)}$

20. $\tan(2A) = \dfrac{2\tan(A)}{1-\tan^2(A)}$

21. $A\sin(C) + B\cos(C) = (A^2 + B^2)^{\frac{1}{2}}\cos(C - \tan^{-1}(A/B))$

22. $\cos^2(A) = [1 + \cos(2A)]/2$

23. $\sin^2(A) = [1 - \cos(2A)]/2$

24. $\cos^2(A) + \sin^2(A) = 1$

25. $1 + \tan^2(A) = \sec^2(A)$

26. $1 + \cot^2(A) = \mathrm{cosec}^2(A)$

27. $\sin(3A) = 3\sin(A) - 4\sin^3(A)$

28. $\cos(3A) = 4\cos^3(A) - 3\cos(A)$

29. $\tan(3A) = \dfrac{3\tan(A) - \tan^3(A)}{1 - 3\tan^2(A)}$

30. $\tan(A) \pm \tan(B) = \dfrac{\sin(A \pm B)}{\cos A \cos B}$

31. $\tan^{-1}(A) \pm \tan^{-1}(B) = \tan^{-1}\left[\dfrac{A \pm B}{1 \mp AB}\right]$

32. If $\phi(\omega) = \tan^{-1}[u(\omega)/v(\omega)]$, then $\dfrac{d\phi(\omega)}{d\omega} = \dfrac{v\,du - u\,dv}{u^2 + v^2}$

Finite Summation Formulae

1. $\displaystyle\sum_{k=0}^{n} k = \frac{n(n+1)}{2}$

2. $\displaystyle\sum_{k=0}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}$

3. $\displaystyle\sum_{k=0}^{n} k^3 = \frac{n^2(n+1)^2}{4}$

4. $\displaystyle\sum_{k=0}^{n} a^k = \frac{1-a^{n+1}}{1-a}, \; a \neq 1$

5. $\sum_{k=0}^{n} k\,a^k = \frac{a\left[1-(n+1)a^n + na^{n+1}\right]}{(1-a)^2}$

6. $\sum_{k=0}^{n} k^2\,a^k = \frac{a\left[(1+a)-(n+1)^2 a^n + (2n^2+2n-1)a^{n+1} - n^2 a^{n+2}\right]}{(1-a)^3}$

Infinite Summation Formulae

1. $\sum_{k=0}^{\infty} a^k = \frac{1}{(1-a)},\ |a|<1$

2. $\sum_{k=0}^{\infty} k\,a^k = \frac{a}{(1-a)^2},\ |a|<1$

3. $\sum_{k=0}^{\infty} k^2\,a^k = \frac{a^2+a}{(1-a)^3},\ |a|<1$

SERIES B.2

Taylor Series

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{f^{(n)}(a)}{n!}(x-a)^n + \cdots$$

MacLaurin Series

1. $f(x) = f(0) + \frac{f'(a)}{1!}x + \frac{f''(0)}{2!}x^2 + \cdots + \frac{f^{(n)}(0)}{n!}x^n + \cdots$

   where $f^{(n)}(0) = \left.\frac{\mathrm{d}^n f(x)}{\mathrm{d}x^n}\right|_{x=0}$

2. $f(x+a) = f(x) + a\frac{\mathrm{d}f(x)}{\mathrm{d}x} + \frac{a^2}{2!}\frac{\mathrm{d}^2 f(x)}{\mathrm{d}x^2} + \frac{a^3}{3!}\frac{\mathrm{d}^3 f(x)}{\mathrm{d}x^3} + \cdots$

Binomial Series

1. $(x+y)^n = x^n + nx^{n-1}y + \frac{n(n-1)}{2!}x^{n-2}y^2 + \frac{n(n-1)(n-2)}{3!}x^{n-3}y^3 + \cdots,\ y^2 < x^2$

2. $(1+x)^n = 1 + nx + \frac{n(n-1)}{2!}x^2 + \cdots, |x|<1$

Exponential Series

1. $e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

2. $a^x = 1 + x\ln a + \frac{(x\ln a)^2}{2!} + \frac{(x\ln a)^3}{3!} + \cdots$

Logarithmic Series

1. $\ln(x) = 2\left(\frac{x-1}{x+1}\right) + \frac{2}{3}\left(\frac{x-1}{x+1}\right)^3 + \frac{2}{5}\left(\frac{x-1}{x+1}\right)^5 + \cdots, x > 0$

2. $\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots, |x| < 1$

Trigonometric Series

1. $\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$

2. $\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots$

3. $\tan x = x + \frac{1}{3}x^3 + \frac{2}{15}x^5 + \cdots$

4. $\sin^{-1} x = x + \frac{1}{6}x^3 + \frac{3}{40}x^5 + \cdots$

5. $\tan^{-1} x = x - \frac{1}{3}x^3 + \frac{1}{5}x^5 - \cdots, |x| < 1$

6. (a) $\sin c\ x = \frac{\sin \pi x}{\pi x} = 1 - \frac{1}{3!}(\pi x)^2 + \frac{1}{5!}(\pi x)^4 - \cdots$

   (b) $\sin c\ x = \frac{\sin x}{x} = 1 - \frac{1}{3}x^2 + \frac{1}{5}x^4 - \cdots$

Geometric Series

1. $\frac{1}{1-a} = \sum_{n=0}^{\infty} a^n, \ |a| < 1$

2. $\frac{1-x^n}{1-x} = 1 + x + x^2 + \cdots + x^{n-1}$

Indefinite Integrals

1. $\int x^n \, dx = \frac{x^{n+1}}{(n+1)} + C, \quad n \neq -1$

2. $\int x^{-1} \, dx = \ln(x) + C$

3. $\int a^x \, dx = \frac{a^x}{\ln a} + C$

4. $\int \ln x \, dx = x \ln x - x + C$

5. $\int e^{ax} \, dx = \frac{e^{ax}}{a} + C, \qquad a \neq 0$

6. $\int x^n e^{ax}\, dx = \frac{x^n e^{ax}}{a} - \frac{n}{a}\int x^{(n-1)} e^{ax} dx + C$

7. $\int x e^{ax^2}\, dx = \frac{1}{2a} e^{ax^2} + C$

8. $\int \sin(ax)\, dx = -\frac{\cos(ax)}{a} + C$

9. $\int \cos(ax)\, dx = \frac{\sin(ax)}{a} + C$

10. $\int \tan x\, dx = -\ln \cos x + C$

11. $\int \cot x\, dx = \ln \sin x + C$

12. $\int \sec x\, dx = \ln(\sec x + \tan x) + C = \ln \tan\left(\frac{\pi}{4} + \frac{x}{2}\right) + C$

13. $\int \text{cosec}\, x\, dx = \ln(\text{cosec}\, x - \cot x)\, C = \ln \tan \frac{x}{2} + C$

14. $\int \sec^2 x\, dx = \tan x + C$

15. $\int \text{cosec}^2 x\, dx = -\cot x + C$

16. $\int \sec x \tan x\, dx = \sec x + C$

17. $\int \text{cosec}\, x \cot x\, dx = -\text{cosec}\, x + C$

18. $\int \sin^n x\, dx = -\frac{\sin^{n-1} x \cos x}{n} + \frac{n-1}{n}\int \sin^{n-2} x\, dx$

19. $\int \sin^2(ax)\, dx = \frac{[2ax - \sin(2ax)]}{4a} + C$

20. $\int \cos^2(ax)\, dx = -\frac{[2ax + \sin(2ax)]}{4a} + C$

21. $\int x^n \sin(ax)\, dx = \frac{[-x^n \cos(ax) + n\int x^{n-1} \cos(ax) dx]}{a}$

22. $\int x^n \cos(ax)\, dx = \frac{[x^n \sin(ax) - n\int x^{n-1} \sin(ax) dx]}{a}$

23. $\int \cos(ax)\cos(bx) dx = \frac{\sin[(a-b)x]}{2(a-b)} + \frac{\sin[(a+b)x]}{2(a+b)} + C, \quad a \neq b$

24. $\int \sin(ax)\sin(bx) dx = \frac{\sin[(a-b)x]}{2(a-b)} - \frac{\sin[(a+b)x]}{2(a+b)} + C, \quad a \neq b$

25. $\int \sin(ax)\cos(bx) dx = \frac{\cos[(a-b)x]}{2(a-b)} - \frac{\cos[(a+b)x]}{2(a+b)} + C, \quad a \neq b$

26. $\int e^{ax} \sin(bx)\, dx = \frac{e^{ax}[a \sin[(bx) - b \cos(bx)]}{(a^2 + b^2)} + C$

27. $\int e^{ax}\cos(bx)\,dx = \dfrac{e^{ax}[a\cos[(bx)+b\sin(bx)]}{(a^2+b^2)}+C$

28. $\int \dfrac{dx}{a+bx} = \dfrac{1}{b}\ln(a+bx)+C$

29. $\int (a+bx)^n\,dx = \dfrac{(a+bx)^{n+1}}{b(n+1)}+C, \qquad n \neq -1$

30. $\int \dfrac{dx}{a^2+b^2x^2} = \dfrac{1}{ab}\tan^{-1}\left(\dfrac{bx}{a}\right)+C$

31. $\int \dfrac{dx}{a^2-x^2} = \dfrac{1}{2b}\ln\dfrac{a+x}{a-x}+C \quad \text{or} \quad \dfrac{1}{2a}\ln\dfrac{x+a}{x-a}+C$

32. $\int \dfrac{x^2dx}{a^2+b^2x^2} = \dfrac{x}{b^2}-\dfrac{a}{b^3}\tan^{-1}\left(\dfrac{bx}{a}\right)+C$

33. $\int \sqrt{a+bx}\,dx = \dfrac{2}{3b}\sqrt{(a+bx)^3}+C$

34. $\int x\sqrt{a+bx}\,dx = -\dfrac{2(2a-3bx)\sqrt{(a+bx)^3}}{15b^2}+C$

35. $\int \sqrt{x^2+a^2}\,dx = \dfrac{x}{2}\sqrt{x^2+a^2}+\dfrac{a^2}{2}\ln\left(x+\sqrt{x^2+a^2}\right)+C$

36. $\int x\sqrt{x^2+a^2}\,dx = \dfrac{\sqrt{(x^2+a^2)^3}}{3}+C$

37. $\int \dfrac{dx}{\sqrt{x^2+a^2}} = \ln\left(x+\sqrt{x^2+a^2}\right)+C$

38. $\int \dfrac{x\,dx}{\sqrt{x^2+a^2}} = \sqrt{x^2+a^2}+C$

39. $\int \dfrac{dx}{\left(\sqrt{a^2-x^2}\right)} = \sin^{-1}\left(\dfrac{x}{a}\right)+C$

40. $\int \dfrac{x\,dx}{\sqrt{a^2-x^2}} = -\sqrt{a^2-x^2}+C$

41. $\int \sqrt{a^2-x^2}\,dx = \dfrac{x}{2}\sqrt{a^2-x^2}+\dfrac{a^2}{2}\sin^{-1}\left(\dfrac{x}{a}\right)+C$

42. $\int x\sqrt{a^2-x^2}\,dx = \dfrac{\sqrt{(a^2-x^2)^3}}{3}+C$

43. $\int \dfrac{dx}{\sqrt{x^2-a^2}} = \ln\left(x+\sqrt{x^2-a^2}\right)+C$

44. $\int \dfrac{x\,dx}{\sqrt{x^2-a^2}} = \sqrt{x^2-a^2}+C$

45. $\int \sqrt{x^2-a^2}\ \mathrm{d}x = \frac{x}{2}\sqrt{x^2-a^2} - \frac{a^2}{2}\ln\left(x+\sqrt{x^2-a^2}\right)+C$

46. $\int x\sqrt{x^2-a^2}\ \mathrm{d}x = \frac{\sqrt{\left(x^2-a^2\right)}}{3}+C$

47. $\int \frac{\mathrm{d}x}{x\sqrt{x^2-a^2}} = \frac{1}{a}\sec^{-1}\left(\frac{x}{a}\right)+C$

48. $\int \frac{\sqrt{x^2-a^2}}{x}\mathrm{d}x = \sqrt{x^2-a^2} - a\cos^{-1}\left(\frac{x}{a}\right)+C$

49. $\int \sqrt{\frac{a+x}{b+x}}\ \mathrm{d}x = \sqrt{(a+x)(b+x)}+(a-b)\ln\left(\sqrt{a+x}+\sqrt{b+x}\right)+C$

50. $\int \sqrt{\frac{a+x}{b-x}}\ \mathrm{d}x = -\sqrt{(a+x)(b-x)}-(a+b)\sin^{-1}\sqrt{\frac{b-x}{a+b}}+C$

51. $\int \frac{\mathrm{d}x}{\sqrt{(x-a)(b-x)}} = 2\sin^{-1}\sqrt{\frac{x-a}{b-a}}+C$

52. $\int f(x)\left[\frac{\mathrm{d}g(x)}{\mathrm{d}x}\right]\mathrm{d}x = f(x)\ g(x) - \int g(x)\left[\frac{\mathrm{d}f(x)}{\mathrm{d}x}\right]\mathrm{d}x$

Definite Integrals

1. $\int_0^\infty \left[\frac{a}{a^2+x^2}\right]\mathrm{d}x = \frac{\pi}{2}, a>0$

2. $\int_0^\infty x^n e^{-ax}\ \mathrm{d}x = \frac{n!}{a^{n+1}}, a>0$, and n is a positive integer

3. $\int_0^\infty x^2 e^{-ax^2}\ \mathrm{d}x = \frac{1}{4a}\sqrt{\frac{\pi}{a}},\quad a>0$

4. $\int_0^\infty e^{-a^2x^2}\ \mathrm{d}x = \frac{\sqrt{\pi}}{2a}, a>0$

5. $\int_0^\infty \sin^2(nx)\ \mathrm{d}x = \int_0^\pi \cos^2(nx)\ \mathrm{d}x = \pi/2,\ n$ is an integer

6. $\int_0^\infty \sin c(ax)\ \mathrm{d}x = \frac{1}{2a},\quad a>0$

7. $\int_0^\infty \sin c^2(ax)\ \mathrm{d}x = \frac{1}{2a},\quad a>0$

8. $$\int_0^{\infty} e^{-ax}\cos(bx)\,dx = \frac{a}{a^2+b^2}, \quad a>0$$

9. $$\int_0^{\infty} e^{-ax}\sin(bx)\,dx = \frac{b}{a^2+b^2}, \quad a>0$$

10. $$\int_0^{\infty} \frac{x\sin(ax)}{b^2+x^2}\,dx = \frac{\pi}{2}e^{-ab}, \quad a>0 \text{ and } b>0$$

11. $$\int_0^{\infty} \frac{\sin(ax)}{b^2+x^2}\,dx = \frac{\pi}{2b}e^{-ab}, \quad a>0 \text{ and } b>0$$

12. $$\int_0^{\infty} \frac{\cos(ax)}{b^2-x^2}\,dx = \frac{\pi}{4b^3}[\sin(ab)-ab\cos(ab)], \quad a>0 \text{ and } b>0$$

13. $$\int_0^{\pi/2} \sin^n(x)\,dx = \int_0^{\pi/2} \cos^n(x)\,dx = \begin{cases} \dfrac{(1)(3)\cdots(n-1)(\pi)}{(2)(4)\cdots(n)(2)}, & n \text{ is an even integer}, \quad n \neq 0 \\ \dfrac{(2)(4)\cdots(n-1)}{(1)(3)\cdots(n)}, & n \text{ is an odd integer}, \quad n \neq 1 \end{cases}$$

14. $$\int_0^{\pi} \sin(nx)\sin(mx)\,dx = \int_0^{\pi} \cos(nx)\cos(mx)\,dx = 0, \; n \text{ and } m \text{ are unequal integers.}$$

15. $$\int_0^{\pi} \sin(nx)\cos(mx)\,dx = \begin{cases} 2n/(n^2-m^2), & n-m \text{ is an odd integer} \\ 0, & n-m \text{ is an even integer} \end{cases}$$

16. $$\int_0^{\pi/2} \log\sin x\,dx = -\frac{\pi}{2}\log 2$$

17. $$\int_0^{a} f(x)\,dx = \int_0^{a} f(a-x)\,dx$$

18. $$\int_a^{b} f(x)\left[\frac{dg(x)}{dx}\right]dx = f(x)\,g(x)\Big|_a^b - \int_0^{b} g(x)\left[\frac{df(x)}{dx}\right]dx$$

C Appendix

C.1 BUTTERWORTH POLYNOMIALS

N	Factors	Polynomial
1	$s + 1$	$s + 1$
2	$s^2 + \sqrt{2s} + 1$	$s^2 + \sqrt{2s} + 1$
3	$(s + 1), (s^2 + s + 1)$	$s^3 + 2s^2 + 2s + 1$
4	$(s^2 + 0.765\ s + 1), (s^2 + 1.848s + 1)$	$s^4 + 2.613s^2 + 3.414s^2 + 2.613\ s + 1$
5	$(s + 1), (s^2 + 0.618s + 1), (s^2 + 1.618s + 1)$	$s^5 + 3.236s^4 + 5.236s^3 + 5.236s^2 + 3.236s + 1$
6	$(s^2 + 0.518s + 1), (s^2 + \sqrt{2s} + 1), (s^2 + 1.932s + 1)$	$s^6 + 3.864s^5 + 7.464s^4 + 9.142s^3 + 7.464s^2 + 3.864s + 1$
7	$(s + 1), (s^2 + 0.44s + 1), (s^2 + 1.247s + 1),$ $(s^2 + 1.802s + 1)$	$s^7 + 4.494s^6 + 10.098s^5 + 14.592s^4 + 14.592s^3 +$ $10.098s^2 + 4.494s + 1$
8	$(s^2 + 0.3s + 1), (s^2 + 1.111s + 1), (s^2 + 1.166s +$ $1), (s^2 + 1.962s + 1)$	$s^8 + 5.126s^7 + 13.137s^6 + 21.846s^5 + 25.688s^4 +$ $21.846s^3 + 13.137s^2 + 5.126s + 1$
9	$(s + 1), (s^2 + 0.347s + 1), (s^2 + s + 1),$ $(s^2 + 1.532s + 1), (s^2 + 1.879s + 1)$	$s^9 + 5.759s^8 + 16.582s^7 + 31.163s^6 + 41.986s^5 +$ $41.986s^4 + 31.163s^3 + 16.582s^2 + 5.757s + 1$
10	$(s^2 + 0.313s + 1), (s^2 + 0.908s + 1), (s^2 + \sqrt{2s}$ $+1), (s^2 + 1.792s + 1), (s^2 + 1.975s + 1)$	$s^{10} + 6.393s^9 + 20.432s^8 + 42.802s^7 + 64.882s^6 +$ $74.233s^5 + 64.882s^4 + 42.802s^3 + 20.432s^2 + 6.393s + 1$

C.2 CHEBYSHEV POLYNOMIALS

N	$C_N(\omega)$
0	1
1	ω
2	$2\omega^2 - 1$
3	$4\omega^3 - 3\omega$
4	$8\omega^4 - 8\omega^2 + 1$
5	$16\omega^5 - 20\omega^3 + 5\omega$
6	$32\omega^6 - 48\omega^4 + 18\omega^2 - 1$
7	$64\omega^7 - 112\omega^5 + 56\omega^2 - 7\omega$
8	$128\omega^8 - 256\omega^6 + 160\omega^4 - 32\omega^2 + 1$
9	$256\omega^9 - 576\omega^7 + 432\omega^5 - 120\omega^3 + 9\omega$
10	$512\omega^{10} - 1280\omega^8 + 1120\omega^6 - 400\omega^4 + 50\omega^2 - 1$

D Appendix

D.1 SPECIFICATION AND IMPULSE RESPONSE FOR FIR FILTER DESIGN BY FOURIER SERIES METHOD

Type of filter	Specification	Impulse Response
Lowpass filter	$H_d\left(e^{j\omega}\right)=\begin{cases}1 & , -\omega_c \le \omega \le \omega_c \\ 0 & , -\pi \le \omega < -\omega_c \\ 0 & , \omega_c < \omega \le \pi\end{cases}$	$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega=\frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $\left[\text{since } H_d\left(e^{j\omega}\right)=0 \text{ for } -\pi \le \omega < -\omega_c \text{ and } \omega_c < \omega \le \pi\right]$
Highpass filter	$H_d\left(e^{j\omega}\right)=\begin{cases}1 & , -\pi \le \omega \le -\omega_c \\ 1 & , \omega_c \le \omega \le \pi \\ 0 & , -\omega_c < \omega < \omega_c\end{cases}$	$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{+\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega=\frac{1}{2\pi}\int_{-\pi}^{-\omega_c} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $+\frac{1}{2}\int_{\omega_c}^{\pi} e^{-j\omega\alpha}e^{-j\omega n}d\omega$ $\left[\text{since } H_d\left(e^{j\omega}\right)=0 \text{ for } -\omega_c < \omega < \omega_c\right]$
Bandpass filter	$H_d\left(e^{j\omega}\right)=\begin{cases}1 & , -\omega_{c2} \le \omega \le -\omega_{c1} \\ 1 & , \omega_{c1} \le \omega \le \omega_{c2} \\ 0 & , -\pi \le \omega < -\omega_{c2} \\ 0 & , -\omega_{c1} < \omega < \omega_{c1} \\ 0 & , \omega_{c2} < \omega \le \pi\end{cases}$	$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{+\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega=\frac{1}{2\pi}\int_{-\omega_{c2}}^{-\omega_{c1}} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $+\frac{1}{2}\int_{\omega_{c1}}^{\omega_{c2}} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $\left[\text{since } H_d\left(e^{j\omega}\right)=0 \text{ for } -\pi \le \omega < -\omega_{c2};\ -\omega_{c1} < \omega < \omega_{c1} \text{ and } \omega_{c2} < \omega \le \pi\right]$
Bandpass filter	$H_d\left(e^{j\omega}\right)=\begin{cases}1 & , -\pi \le \omega \le -\omega_{c2} \\ 1 & , -\omega_{c1} \le \omega \le \omega_{c1} \\ 1 & , \omega_{c2} \le \omega \le \pi \\ 0 & , -\omega_{c2} < \omega < -\omega_{c1} \\ 0 & , \omega_{c1} < \omega < \omega_{c2}\end{cases}$	$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{+\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega=\frac{1}{2\pi}\int_{-\pi}^{-\omega_{c2}} e^{j\omega\alpha}e^{j\omega n}d\omega$ $+\frac{1}{2}\int_{-\omega_{c1}}^{+\omega_{c1}} e^{-j\omega\alpha}e^{j\omega n}d\omega+\frac{1}{2\pi}\int_{\omega_{c2}}^{\pi} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $\left[\text{since } H_d\left(e^{j\omega}\right)=0 \text{ for } -\omega_{c2} < \omega < -\omega_{c1} \text{ and } \omega_{c1} < \omega < \omega_{c2}\right]$

D.2 NORMALIZED IDEAL FREQUENCY RESPONSE AND IMPULSE RESPONSE FOR FIR FILTER DESIGN USING WINDOWS

Type of filter	Specification	Impulse Response
Lowpass filter	$H_d\left(e^{j\omega}\right)=\begin{cases} e^{-j\omega\alpha} & , -\omega_c \le \omega \le \omega_c \\ 0 & , -\pi \le \omega < -\omega_c \\ 0 & , \omega_c < \omega \le \pi \end{cases}$	$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega=\frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $\left[\text{since } H_d\left(e^{j\omega}\right)=0 \text{ for } -\pi \le \omega < -\omega_c \text{ and } \omega_c < \omega \le \pi\right]$
Highpass filter	$H_d\left(e^{j\omega}\right)=\begin{cases} e^{-j\omega\alpha} & , -\pi \le \omega \le -\omega_c \\ e^{-j\omega\alpha} & , \omega_c \le \omega \le \pi \\ 0 & , -\omega_c < \omega < \omega_c \end{cases}$	$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{+\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega=\frac{1}{2\pi}\int_{-\pi}^{-\omega_c} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $+\frac{1}{2}\int_{\omega_c}^{\pi} e^{-j\omega\alpha}e^{-j\omega n}d\omega$ $\left[\text{since } H_d\left(e^{j\omega}\right)=0 \text{ for } -\omega_c < \omega < \omega_c\right]$
Bandpass filter	$H_d\left(e^{j\omega}\right)=\begin{cases} e^{-j\omega\alpha} & , -\omega_{c2} \le \omega \le -\omega_{c1} \\ e^{-j\omega\alpha} & , \omega_{c1} \le \omega \le \omega_{c2} \\ 0 & , -\pi \le \omega < -\omega_{c2} \\ 0 & , -\omega_{c1} < \omega < \omega_{c1} \\ 0 & , \omega_{c2} < \omega \le \pi \end{cases}$	$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{+\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega=\frac{1}{2\pi}\int_{-\omega_{c2}}^{-\omega_{c1}} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $+\frac{1}{2}\int_{\omega_{c1}}^{\omega_{c2}} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $\left[\text{since } H_d\left(e^{j\omega}\right)=0 \text{ for } -\pi \le \omega < -\omega_{c2};\ -\omega_{c1} < \omega < \omega_{c1} \text{ and } \omega_{c2} < \omega \le \pi\right]$
Bandpass filter	$H_d\left(e^{j\omega}\right)=\begin{cases} e^{-j\omega\alpha} & , -\pi \le \omega \le -\omega_{c2} \\ e^{-j\omega\alpha} & , -\omega_{c1} \le \omega \le \omega_{c1} \\ e^{-j\omega\alpha} & , \omega_{c2} \le \omega \le \pi \\ 0 & , -\omega_{c2} < \omega < -\omega_{c1} \\ 0 & , \omega_{c1} < \omega < \omega_{c2} \end{cases}$	$h_d(n)=\frac{1}{2\pi}\int_{-\pi}^{+\pi} H_d\left(e^{j\omega}\right)e^{j\omega n}d\omega=\frac{1}{2\pi}\int_{-\pi}^{-\omega_{c2}} e^{j\omega\alpha}e^{j\omega n}d\omega$ $+\frac{1}{2}\int_{-\omega_{c1}}^{+\omega_{c1}} e^{-j\omega\alpha}e^{j\omega n}d\omega+\frac{1}{2\pi}\int_{\omega_{c2}}^{\pi} e^{-j\omega\alpha}e^{j\omega n}d\omega$ $\left[\text{since } H_d\left(e^{j\omega}\right)=0 \text{ for } -\omega_{c2} < \omega < -\omega_{c1} \text{ and } \omega_{c1} < \omega < \omega_{c2}\right]$

Index