Digital Communication

About the Author

T L Singal is currently Professor in the Department of Electronics and Communication Engineering, School of Electronics and Electrical Engineering, Chitkara University, Punjab. He obtained his Bachelor of Engineering degree with honours in Electronics and Communication Engineering from National Institute of Technology (NIT), Kurukshetra, in 1981. He pursued Master of Technology in Electronics and Communication Engineering from Punjab Technical University, Jalandhar. During 1981–2002, he worked in the field of wireless and cellular communications with leading telecom organizations in India, Germany, and USA. He managed various VHF/UHF wireless communication projects in reputed companies like HAL, Hyderabad, and PSIDC, Chandigarh. He visited Germany on business trips for technical know-how transfer of Multi-Access Wireless Communication Systems during 1990–92. He has also worked as Senior Network Consultant with Flextronics Network Services, Dallas (Texas), USA, and offered optimization solutions in the domain of GSM and CDMA 2G/3G cellular technologies for various clients such as Nokia, AT&T, Nortel, Cingular Wireless, Voice Stream, USA, during 2000–2002. Since 2003, he has been working as Senior Faculty with Punjab Technical University and Chitkara University.

Prof Singal's areas of interests include modeling and performance analysis of digital communication systems, seamless mobility in future-generation digital cellular networks, and convergence of wireless and internet technologies. He has convened international conferences and workshops on Wireless Networks and Embedded Systems, and edited proceedings of conferences. He has presented several technical research papers in international journals and conferences.

Digital Communication

T L Singal

Professor
Department of Electronics and Communication Engineering
School of Electronics and Electrical Engineering
Chitkara University
Rajpura, Punjab

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 110 016

Digital Communication

Print Edition
ISBN 13: 978-93-392-1952-9
ISBN 10: 93-392-1952-X

E-Book Edition
ISBN 13: 978-93-392-1953-6
ISBN 10: 93-392-1953-8

Managing Director: *Kaushik Bellani*
Head—Products (Higher Education & Professional): *Vibha Mahajan*

Assistant Sponsoring Editor: *Koyel Ghosh*
Senior Editorial Researcher: *Sachin Kumar*
Editorial Executive: *Piyali Chatterjee*

Manager—Production Systems: *Satinder S Baveja*
Assistant Manager—Editorial Services: *Sohini Mukherjee*
Senior Production Manager: *P L Pandita*
Senior Graphic Designer—Cover: *Meenu Raghav*

Senior Publishing Manager: *Shalini Jha*
Assistant Product Manager: *Tina Jajoriya*

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

Typeset at Text-o-Graphics, B-1/56, Aravali Apartment, Sector-34, Noida 201 301, and printed at

Cover Printer:

Visit us at: www.mheducation.co.in

Dedicated to

My respected ***didi*** *and* ***jijaji***

Smt Tirthi Rani *and* ***Shri Mukand Lal Goyal***

Contents

Preface

Introduction to the Course

Given below are some phenomena and instances that form a part of our everyday lives:

1. Signal jamming
2. Digital cellular mobile phones
3. HDTV transmissions

What is common amongst the above?—Principles of Digital Communication!!

Digital communication involves transmission of either digitized analog signals or digital data stream using baseband or bandpass communication channels. Recently, we have seen exponential growth in the usage of digital communication technologies in digital cellular mobile phones and high-definition TV transmissions to web services. The baseband transmission deals with representation of continuously varying analog data by encoded data sequences, followed by suitable voltage waveforms by means of line codes. Several baseband digital signals can be multiplexed and sent over a broadband channel. Digital data is modulated using high-frequency analog carrier signals for bandpass transmissions via wireless communication channels. Source-coding and channel-coding techniques are employed for efficient and reliable digital transmissions. For increased immunity to natural noise, interference and signal jamming, and for providing secured digital transmission, spread-spectrum methods are used in communications.

Target Audience

This book is primarily intended as a textbook for undergraduate as well as postgraduate students of Electronics and Communication Engineering. The Digital Communication course is compulsory in the third year of BTech (ECE) and first year of MTech (ECE) as Advanced Digital Communication. The contents of the book have been selected from the related curriculum prescribed in reputed technical universities in India. This book will be useful for

- All UG and PG engineering students of Electronics and Communication Engineering, Electrical and Electronics Engineering, and Telecommunication Engineering
- All students preparing for IETE and AMIE examinations
- All aspirants of various competitive examinations including GATE
- All candidates appearing for UPSC, Indian Engineering Services (IES), PSUs, and other technical interviews by telecom industries
- Practicing professionals in the field of telecommunications, as a reference guide

Objective of this Book

In order to prepare students to ace a course on Digital Communication or its examinations, many a literature has been made available. During my interactions with undergraduate and postgraduate students of Electronics and Communication Engineering for several years, I have felt the need to convert the words penned down in the books, articles, journals, etc., into a more interactive and industry-design oriented format. With an objective of presenting theory in a crisp and easy-to-understand manner along with rich pedagogical features, the content in this book will enable the students to

- Learn the fundamental concepts and acquire competencies for each topic,
- Apply theoretical concepts to solve easy as well as difficult problems, and
- Test and analyze the technical skills acquired.

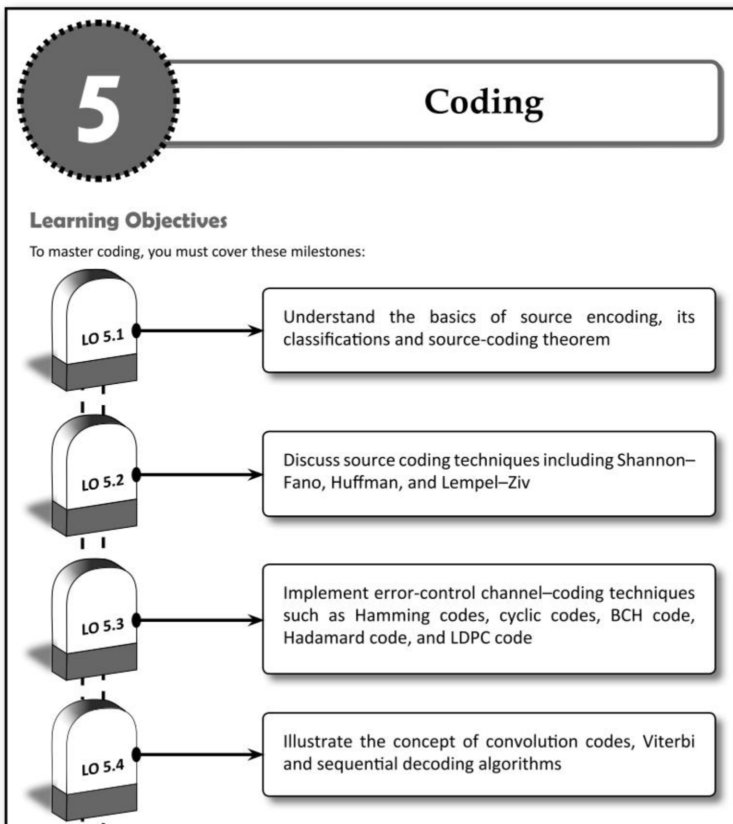
Learning Tools

In designing this book, we have focused on the Learning Objective (LO) oriented approach. This is an educational process that emphasizes on developing engineering skill in the student, and testing the outcomes of the study of a course, as opposed to rote learning. We believe it's not *what is being taught*, rather it's *what is being learnt* that is the key to the student's overall development. This approach creates an ability to acquire knowledge and apply fundamental principles to analytical problems and applications.

Each of the 6 chapters follows a common structure with a range of learning and assessment tools for instructors and students.

- Learning Objectives

Each chapter begins with a set of Learning Objectives that are directly tied to the chapter content. These help students better anticipate what they will be studying and help instructors measure a student's understanding. The sections of the chapters are structured in a modular way, which help in systematic concept development.



- **Arrangement of Pedagogy**

The pedagogy is arranged as per levels of difficulty to help students better assess their learning. This assessment of the levels of difficulty is derived from the Bloom's Taxonomy.

○○● indicates Level 1 and Level 2 i.e., Knowledge and Comprehension based easy-to-solve problems.

○●● indicates Level 3 and Level 4 i.e., Application and Analysis based medium-difficulty problems.

●●● indicates Level 5 and Level 6 i.e., Synthesis and Evaluation based high-difficulty problems.

Self-Assessment Exercise linked to LO 6.1

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/160

Q6.1.1 What do you understand by spread-spectrum modulation? Mention the primary advantage of spread-spectrum communication. ○○●

Q6.1.2 Give an example of the benefit of using a modulation carrier bandwidth significantly wider than the baseband bandwidth. ○●●

Q6.1.3 Compare and contrast between wideband FM and spread-spectrum signal. ○●●

Q6.1.4 Calculate the processing gain if the information bit rate is 1500 bps and the bandwidth of spread-spectrum signal is 21.5 MHz. ○●●

Q6.1.5 What is meant by chips of the PN sequence? List different techniques of PN sequences. ○○●

Q6.1.6 Describe briefly Balance property and Run property in uniform distribution as criterion to validate a PN sequence. ○●●

Q6.1.7 Design the Gold sequence for two *m*-sequences given as
m-sequence 1: 10011001110101010100101001011100
m-sequence 2: 11001001001100111011010101010 ●●●

Note ○○● Level 1 and Level 2 Category
○●● Level 3 and Level 4 Category
●●● Level 5 and Level 6 Category

- **Use of Technology**

In bringing out this edition, we have taken advantage of recent technological developments to create a wealth of useful information not present in the physical book. For students using smartphones and tablets, scanning **QR codes** located within the chapters gives them immediate access to more resources like **Chapter 0: Preliminaries** and **Answers to**

- Self-Assessment Exercises
- Mid-Chapter Checks
- Objective-Type Questions
- Short-Answer Type Questions
- Discussion Questions
- Problems

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/47

- **Highlighting Useful Information through Special Features**

Special features such as *Important*, *Good to Know!*, *Attention*, etc., are inserted throughout the text to draw attention to important anecdotes.

Question *How can you convert digital data, usually available in the form of 0s and 1s, to digital signals that must represent digital data in the form of appropriate voltage levels on a wireline?*

Solution Probably the simplest process is ***line coding***, also known as ***transmission coding***, or ***digital baseband signaling***. You will find the details of line codes and its types in Section 2.1.

Good to Know!
- There are *several* line codes available for electrical representation of binary data stream. These line codes differ from one another with respect of desirable *line properties* which are necessary for efficient transmi
- Due to dispersive nature of the c pulse. This results into *Intersymbo* in the reconstructed data stream at

to represent the symbols 1 and 0, respectively, and there is no 0 V level. Accordingly, there is only polar NRZ-L line coding technique, not polar RZ.

Differential line-coding techniques such as polar NRZ-M (M stands for Mark), a binary data 1 (or mark) is represented by a change in voltage level from its previously held voltage level, and a binary data 0 is represented by no change in voltage level from its previous one (that is, it remains same). Polar NRZ-M is primarily used in magnetic tape recording. Polar NRZ-S (S stands for Space) is complement of NRZ-M. **ATTENTION**

In *Manchester polar line code*, a binary symbol 0 is represented by a –*V* pulse during first half of the bit period followed by a +*V* pulse during the second half of the bit period, and a binary symbol 1 is represented by a +*V* pulse during the first half of the bit period followed by a –*V*

Type 4: Manchester Polar Line Code

- Learning Outcomes and Key Concepts

Learning Outcomes and Key Concepts at the end of each chapter help reconnect the ideas initiated, with the outcome achieved.

Key Concepts

- alternate mark inversion (AMI)
- asynchronous TDM
- bipolar code
- framing bits
- E-lines
- eye diagram
- interleaving
- line code
- Manchester line code
- multiplexing
- non-return-to-zero (NRZ)
- polar line code
- polar NRZ
- return-to-zero (RZ) code
- synchronous transmission
- time-division multiplexing (TDM)
- T-lines
- unipolar code

Learning Outcomes

- Information theory is a branch of probability theory, which can be applied to the study of communication systems.
- Information theory deals with the measure for an amount of information and the means to apply it to improve the communication of information.
- The probability of an event and the amount of information associated with it are inversely related to each other.

LO 4.1

- The source may be described in terms of average amount of information per individual message, known as entropy of the source.
- The entropy of a source is a function of the probabilities of the source symbols that constitute the alphabet of the source.
- Since the entropy is a measure of uncertainty, the entropy is maximum when the associated probability distribution generates maximum uncertainty.
- It is true that average information is maximum when all the messages are equally likely.

LO 4.2

- A binary symmetric channel is the simplest form of a discrete memoryless channel.
- A binary symmetric channel is symmetric because the probability of receiving a binary logic 1 if a 0 is sent is exactly the same as the probability of receiving a binary logic 0 if a 1 is sent.

LO 4.3

- The channel capacity represents the maximum rate at which data can be transferred between transmitter and receiver, with an arbitrarily small probability of error.
- When the system operates at a rate greater than C, it is liable to incur high probability of error.

LO 4.4

- MATLAB

MATLAB exercises are provided chapter-wise through QR codes and URL.

MATLAB simulation exercises on *M*-ary digital modulation techniques,

Scan the QR code given here OR visit: http://qrcode.flipick.com/index.php/129

ATTENTION QPSK and its variants are the most widely used digital modulation technique in modern digital wireless communication systems. This topic is important from examination point of view too.

3.2.4 Minimum Shift Keying (MSK)

Recall Coherent binary FSK receivers use the phase information contained in the received signal only for synchronization, not for improved detection.

Problem-Solving and Critical Thinking

- Within-The-Chapter Assessment Tools

This is by far the best feature! Pedagogy has been arranged within the text and linked after every learning objective. Great retention-through-looping mechanism!

SOLVED EXAMPLE 6.2.2 **Number of Frequencies in FHSS**

Calculate the minimum number of frequencies required for a frequency-hopping spread-spectrum communication system if the frequency multiplication factor is 7.

Solution Let there are $N = 2^k$ number of different frequencies which form 2^k number of distinct channels, where k is the frequency multiplication factor or the number of codes in PN code generator in an FHSS communication system. That is,

For given $k = 7$, $N = 2^7 = 128$ **Ans.**

SOLVED EXAMPLE 6.2.3 **Fast-Frequency Hopping**

A frequency-hopping spread-spectrum communication system utilizes fast-hop technique at the hop rate of 10 hops per information bit. If the information bit rate is 2800 bps, what is the frequency separation?

Solution In a frequency-hopping spread-spectrum communication system,

Frequency separation = hop rate × bit rate

For given hop rate = 10 hops per information bit and information bit rate = 2800 bps,

Frequency separation = 10 × 2800 = 28000 Hz or 28 kHz **Ans.**

SOLVED EXAMPLE 6.2.4 **Concept of Slow-Frequency Hopping**

...opping system, long data packets are transmitted over the wireless channel. ...ence of frequencies programmed in PN code generator is f_3, f_5, f_6, f_1, f_4, ...turning to the first frequency, f_3. Draw a suitable diagram to illustrate the

Mid-Chapter Check

So far you have learnt the following:

- Principles of Spread-Spectrum Modulation
- PN Sequences and their Properties
- Frequency Hopping Spread-Spectrum (FHSS) Technique
- Direct Sequence Spread-Spectrum (DSSS) Technique

Therefore, you are now skilled to complete the following tasks:

MQ6.1 Find the processing gain if the RF signal bandwidth of frequency-hopping spread-spectrum communication system is 129 MHz, and the information bit rate is 2800 bps.

MQ6.2 A speech signal is bandlimited to 3.3 kHz, and uses 128 quantization levels to convert it into digitized analog information data. This is required to be transmitted by a pseudorandom noise spread-spectrum communication system. Calculate the chip rate needed to obtain a processing gain of 21 dB.

MQ6.3 Create properties of a truly random binary maximum-length sequence.

MQ6.4 "Orthogonal codes play a major role in spread-spectrum techniques and permit a number of signals to be transmitted on the same nominal carrier frequency and occupy the RF bandwidths". Justify the statement.

LO 6.1

MQ6.5 Select the additional hardware needed with frequency-hopping spread-spectrum transmitter as compared to that of direct-sequence spread-spectrum transmitter.

MQ6.6 Outline the necessity of employing error-correction coding scheme in a frequency hopping spread-spectrum system.

MQ6.7 An FH-BPSK system using 32 orthogonal codes serves 40 users. Design the signal-to-interference ratio (SIR) per user if the system bandwidth is 20 MHz, and the system operates at the signaling rate of 32 kbps.

MQ6.8 Plan the clock rate of PN code-generator for a frequency-hopping spread-spectrum communication system utilizing fast-hop technique at the hop rate of 10 hops per information bit, having 512 number of different frequencies, information bit rate of 2800 bps, and final RF multiplication factor of 9.

MQ6.9 Paraphrase the significance of direct-sequence spread-spectrum (DSSS) modulation.

MQ6.10 How is direct-sequence spread-spectrum used in pass band transmission? Mention the nature of spread-spectrum signal at the DSSS receiver.

LO 6.2

Sure of what you have learnt so far? For answers, scan the QR code

OR

visit http://qrcode.flipick.com/index.php/42

Be ready for the next sections!

- Chapter-End Exercises

More than 300 carefully designed chapter-end exercises arranged as per levels of difficulty include Hands-on Projects, Objective-Type Questions, Short-Answer-Type Questions, Discussion Questions, Problems, and Critical Thinking Questions to enhance knowledge and test technical skills, with answers given through QR codes and URL.

Hands-on Projects

3.1 The IC MAX1472 is a crystal-referenced phase-locked loop (PLL) VHF/UHF transmitter designed to transmit ASK/OOK data in the 300 MHz to 450 MHz frequency range. It supports data rates up to 100 kbps, and adjustable output power to more than +10 dBm into a 50 Ω load. Design, fabricate and test a typical application using it.

3.2 Design and fabricate wireless transmitter and receiver operating at 433 MHz using ASK RF modules alongwith IC HT12E encoder and IC HT12D decoder.

3.3 Design a circuit to realize binary frequency shift-keying (BFSK) demodulator using PLL with standard IC 565. Apply its output signal to 3-stage *RC* low-pass filter and then use a comparator to obtain binary data output waveform.

3.4 Design a complete FSK communication system comprising of square-wave generator, FSK modulator and demodulator, comparator and display unit, using various linear ICs such as Timer IC555, PLL IC565, Comparator IC 741/351 and digital ICs as display drivers and LCD display.

3.5 Digital satellite broadcast reception set top box uses QPSK demodulation error correction IC CXD19611Q alongwith IC CXA3108Q channel selection and IC CXA3038N quadrature demodulation. Study its functional block schematic and design the complete circuit.

3.6 Baseband MSK is a robust means of transmitting data in wireless systems where the data rate is relatively low compared to the channel bandwidth. Design and implement MSK modulation and demodulation circuits using MX-COM's devices such as MX429 and MX469 as single chip solution. As an alternative method, you may realize MSK modulation circuit by directly injecting NRZ data into a frequency modulator with its modulation index set for 0.5.

Objective-Type Questions

For Interactive Quiz with answers, scan the QR code given here

OR

3.1 Identify the type of digital modulation technique that allows more bits per symbol and, therefore, greater speed in a given bandwidth than other digital modulation techniques.
(a) OQPSK
(b) π/4-QPSK
(c) 16-PSK
(d) QAM

3.2 Which parameter of an analog sine-wave carrier signal can be modulated by digital data information, leading to the most popular type of digital modulation technique?
...litude
...ency

Short-Answer-Type Questions

For answers, scan the QR code given here

4.1 Define entropy. State the condition for the entropy of the source to attain the maximum value.
4.2 Bring out clearly similarities and dissimilarities between the terms Information and Entropy.
4.3 How can the amount of information be measured which has been gained after the occurrence of the event? Under what conditions is ... zero?
4.4 Give an account of the various properties of informatio...
4.5 The entropy of a source is a function of the message p... that entropy is maximum when all the messages are equ...
4.6 Outline the upper and lower bounds on entropy.
4.7 Discriminate between conditional entropy and joint ent...
4.8 Determine the entropy of a binary memoryless source ... 1 occur with equal probability.
4.9 Why does downloading an audio or video stored file fro... take much longer time than it requires playing? Use Sh... to justify your answer.
4.10 What is the practical difficulty in achieving trans... theoretical capacity of the channel?
4.11 Specify source information rate for equiprobabl... equiprobable symbols, how is the source informatio... equiprobable symbols?
4.12 Why is white Gaussian noise considered the worst po... interference associated with signal transmission?
4.13 Why do we need channel coding? State Shannon's channel-coding theorem.
4.14 How does increase in data rate affect the quality of a received signal?
4.15 The Shannon–Hartley theorem shows that theoretically information rate and bandwidth are interchangeable. Give an example of the application of the Shannon–Hartley theorem.

Problems

For answer keys, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/161

6.1 Two *m*-sequences are generated in two different chain of shift registers (SRs), and then these two sequences are bit-by-bit XORed. Predict the number of shift registers required to obtain a processing gain of 45.15 dB.
6.2 If the PN generator is driven by a clock signal at the rate of 10 MHz, and the feedback register has 37 stages, Solve the total length of sequence in hours.
6.3 A pseudo-random (PN) sequence is generated using a feedback-shift register with four number of memory elements. What is the PN sequence length, N?
6.4 A spread-spectrum communication system has information bit duration of 4.095 ms, and PN chip duration of 1.0 μs. Design the value of shift-register length, $N = 2^m - 1$.
6.5 A frequency-hopping spread-spectrum communication system utilizes fast-hop technique at the hop rate of 16 hops per information bit, information bit rate of 2400 bps, frequency multiplication factor of 8. Calculate the minimum of different frequencies required to be generated by the frequency synthesizer, assuming this number to be a power of 2.

Discussion Questions

4.1 The digital pulse waveforms are propagated between source and destination throug... various levels of coding as required by different system applications. Elucidate th... functional block schematic of a typical digital communication system for reliabl... transmission of analog information. **[LO 4.1]**
4.2 Consider a binary source for which the symbol 0 occurs with probability p_0 and th... symbol 1 occurs with probability $p_1 = (1 - p_0)$. Show that the entropy is zero whe... symbol 0 occurs with either probability $p_0 = 0$, or $p_0 = 1$. Under what conditions th... entropy would be unity? **[LO 4.2]**
4.3 "The entropy of the extended discrete memoryless source is equal to *n* times *H*, th... entropy of the original discrete memoryless source". Justify this statement with the hel...

Critical Thinking Questions

6.1 Spreading of narrow bandwidth of baseband signal is accomplished through the use of a spreading code, called pseudo-noise (PN) code. State the two most important properties and three criteria which can be used to validate that a sequence of numbers is practically random. Draw a simple arrangement to generate PN sequences and explain the procedure for the same. **[LO 6.1]**
6.2 The most practical system applications employing direct sequence spread-spectrum (DSSS) techniques use digital carrier modulation formats such as BPSK and QPSK. Why is it necessary to employ coherent detection scheme at the DSSS receiver? Discuss various aspects of DSSS demodulator with coherent BPSK detection with reference to suitable functional block schematic diagram. **[LO 6.2]**
6.3 In the practical design of DS-CDMA digital cellular systems, there are three system parameters that affect the capacity as well as the bandwidth efficiency of the system. Recite them and write an expression for approximating the number of simultaneous users that can be supported in a practical multicell CDMA system, incorporating these three performance improving factors. **[LO 6.3]**

Roadmap of Various Target Courses

The book may be used as a textbook for a one-semester course on 'Digital Communications', 'Advanced Digital Communications', and 'Information Theory and Coding' at UG and PG levels of related engineering discipline. This book will also serve as a pre-requisite to study advanced courses such as 'Wireless and Mobile Communication', 'Digital Wireless Communications', 'Spread-Spectrum Communications' in subsequent semesters.

Organization of the Book

Chapter 0 in the book covers Electronic Communications System, Electronic Signal, etc., all of which are prerequisites to a course on Digital Communication and are taught to students in detail in previous semesters. A brief introduction has been retained within the book and a thorough review has been provided through a QR code.

Chapter 1 deals with analog information-encoding techniques (waveform coding) such as pulse code modulation (PCM), differential pulse code modulation (DPCM), adaptive PCM (ADPCM), delta modulation (DM), and adaptive delta modulation (ADM). In the beginning, the statements of Nyquist sampling theorems are given which is followed by discussions on practical aspects of sampling.

Chapter 2 explains various baseband signaling formats (line coding), the issue of intersymbol interference (ISI) and their possible solutions such as Nyquist pulse shaping and equalization filtering methods. It is followed by digital multiplexing techniques that are necessary for digital transmission of several baseband data over broadband digital channels. Finally, American and European digital signal hierarchy standards are covered with data-rate calculations.

Chapter 3 discusses various digital modulation techniques such as ASK, FSK, PSK, DBPSK, DEPSK, QPSK and their derivatives, QAM, MSK, and GMSK. Methods like carrier synchronization for coherent detection and performance analysis of digital modulation schemes are also covered here.

In **Chapter 4**, the fundamental concepts of information theory such as probabilistic behavior of information, entropy and its properties, characteristics of discrete memoryless channels, and mutual information are described. Shannon's channel-coding theorem and channel capacity are also discussed here.

The focus of **Chapter 5** is to implement source-coding and channel-coding techniques in digital communication so as to ensure efficient and reliable transfer of data. Major topics covered include basics of source encoding, various source-coding techniques such as Shannon–Fano, Huffman, and Lempel–Ziv codes. Error-control channel coding based on linear block codes such as Hamming, cyclic, BCH, Hadamard, LDPC, as well as convolution coding and Viterbi decoding algorithms are explained thoroughly. Finally, burst error-correction techniques including interleaving, RS codes, Turbo codes are also discussed.

Chapter 6 deals with various aspects of spread-spectrum communications. These include principles of spread-spectrum modulation, frequency-hopping and direct-sequence spread-spectrum (DSSS) techniques, application of DSSS in a multiuser CDMA based cellular system, and different multiple access techniques used in wireless communications.

With an objective to review the pre-requisites and provide additional topical coverage, **Appendix A** presents an overview of random signal theory, **Appendix B** gives the Fourier transform, **Appendix C** describes power spectra of discrete PAM signals, and **Appendix D** explains Wiener optimum filters for waveform estimation. **Appendices E** and **F** give important mathematical formulae, and abbreviations and acronyms respectively.

Get More on the Web

There are a number of supplementary resources which are available on the book's website *http://highered.mheducation.com/sites/933921952x*.

For Instructors

- Solutions Manual, PowerPoint Lecture Slides

For Students

- Additional Reading Material

Acknowledgements

Writing this book in tune with mandatory requirements of Outcome-Based Education and Learning-Objective-oriented approach involved extensive research and efforts. I am grateful to all those who directly or indirectly provided me guidance and support. At the outset, I would like to express my gratitude for the encouragement and inspiration received from Dr Ashok Chitkara, Chancellor, Chitkara University; Dr Madhu Chitkara, Vice Chancellor, Chitkara University; Dr Archana Mantri, Pro-Vice Chancellor, Chitkara University, Punjab, and my colleagues of Chitkara University.

I would like to thank the editorial team at McGraw Hill Education (India) for bringing out this book in its present form.

The dream of my beloved parents, who wished me to be a mentor for aspiring young engineering students, is fulfilled through the publication of this book. Their blessings are similar to that bestowed by the Almighty. I remain indebted to my wife, Mrs Pinki, and my son, Er Pankaj, for their continuous support.

Special thanks to the reviewers mentioned here for taking out time and providing encouraging comments and valuable suggestions regarding improvement of the manuscript.

Faculty Reviewers

Amber Khan	*Jamia Millia Islamia, New Delhi*
Bhasker Gupta	*Jaypee University of Information Technology, Waknaghat, Himachal Pradesh*
Anupama Marwaha	*Sant Longowal Institute of Engineering and Technology, Sangrur, Punjab*
Arvind Kumar	*National Institute of Technology (NIT), Kurukshetra, Haryana*
Shelej Khera	*BRCM College of Engineering and Technology, Bahal, Haryana*
Amit Sehgal	*G L Bajaj Institute of Technology and Management, Greater Noida, Uttar Pradesh*
Mohit Kumar Srivastava	*Maharana Pratap Engineering College (MPEC), Kanpur, Uttar Pradesh*
Puneet Srivastava	*Raj Kumar Goel Institute of Technology, Ghaziabad, Uttar Pradesh*
Lokesh Bansal	*ITS Engineering College, Greater Noida, Uttar Pradesh*
Pankaj Agarwal	*JP Institute of Engineering and Technology, Meerut, Uttar Pradesh*
Atul Bansal, Abhay Chaturvedi	*GLA University, Mathura, Uttar Pradesh*
Manisha Sharma	*Bhilai Institute of Technology, Durg, Chattisgarh*
Aparna Kushwah	*University Institute of Technology-RGPV, Bhopal, Madhya Pradesh*

Vishnu Priye	*Indian School of Mines (ISM), Dhanbad, Jharkhand*
Ajit Panda	*National Institute of Science and Technology (NIST), Brahmapur, Odisha*
Partha Sarathi Khuntia	*Konark Institute of Science and Technology (KIST), Jatni, Odisha*
Barnali Dey	*Sikkim Manipal Institute of Technology, Gangtok, Sikkim*
Makrand Kulkarni	*K J Somaiya College of Engineering, Mumbai, Maharashtra*
Kalawati Patil	*Thakur College of Engineering and Technology, Mumbai, Maharashtra*
G R Patil	*Army Institute of Technology, Pune, Maharashtra*
Bharat Chaudhari	*International Institute of Information Technology, Pune, Maharashtra*
Shripad P Mohani, R A Patil	*College of Engineering (COEP), Pune, Maharashtra*
Rajeshree Raut	*Shri Ramdeobaba College of Engineering and Management, Nagpur, Maharashtra*
Vijay Ghanwat	*Shri Guru Gobind Singhji Institute of Engineering and Technology, Nanded, Maharashtra*
Mehul G Thakkar, Mikita Gandhi	*A D Patel Institute of Technology, Vallabh Vidyanagar, Gujarat*
Namrata V Langhnoja	*Government Engineering College, Rajkot, Gujarat*
Smith S Thavalapill	*Dharmsinh Desai University, Nadiad, Gujarat*
R. Vinod Kumar	*Sona College of Technology, Salem, Tamil Nadu*
T Lakshmi Kandhan	*Anna University, Chennai, Tamil Nadu*
Ravi Shankar	*Sapthagiri College of Engineering, Bengaluru, Karnataka*
Shashi Kumar	*Acharya Institute of Technology, Bengaluru, Karnataka*
K L Sudha	*Dayananda Sagar College of Engineering, Bengaluru, Karnataka*
Sridhar Iyer	*Christ University, Bengaluru, Karnataka*

Faculty Reviewers' Comments

Some of the comments offered by the peer reviewers on the typescript (which provided immense motivation and encouragement) are reproduced below:

"The additional features used in-between are excellent and are very helpful for the students to understand the topics."

Arvind Kumar
National Institute of Technology (NIT), Kurukshetra, Haryana

"The Recall, Important, and Attention features are given nicely and will help students in better understanding and retention of the concepts."

R A Patil
College of Engineering, Pune, Maharashtra

"I found that the clarity is good. The flow is nicely maintained. The key terms are presented at right places. The content is a good mix of principle and its use or application."

Vijay Ghanwat
Shri Guru Gobind Singhji Institute of Engineering and Technology, Nanded, Maharashtra

"Good collection of questions at the end of the chapter. Sufficient content has been included to map various LOs. The collection of topics in this chapter may be of more interest for the readers of that level."

Amit Sehgal
G L Bajaj Institute of Technology and Management, Greater Noida, Uttar Pradesh

"The writing has a beautiful flow as if the book is actually speaking to the reader."

Barnali Dey
Sikkim Manipal Institute of Technology, Gangtok, Sikkim

"Discussion on baseband transmission and multiplexing is presented with sufficient details. It will be useful for students who will be taking a course on data communication and computer networking."

G R Patil
Army Institute of Technology, Pune, Maharashtra

"For the first-time reader, explanation for all questions that may arise while reading is answered."

K L Sudha
Dayananda Sagar College of Engineering, Bengaluru, Karnataka

Students Who Were Helpful

Some of the comments offered by the student reviewers on the typescript are reproduced below. For more comments, visit *https://www.facebook.com/pages/Digital-Communication/655459447899660*

Soumyadeep Das
Jalpaiguri Government Engineering College, Jalpaiguri, West Bengal

- The first intriguing thing in this book is that every chapter has LEARNING OBJECTIVES at the beginning. The LO column mainly states the concepts discussed in the chapter, which comes in very handy for first-time readers, especially when you don't know what to assimilate while going through the chapter.
- I personally prefer books with an extensive set of problems because trying to solve them really helps you implement the concepts you have just learnt.
- Some unique features like ATTENTION, RECALL, etc., unlike other books is very helpful in retaining the important concepts of the book.

Subhechcha Roy Choudhury
Jalpaiguri Government Engineering College,
Jalpaiguri, West Bengal

Before providing any statement regarding the review, let us recall what we exactly look for in a technical book in the engineering domain. The simple rule is the following: The ease with which the knowledge and the information is passed on. That way, even critical things get into the head easily. McGraw Hill well accomplishes the requirement by following the golden step in the book.

The LOs are stated before the chapter commences to allow students get the notion about the motion of the chapter. Following this brief, the concepts provided, rightly justify the objectives mentioned at the beginning.
The link between the topics are well connected by the RECALL tab where one gets the feed required from the earlier chapter.

The writing style, as I mentioned, demand appreciation. Usage of block diagrams, tables, subheadings, key points, borders, technical yet simple statements make it easier for us to grab even the difficult aspects conveniently.

Prashant Rana
Dharmsinh Desai University, Nadiad, Gujarat

Recapitulation of the previous chapter is very good for quick revision. Even if one hasn't studied the previous chapter then too one can make out the important things of the previous chapter.

Pooja Patil
Dharmsinh Desai University, Nadiad, Gujarat

The writing style of the author is quite impressive. All the key terms and concepts are explained in an easy, understandable manner. And it follows a nice sequential flow.

The classification of the questions on the basis of difficulty levels is an important feature; it helps the reader for self-assessment. And solving these questions also comes in handy for improving skills.

Nikhil Ramtri

Dharmsinh Desai University, Nadiad, Gujarat

- Features like RECALL, etc., serve an important role in the book. These topics force oneself to ask questions and to pay more attention to the important concepts present in the book.
- The *Questions* feature will be a great help as it starts from simple questions and draws the attention of each and every student. It then takes you to the harder parts, forcing you to read the book—a good activity!

Phalguni Shukla

GLA University, Mathura, Uttar Pradesh

- All the topics are well covered with a good link between them, which maintains a flow for easy understanding of the concepts. Also, I liked the *Self-Assessment Exercises* after the end of each LO. This would clear the basics and firm our foundation.
- Example is the thing that a reader ***needs*** to understand a concept very clearly, and the text is well covered with examples and flow diagrams, so there is no ambiguity in readers' minds.
- All types of questions are covered, from easy to difficult, which offers variety to the reader.

Other students who also helped us with their feedback:

Nidhi Agrawal	*GLA University, Mathura, Uttar Pradesh*
Vaibhav Bohra, Aastha Srivastava, Abhimit Ghosh, Rupal Jain	*Bhilai Institute of Technology, Durg, Chattisgarh*
Rushin Patel, Niket Thakkar	*Dharmsinh Desai University, Nadiad, Gujarat*
Saikat Das, Sanket Mishra	*Jalpaiguri Government Engineering College, Jalpaiguri, West Bengal*

Feedback Request from Author

I am sure every student will find this book rich in content along with unique pedagogical features which fully justifies the learning-objective-oriented text. This will certainly give a clear-cut advantage over other books on a similar course. The academic community as a whole will enjoy simplified yet an extensive and elaborate approach to every topic covered in the book. All efforts have been made to make this book error-free, but I believe there is always ample scope of improvement in all our efforts. Your valuable suggestions/feedback are most welcome at *tarsemsingal@gmail.com*.

T L Singal

Publisher's Note

All our books are developed with a lot of love and sincere hard work. If you would like to make a suggestion or share your views, write in to us at *info.india@mheducation.com*, mentioning the title and author's name in the subject line.

0 Preliminaries

INTRODUCTION

Electronic communications serve the purpose of transfer of user information from one device to another via some form of communications channel. A *digital signal* is one in which the amplitude of the signal maintains a constant level for some period of time and then changes instantaneously to another constant level. *Baseband digital transmission* means sending a digital signal over a channel without changing the form of a digital signal. *Broadband digital transmission* means changing the digital signal to an analog signal using digital modulation technique. *Modulation* is the process of facilitating the translation of low-frequency baseband signal to high-frequency transmission signal. In fact, at least one attribute (amplitude, frequency, or phase) of a high-frequency analog signal, called the *carrier* (because it carries the information), is in proportion to the instantaneous value of the baseband (also called modulating) signal at the transmitter end. *Digital modulation* requires less power to transmit, makes better use of the available bandwidth, performs better even in the presence of other interfering signals, and can offer compatible error-correcting techniques with other digital systems. *Multiplexing* allows more than one signal to be transmitted concurrently over a single medium, e.g., Frequency Division Multiplexing (FDM).

Dear student, for detailed pre-requisite for the course on Digital Communication. Please scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/30

1 PCM, and Delta Modulation and Demodulation

Learning Objectives

To master PCM, and Delta Modulation and Demodulation, you must cover these milestones:

LO 1.1 Understand various practical aspects of sampling such as methods of sampling, Nyquist sampling theorems, aliasing, PAM/TDM application

LO 1.2 Analyze the functional processes (sampling, quantization, and binary encoding) of pulse-code modulation (PCM) technique

LO 1.3 Generate variants of PCM such as differential PCM (DPCM) and adaptive differential PCM (ADPCM)

LO 1.4 Implement delta modulation and adaptive delta modulation techniques

Essence of Pulse Code Modulation

Recently, we have seen exponential growth in digital communication technologies ranging from digital cellular mobile phones and high-definition TV broadcasting systems to wireless Internet communication networks. In modern electronic communication systems, most information (also known as baseband signals)—whether it is analog in nature such as voice, music, and video, or, in digital form such as binary data from a computer—is transmitted using digital transmission techniques. The conversion from an analog signal to digital data representation involves three sequential steps—sampling, quantization, and encoding. In the first step, an analog signal is sampled at discrete times and then subsequently transmitted by means of analog pulse modulation, known as **Pulse Amplitude Modulation (PAM)**. For simultaneous transmission of many baseband signals over a common channel, PAM signals are digitally multiplexed using time-division multiplexing techniques. In the second step, an analog signal is not only sampled at discrete times but the samples are also quantized to discrete levels, the process known as quantization. The quantized digitized signal is an approximation of the original analog signal, known as **quantized PAM**. Finally, each quantized sample is encoded into a sequence of binary pulses, known as **Pulse Code Modulation (PCM)**. In this way, we achieve digital representation of an analog signal.

PCM is a kind of source coding, i.e., the conversion from analog signal to digital signal. After converted to digital signal, it is easy for us to process the signal such as encoding, filtering the unwanted signal and so on. Besides, the quality of digital signal is better than analog signal. This is because the digital signal can be easily recovered by using comparator. PCM modulation is commonly used in audio and telephone transmission. The main advantage is that PCM modulation only needs 8 kHz sampling frequency to maintain the original quality of audio.

INTRODUCTION

Digital Representation of Signals

We know that analog signals are characterized by data whose values vary over a continuous range of amplitude levels and are defined for a continuous range of time. For example, a particular speech waveform has amplitudes which may vary over a continuous range, a part of recorded music, or the temperature of a certain location over time which can assume an infinite number of possible values.

These types of analog signals are converted to digital data having only a finite number of possible values using a device known as *codec* (coder-decoder). The digital signal thus produced is known as *digitized analog data*, and the process is also known as *digitization*. On the receiver side, a similar codec device converts the received bit stream to the original analog information data. The encoding of analog information such as voice, text, image, video, or physical sensor output to produce digital data and then converting it to suitable signalling formats makes digital transmission possible.

Most electronic communication systems, wireline or wireless, are going “digital”. The principle feature of a digital communication system is that during a finite interval of time, it sends a waveform from a finite set of possible waveforms at the transmitter. At the receiver, the objective is just to determine from a noise-affected signal which waveform from the finite set of waveforms was sent by the transmitter. So it is quite easy to regenerate digital signals.

Digital transmission refers to transmission of digital signals between two or more points in an electronic communications system. If the original information is in analog form then it needs to be converted to digital pulses prior to transmission and converted back to analog signals in the receiver. The conversion of analog signal to digitized pulses is known as *waveform coding*. The digitized signals may be in the form of binary or any other form of discrete-level digital pulses. Figure 1A depicts a simple model of digital transmission of analog data by using waveform coding.

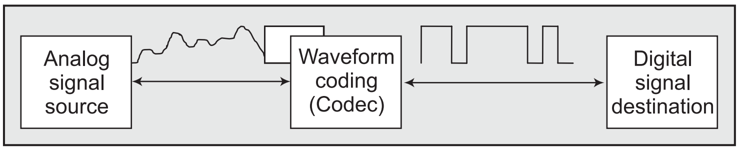


Figure 1A Analog-to-Digital Encoding

An analog information signal is converted to digital data (sequence of binary symbols) by waveform coding techniques such as pulse code modulation, differential pulse code modulation, or delta modulation. The occurrence of binary digits, 1s and 0s, is not uniform in any digital data sequence. These binary digits are converted into electrical pulses or waveforms.

Digital Transmission — Advantages

Digital transmission is the most preferred method for transmission of either analog signal (after digitization) or digital data. Why is that so? Let us discuss some of its inherent advantages.

- ***Noise immunity.*** Digital signals are inherently less susceptible to interference and external noise since the relative levels of received pulses are evaluated during a precise time interval at the receiver. Digital communication systems are more resistant to additive noise because they use signal regeneration rather than signal amplification as in analog transmission systems.
- ***Error detection and correction.*** Digital signals are simpler to measure and compare the error performance of different digital communication systems. Digital circuits are less subject to distortion and interference mainly because binary digital signals are two-state (on or off) signals. Transmission errors can be easily detected and corrected with reasonable accuracy. An important measure of system performance in a digital communication system is the probability of error, that is, the probability of incorrectly detecting a digit. Using digital signal processing techniques, extremely low error rates with capability of producing high signal fidelity are possible by employing appropriate error detection and correction schemes.
- ***Ease of multiplexing.*** Digital signals are better suited for processing and combining many baseband signals using time-division multiplexing techniques. Even digitized analog signals are processed using digital signal-processing methods, which includes bandlimiting the signals with filters and amplitude equalization.
- ***Integration of analog and digital information data.*** Both analog and digital information data can be processed digitally using latest digital hardware and software technologies. Thus, digital transmission meets the user requirement of integrating voice, digital data, and video using the same electronic communication systems.
- ***Use of signal regenerators.*** Digital transmission systems use signal regeneration rather than signal amplification as in analog transmission systems. Digital regenerators sample the received noisy signals, then reproduce an entirely new digital signal with same signal-to-noise ratio (S/N) as that of original transmitted signal. So digital signals can be transmitted over much longer distances without signal deterioration.

- ***Data integrity.*** Due to usage of digital repeaters or signal regenerators for longer distance transmission of digital data, the effects of noise and other transmission impairments can be minimized to a large extent. It is relatively simpler to store digital signals. Moreover, the transmission data rate can be made adaptable to interface with different type of equipments.
- ***Data security.*** Various encryption techniques can be easily applied to digitized analog data and digital data in digital transmission.
- ***Ease of evaluation and measurement.*** The received digital pulses are evaluated during a pre-defined precise sample interval, and a decision is made whether the received digital pulse is above or below a specified threshold level.
- ***More suitable for processing.*** Digital transmission is more convenient for processing, regeneration, and amplification of digital signals. The digital signal processing includes filtering, equalizing, phase-shifting, and storing of digital data. There are techniques (source coding) for removing redundancy from a digital transmission, so as to minimize the amount of information that has to be transmitted. Also, there are techniques (channel coding) for adding controlled redundancy to a digital transmission, such that errors that occur during transmission may be corrected at the receiver.
- ***Improved performance.*** Digital signals can be easily measured and evaluated. It enables to compare the error performance in terms of bit-error rate (BER) of one digital system to another. Moreover, transmission errors can be accurately detected and corrected.
- ***Advancement of digital technology.*** The advent of VLSI, microcontroller, digital signal processing, and embedded systems technology has paved way for miniaturization of electronic devices, besides considerable reduction in power consumption, weight, and cost.
- ***Utilization of available capacity.*** Using optical fibers and satellite transmissions, very high bandwidth is available which can be optimally utilized with efficient digital multiplexing techniques.

Digital Transmission — Disadvantages

- Transmission of digitally encoded analog signals requires significantly more bandwidth than that required for transmission of the original analog signal.
- Digital transmission requires precise time synchronization between the clocks used in transmitters and receivers, thereby a significant part of their resources are required to be allocated.
- There is need of additional hardware for encoding for converting analog signals to digital pulses prior to digital transmission, as well as additional decoding circuitry at the receiver to convert back to the original analog signals.
- There is another aspect—the quality of service can degrade all of a sudden from very good to very poor when the signal-to-noise ratio drops below a specified threshold level in digital transmission.
- Digital transmission systems are incompatible with existing analog transmission facilities.

A PRIMER

A Typical Digital Communication Link

It is required to use complicated signal-processing techniques to implement various functional needs of digital communication link efficiently. Some of these functions include source-coding and channel-coding techniques, sophisticated timing and fast acquisition operations for synchronizing, and for the efficient and reliable transmission of data over the communication channel. Passband modulation techniques are required to be used for the transmission of digitized speech and data over a wireless channel. This necessitates the use of synchronization for the locally generated carrier frequency, carrier phase, and symbol timing at the receiver. Figure 1B shows the block diagram of a typical digital communication link.

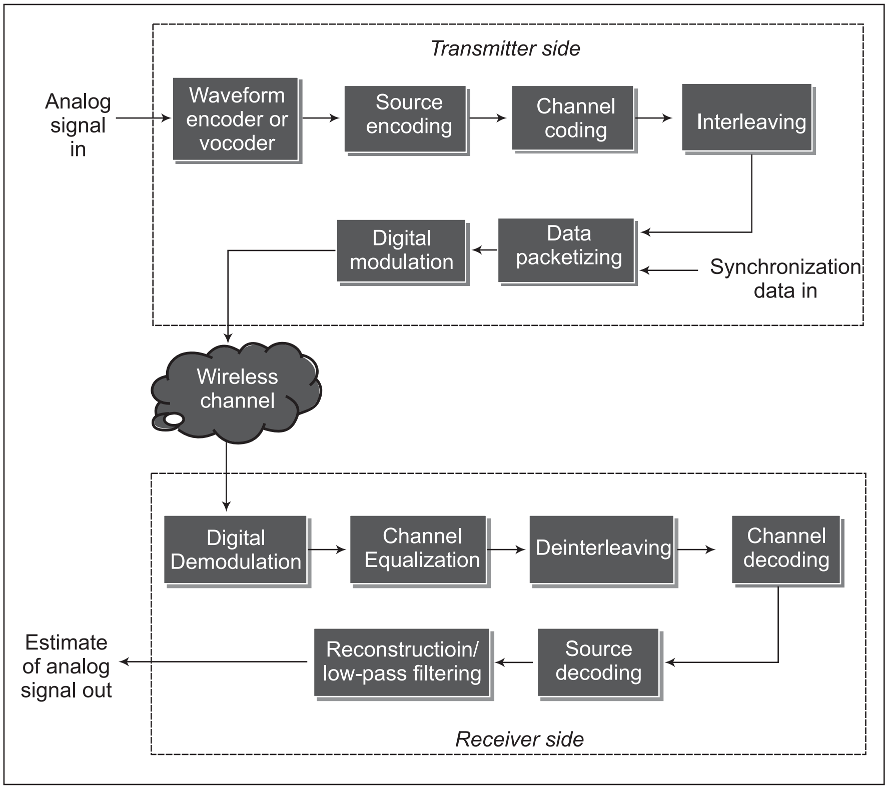


Figure 1B Block Diagram of a Typical Digital Communication Link

Steps of Signal Processing

- The analog signal input is first sampled and quantized to convert an analog signal into equivalent digitized analog signal.
- In order to remove redundant information, the digitized analog signal is source encoded without compromising the ability of the receiver to provide a high-quality reproduction of the original signal.
- The channel encoder introduces controlled redundancy bits into the analog-encoded signal to provide protection against channel noise.
- A wireless channel produces errors in the form of data bursts, mainly due to deep signal fades.
- To mitigate this particular channel impairment, an interleaver is used for the purpose of pseudo-randomizing the order of the binary symbols in the channel-encoded signal in a deterministic manner.
- The function of a packetizer is to convert the encoded and interleaved sequence of digitized analog data into successive packets. Each packet occupies a significant part of a basic data

frame. Each frame also includes synchronization bits in order to synchronize the timing operations in the receiver with the corresponding ones in the transmitter. Knowing the estimate of the channel impulse response, channel equalization at the receiving end of the digital communication link is made possible.

- The packetized speech data is then modulated onto a sinusoidal carrier for transmission over the wireless channel.
- The receiver side consists of a cascade of several functional blocks in order to reverse the corresponding operations performed by the transmitter and the wireless channel.
- The digital demodulator converts the modulated received RF signal into its baseband form without any loss of information.
- The baseband processor operates on the resulting complex baseband signal to estimate the unknown channel impulse response, and channel equalization.
- The resulting output is then deinterleaved, channel decoded, source decoded, and low-pass filtered for final delivery of an estimate of the original analog signal to the receiver output.

IMPORTANT! With digital transmission systems, a wireline medium such as a pair of wires, coaxial cable, or an optical fiber cable is required since the pulses can propagate down the physical medium. Digital pulses cannot be propagated through a wireless medium such as free space or the earth's atmosphere.

In this chapter...

LO 1.1 >
- We begin with a description of the sampling process based on the *sampling theorem* which determines that the sampling rate must be large enough to allow the analog signal to be reconstructed from the samples with adequate accuracy.

LO 1.2 >
- Then, we discuss the process of obtaining digital representation of analog signal through the process of sampling, quantization, and binary encoding, leading to *Pulse Code Modulation (PCM)*.

LO 1.3 >
- The discussion is carried forward to evolve variants of PCM such as *Differential PCM (DPCM) and Adaptive DPCM (ADPCM)*.

LO 1.4 >
- For efficient digital pulse modulation techniques, we employ *Delta Modulation (DM)*, and *Adaptive Delta Modulation (ADM)*.

LO 1.1

1.1 PRACTICAL ASPECTS OF SAMPLING

Why Sampling? As stated earlier, the first step in the evolution from analog to digital transmission is the conversion of analog information to digital representation. In order to achieve this, an analog signal is sampled at discrete times. This means that an analog signal is converted into a corresponding sequence of pulses that are usually spaced uniformly in time.

IMPORTANT! The process of sampling an analog signal is the fundamental to digital signal processing and digital communication. Clearly, the essence of the sampling process lies in the fact that we choose the sampling rate properly, so that the sequence of samples uniquely defines the original analog signal. Sampling is usually described in the time domain.

1.1.1 Sampling Theorem

The process of sampling is essentially based on the sampling theorem which determines that the sampling rate must be large enough to allow the analog signal to be reconstructed from the samples with adequate accuracy.[1]

Sampling Theorem for Baseband Signal

The *sampling theorem* for a *baseband signal* (strictly bandlimited analog signal of finite energy) may be stated in two equivalent parts:

- A baseband signal having no frequency components higher than f_m Hz may be completely recovered from a knowledge of its samples taken at the rate of at least $2 f_m$ samples per second, that is, the sampling frequency $f_s \geq 2 f_m$.
- A baseband signal having no frequency components higher than f_m Hz is completely described by its sample values taken at uniform intervals less than or equal to $1/(2f_m)$ seconds apart, that is, the sampling interval $T_s \leq 1/(2f_m)$ seconds.

Sampling Theorem for Bandpass Signal

The *sampling theorem* for *bandpass analog signal* may be stated as follows:

- An analog bandpass signal with highest frequency f_h and specified bandwidth $B = f_h - f_l$ where f_l is the lowest frequency, can be recovered from its samples through bandpass filtering by sampling it at $f_s \geq 2(f_h - f_l)$, assuming f_h to be an integral multiple of f_s.

Important Definitions

For Baseband Signal

- The minimum sampling rate $f_s = 2 f_m$ samples per second is called the ***Nyquist rate***.
- The maximum sampling interval $T_s = 1/(2f_m)$ seconds is called the *sampling time*, or ***Nyquist interval***.

For Bandpass Signal

- The minimum sampling rate of $(2B)$ samples per second, for an analog signal bandwidth of B Hz, is called the ***Nyquist rate***.
- The reciprocal of Nyquist rate, $1/(2B)$, is called the ***Nyquist interval***, that is, $T_s = 1/(2B)$.

Dear student... *For proof of the Sampling Theorem, visit student's resources available at http://www.mhhe.com/singal/dc1*

Significance of Nyquist Sampling Theorem

The Nyquist sampling theorem is also called **uniform sampling theorem** because the samples are taken at uniform intervals. Note that the sampling theorem requires the following:

- The sampling rate be fast enough so that at least two samples are taken during the time period corresponding to the highest frequency spectral component present in the analog signal.
- Then these samples uniquely determine the analog signal, and the analog signal may be reconstructed from these samples with no distortion.

[1] The *Sampling Theorem* is the fundamental principle of digital communications. It provides the basis for transmitting analog information signal by use of digital transmission techniques. The analog information signal such as speech or video signals is sampled with a pre-determined train of narrow rectangular pulses in such a way so as to closely approximate the instantaneous sampling process. In the sampling process, a continuous-time varying analog signal is converted into a discrete-time signal by measuring the signal amplitude level at periodic instants of time.

IMPORTANT! The Nyquist sampling theorem establishes the minimum sampling rate (f_s) that must be used to transmit analog signal in a given digital communication system. As per Nyquist criterion, the minimum sampling rate must be equal to twice the highest frequency present in the input analog signal in order to ensure reliable reconstruction of the information signal at the receiver.

Thus, Nyquist criterion is just a theoretically sufficient condition which may allow an analog signal to be reconstructed completely from a set of uniformly spaced discrete-time samples.

ATTENTION There is an interesting result which follows from the Sampling Theorem. It establishes a basic relationship in digital communications, which states that theoretically a maximum of 2B independent pieces of information (samples) per second can be transmitted, error-free, over a noiseless channel of B Hz bandwidth (specified earlier as Shannon capacity for noise-free channel).

What next Let us now apply the Sampling Theorem to analog signals represented by mathematical expressions. This is illustrated with the help of the following examples.

SOLVED EXAMPLE 1.1.1 Nyquist Rate

Using the Nyquist Sampling theorem for a baseband signal, determine the Nyquist rate for an analog signal represented by $s(t) = 10 \sin [2\pi (4 \times 10^3)t]$.

Solution As per the Nyquist sampling theorem for a baseband signal, the Nyquist rate is given as $f_s = 2f_m$ samples per second.
For the given analog signal, $s(t) = 10 \sin [2\pi (4 \times 10^3)t]$; we deduce that $f_m = 4$ kHz.
Hence, Nyquist rate $f_s = 2 \times 4$ kHz = **8 kHz** **Ans.**

SOLVED EXAMPLE 1.1.2 Nyquist Rate and Nyquist Interval

Using the Nyquist sampling theorem for a baseband signal, determine allowable sampling rate and Nyquist interval for an analog signal represented by $s(t) = 5 \cos (100\, \pi t) + 10 \cos (2000\pi t)$.

Solution The given analog signal $s(t) = 5 \cos (100\, \pi t) + 10 \cos (2000\pi t)$ is a composite signal which contains two frequency components that can be deduced as below:

$$2\pi f_1 t = 100\, \pi t \quad \Rightarrow \quad f_1 = 50 \text{ Hz}$$

$$2\pi f_2 t = 2000\, \pi t \quad \Rightarrow \quad f_2 = 1000 \text{ Hz}$$

Thus, highest frequency component present in the signal, $f_m = 1000$ Hz
Allowable sampling rate is the minimum Nyquist rate.
As per the Nyquist sampling theorem for a baseband signal, the minimum Nyquist rate is given as $f_s = 2f_m$ samples per second.

Therefore, $f_s = 2 \times 1000 \text{ Hz} = 2000 \text{ Hz, or } 2 \text{ kHz}$ **Ans.**

We know that the Nyquist interval T_s is related with the Nyquist rate as $T_s = 1/f_s$.
Hence, Nyquist interval $T_s = 1/2000 = 0.5$ ms **Ans.**

SOLVED EXAMPLE 1.1.3 Recommended Sampling Frequency

Consider a composite analog signal represented by

$$s(t) = 3 \cos (50\, \pi t) + 10 \sin (300\, \pi t) - \cos (100\, \pi t)$$

Determine

(a) the highest frequency component present in the given analog signal

(b) Nyquist rate
(c) recommended sampling frequency.

Solution The given analog signal $s(t) = 3\cos(50\,\pi t) + 10\sin(300\,\pi t) - \cos(100\,\pi t)$ contains three frequency components as calculated below:

$$2\pi f_1 t = 50\,\pi t \quad \Rightarrow \quad f_1 = 25 \text{ Hz}$$
$$2\pi f_2 t = 300\,\pi t \quad \Rightarrow \quad f_2 = 150 \text{ Hz}$$
$$2\pi f_3 t = 100\,\pi t \quad \Rightarrow \quad f_3 = 50 \text{ Hz}$$

(a) It is seen that the highest frequency component present in the given analog signal, $f_m = 150$ Hz **Ans.**
(b) Nyquist rate, $f_s = 2f_m = 2 \times 150 \text{ Hz} = 300$ Hz **Ans.**
(c) Recommended sampling frequency, $f_s \geq 2f_m$; $\Rightarrow$ $f_s \geq 300$ Hz **Ans.**

SOLVED EXAMPLE 1.1.4 **Nyquist Rate**

Determine the Nyquist rate and Nyquist interval for the analog signal represented by $s(t) = \dfrac{1}{2\pi}\cos(4000\pi t)\cos(1000\pi t)$.

Solution Using the trigonometric identity $2\cos A\cos B = \cos(A+B) + \cos(A-B)$ for the given analog signal $s(t) = \dfrac{1}{2\pi}\cos(4000\pi t)\cos(1000\pi t)$, we get

$$s(t) = \frac{1}{4\pi}\left[\cos(4000\pi t + 1000\pi t) + \cos(4000\pi t - 1000\pi t)\right]$$

$$\Rightarrow \quad s(t) = \frac{1}{4\pi}\left[\cos(5000\pi t) + \cos(3000\pi t)\right]$$

It contains two frequency components which can be calculated as

$$2\pi f_1 t = 5000\,\pi t; \quad \Rightarrow \quad f_1 = 2500 \text{ Hz}$$
$$2\pi f_2 t = 3000\,\pi t; \quad \Rightarrow \quad f_2 = 1500 \text{ Hz}$$

Therefore, the highest frequency component present in the signal, $f_m = 2500$ Hz

Nyquist rate, $f_s = 2f_m = 2 \times 2500 \text{ Hz} = 5000$ Hz **Ans.**

Nyquist interval, $T_s = \dfrac{1}{f_s}\text{s} = \dfrac{1}{5000 \text{ Hz}} = 0.2$ ms **Ans.**

SOLVED EXAMPLE 1.1.5 **Sampling for Baseband and Bandpass Signals**

Consider an analog composite signal described by the mathematical expression as $s(t) = 5\cos(6000\,\pi t) + 5\cos(8000\,\pi t) + 10\cos(10000\,\pi t)$. Find minimum value of the sampling rate considering the sampling theorem for (a) low-pass signals; and (b) band-pass signals.

Solution

$$s(t) = 5\cos(6000\,\pi t) + 5\cos(8000\,\pi t) + 10\cos(10000\,\pi t) \qquad \text{(Given)}$$

or, $$s(t) = 5\cos(2\pi \times 3000\,t) + 5\cos(2\pi \times 4000\,\pi t) + 10\cos(2\pi \times 5000\,\pi t)$$

Thus, the input analog signal has three frequency components,

$$f_{m1} = 3000 \text{ Hz}, f_{m2} = 4000 \text{ Hz, and } f_{m3} = 5000 \text{ Hz}$$

(a) The minimum value of the sampling frequency considering the *sampling theorem for low-pass signals* is twice the maximum frequency component present in the input signal, that is,

$$f_s = 2\,f_{m3} = 2 \times 5000 \text{ Hz} = 10000 \text{ Hz} \qquad \textbf{Ans.}$$

(b) The minimum value of the sampling frequency considering the *sampling theorem for band-pass signals* is twice the difference between maximum and minimum frequency components present in the input signal, that is,

$$f_s = 2\,(f_{m3} - f_{m1}) = 2 \times (5000 - 3000) \text{ Hz} = 4000 \text{ Hz} \qquad \textbf{Ans.}$$

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 1.1.6 Calculate the maximum value of the Nyquist interval required to sample an analog signal, $s(t) = \cos(500\,\pi t) + 0.5 \sin(1000\,\pi t)$.

Ex 1.1.7 The analog signal $s(t) = 10 \cos(20\,\pi t) \cos(200\,\pi t)$ is sampled at the rate of 250 samples per seconds. Determine the following:
(a) The maximum frequency component present in the analog signal.
(b) Nyquist rate.

ATTENTION For an analog signal $s(t)$ whose highest frequency spectral component is f_h, the sampling frequency f_s must be no less than $f_s = 2\,f_h$ only if the lowest frequency spectral component of $s(t)$ is $f_l = 0$. In the more general case, where $f_l \neq 0$, it may be that the sampling frequency need be no longer than $f_s = 2\,(f_h - f_l)$.

1.1.2 Methods of Sampling

Recall

As discussed in the preceding section, sampling is required to faithfully represent an analog signal by its discrete samples in the time domain. For practical needs, these samples are of finite width in the form of pulses.

Three Methods

There are three distinct methods of sampling:

- ***Ideal Sampling***—An *impulse* at each instant of sampling. It is also known as impulse sampling.
- ***Natural Sampling***—A pulse of short width with *varying* amplitude at each instant of sampling.
- ***Flat-top Sampling***—A pulse of short width with *fixed* amplitude at each instant of sampling.

Ideal Sampling

In *ideal sampling*, an arbitrary analog signal is sampled by a train of impulses at uniform intervals, T_s. An impulse (having virtually no pulse width) is generated at each instant of sampling. Figure 1.1.1 shows the waveform for ideal sampling.

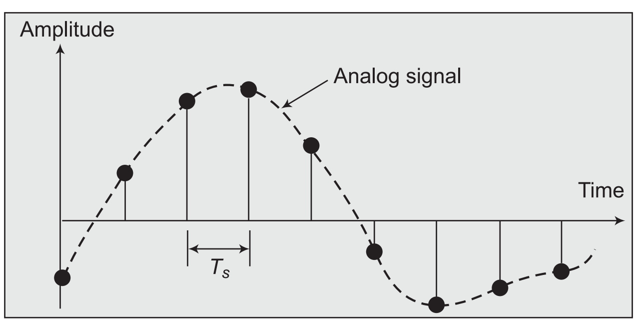


Figure 1.1.1 *Ideal Sampling*

Natural Sampling

Natural sampling refers to sampled signals when tops of the sampled pulses retain their natural shape during the sample interval. In natural sampling, an arbitrary analog signal is sampled by a train of pulses having finite short pulse width occurring at uniform intervals. The amplitude of each rectangular pulse follows the value of the analog information signal for the duration of the pulse. Figure 1.1.2 shows the waveform for natural sampling.

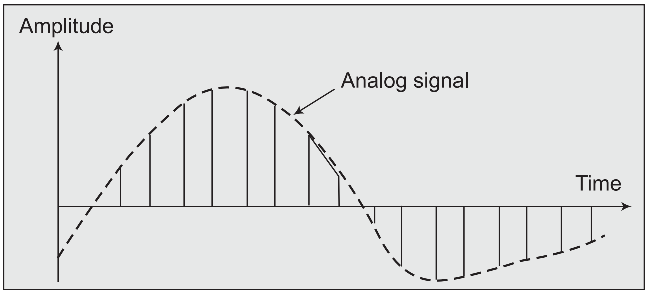


Figure 1.1.2 *Natural Sampling*

The original analog signal can be recovered at the receiver by passing the sampled signal through an ideal low-pass filter without any distortion provided its bandwidth satisfies the condition $f_m < B < (f_s - f_m)$.

It is quite evident that there are certain *disadvantages* of natural sampling. It is difficult for an analog-to-digital converter to convert the natural sample to a digital code. In fact, the output of analog-to-digital converter would continuously try to follow the changes in amplitude levels and may never stabilize on any code.

Flat-top Sampling

Flat-top sampling refers to sampled signals when the tops of the sampled pulses remain constant during the sample interval. In flat-top sampling, an arbitrary analog signal is sampled by a train of pulses having finite short pulse width occurring at uniform intervals. A flat-topped pulse has a constant amplitude established by the sample value of the signal at some point within the pulse interval. We have arbitrarily sampled the signal at the beginning of the pulse, retaining the amplitude of each rectangular pulse at the value of the analog signal, as shown in Figure 1.1.3.

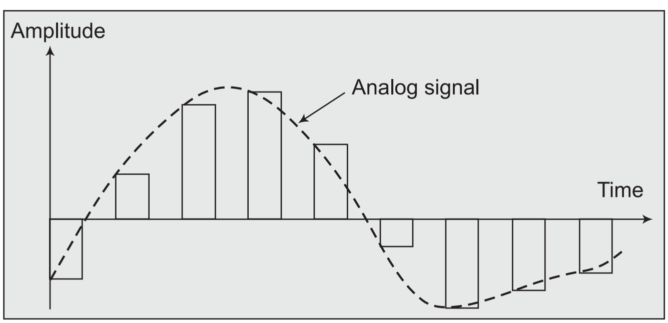


Figure 1.1.3 *Flat-top Sampling*

Flat-top sampling has the merit that it simplifies the design of the electronic circuitry used to perform the sampling operation. A *sample-and-hold circuit* is used to keep the amplitude constant during each pulse in flat-top sampling process.

Aperture Effect

In flat-top sampled signals, the high-frequency contents of the analog signal are lost which results into distortion known as the *aperture effect,* or *aperture distortion*. It is due to the presence of finite pulse width at the instant of sampling in pulse-amplitude modulated signal. In fact, the sampling process in flat-top sampling introduces aperture error because amplitudes of analog signal changes during the sample pulse width.

IMPORTANT!

In *flat-top sampling*, the analog signal cannot be recovered exactly by simply passing the samples through an ideal low-pass filter. It can be easily seen that the use of flat-top samples results into amplitude distortion. In addition, there is a delay by $T_b/2$, where T_b is the width of the pulse, which results into lengthening of the samples during transmission. At the receiver, amplitude distortion as well as delay causes errors in decoded signal. However, the distortion may not be large.

Application

In ideal, natural, or flat-top sampling, the sampling rate must be at least twice the highest frequency contained in the analog signal, according to the Nyquist criterion specified in the sampling theorem. In natural or flat-top sampling, the resulting distortion may be corrected by including an equalizer in cascade with the output low-pass filter.

1.1.3 Significance of Sampling Rate

Recall

The sampling rate must be fast enough so that at least two samples are taken during the period corresponding to the highest frequency spectral component present in the analog information signal. The minimum sampling rate is known as Nyquist rate.

Now, let us discuss practical difficulties in reliable reconstruction of the sampled analog signal in the process of sampling.

- If the sampling rate is larger than $2f_m$ then there is a gap between the highest frequency f_m of the spectrum of the baseband signal and the lower limit $(f_s - f_m)$ of the spectrum of the sampled signal, where f_s is the sampling frequency (i.e., Nyquist rate).
- This range from f_m to $(f_s - f_m)$ is called a *guard band* and is always required in practice, since a low-pass filter with infinitely sharp cutoff frequency is, of course, not realizable.

IMPORTANT!

An increase in the sampling rate above the Nyquist sampling rate increases the width of the guard band, thereby simplifying the design of reconstruction filter. Of course, this would also mean that higher bandwidth is required to transmit the sampled signal.

Example

When sampling is used for transmitting voice signals on telephone lines, the voice signal is limited to f_m = 3.4 kHz, while sampling frequency f_s is selected at 8 kHz (instead of 2 × 3.4 kHz = 6.8 kHz). The guard band is then 8 kHz – 6.8 kHz = 1.2 kHz.

ATTENTION

An interesting special case is the sampling of a sinusoidal signal having the frequency f_m. Here, all the signal power is concentrated precisely at the cut-off frequency of the low-pass reconstruction filter. So, it is recommended to use $f_s > 2f_m$ rather than $f_s = 2f_m$ so as to avoid the situation when all successive samples will be zero if an initial sample is taken at the moment the sinusoid passes through zero.

Aliasing

What happens if sampling rate is less than that of the Nyquist rate? When the sampling rate is reduced (sampling at too low a rate called *undersampling*), such that $f_s < 2f_m$, spectral components of adjacent samples will overlap and some information will be lost. This phenomenon is called *aliasing*.

Aliasing can be defined as the distortion in the sampled analog signal due to the presence of high-frequency components in the spectrum of the original analog signal. It is also known as *aliasing distortion*, or *foldover distortion*.

Define

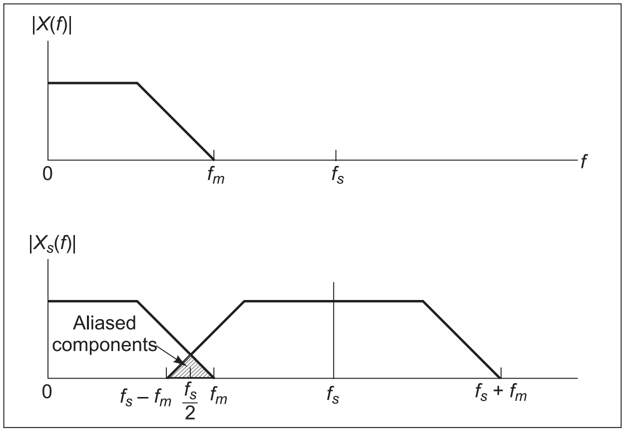


Figure 1.1.4 An Illustration of Aliasing

Note that an arbitrary rectangular waveform has been shown here for simplicity. If $f_s = 2f_m$, the resultant sampled signal is just on the edge of aliasing. In order to separate the signals sufficiently apart, the sampling frequency f_s should be greater than $2f_m$, as stated by the sampling theorem. If aliasing does take place, the interfering frequency component, called as *aliasing frequency*, will be at a frequency

Aliasing Frequency

$$f_a = f_s - f_m$$

where f_a is the frequency component of the aliasing distortion (Hz), f_s is the minimum Nyquist sampling rate (Hz), f_m is the maximum analog input (baseband) frequency (Hz).

It is quite clear that the use of a low-pass reconstruction filter, with its pass-band extending from $-f_s/2$ to $+f_s/2$, where f_s is the sampled frequency, does not yield an undistorted version of the original analog information signal. It results into the portions of the frequency-shifted replicas folded over into the desired frequency spectrum. So we can say that the absolute error between the original analog signal and the signal reconstructed from the sequence obtained by sampling is termed as *aliasing distortion* or *foldover distortion*, or simply *aliasing error*. **In a nutshell, if $f_s < 2f_m$, aliasing will occur.**

Aliasing Distortion

Due to aliasing problem, not only we lose all the components of frequencies above the folding frequency, $f_a/2$, but aliased frequency components reappear as lower frequency components. Such aliasing destroys the integrity of the frequency components below the folding frequency.

Application

1.1.4 Anti-aliasing Filter

An anti-aliasing filter is a *low-pass filter* of sufficient higher order which is recommended to be used prior to sampling. This will attenuate those high-frequency spectral components of the analog signal that do not contribute significantly to the information content of the analog signal.

What is anti-aliasing Filter?

Importance of anti-aliasing filter

There are certain corrective measures for sampling of an analog signal in order to overcome the practical difficulties encountered in sampling and recovery process. A practical procedure for the sampling of an analog signal whose frequency spectrum is not strictly band-limited involves the use of the anti-aliasing filter, also known as *pre-alias filter*.

- The filtered analog information signal (by pre-alias filter) is recommended to be sampled at a rate *slightly higher* than that determined by the Nyquist rate, that is, greater than $2 f_m$ Hz where f_m Hz is the 3 dB high cut-off frequency of the pre-alias filter.
- With such a sampling rate, there are frequency gaps each of width $(f_s - 2f_m)$ Hz between the frequency-shifted replicas of the analog signal. As mentioned earlier, these frequency gaps are generally referred as *guard bands*.

In practice, anti-aliasing filter is used at the front end of the impulse modulator (used for sampling). This enables to exclude the frequency components greater than the required maximum frequency component of the information signal. Thus, the application of sampling process allows the reduction of continuously varying information waveform to a finite limited number of discrete levels in a unit time interval. Figure 1.1.5 shows the use of an anti-aliasing filter to minimize aliasing distortion.

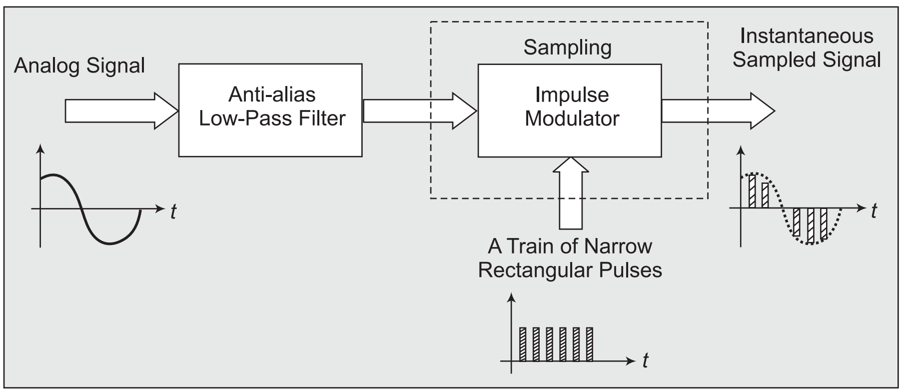


Figure 1.1.5 Minimizing Aliasing Distortion by using a Pre-alias Filter

IMPORTANT!

It is emphasized that the anti-aliasing operation must be performed before the analog signal is sampled. Accordingly, the reconstruction filter (low-pass filter) at the receiver end is designed to satisfy the following characteristics:

- The passband of the reconstruction filter should extend from zero to f_m Hz.
- The amplitude response of the reconstruction filter rolls off gradually from W Hz to $(f_s - 2 f_m)$ Hz.
- The guard band has a width equal to $(f_s - 2 f_m)$ Hz which is non-zero for $(f_s > 2 f_m)$ Hz.

Application

An anti-aliasing filter also helps reduce noise. Generally, noise has a wideband spectrum. Without anti-aliasing, the aliasing phenomenon itself will cause the noise components outside the signal spectrum to appear within the signal spectrum. Anti-aliasing suppresses the entire noise spectrum beyond $f_s/2$.

SOLVED EXAMPLE 1.1.8 **Aliasing Frequency**

A baseband signal having maximum frequency of 30 kHz is required to be transmitted using a digital audio system with a sampling frequency of 44.1 kHz. Estimate the aliasing frequency component available at the output.

Solution For given value of f_s = 44.1 kHz, and f_m= 30 kHz, we observe that $f_s < 2 f_m$. This would result into aliasing frequency which is given as

$$f_a = f_s - f_m = 44.1 \text{ kHz} - 30 \text{ kHz} = 14.1 \text{ kHz}$$

Thus, the output would have the original baseband frequency of 30 kHz as well as aliasing frequency of 14.1 kHz.

SOLVED EXAMPLE 1.1.9 Aliasing Frequency

For a maximum audio input frequency of 4 kHz, determine the minimum sampling rate and the aliasing frequency produced if a 5 kHz input signal were allowed to enter the sampler circuit.

Solution For given maximum audio input analog frequency of f_m = 4 kHz, the recommended sampling rate is given by Nyquist's sampling theorem as $f_s \geq 2 f_m$.
Therefore, $f_s \geq 8$ kHz **Ans.**
If a 5 kHz audio signal enters the sampler circuit, an aliasing frequency f_a of 3 kHz (8 kHz – 5 kHz) is produced. Hence, f_a = 3 kHz **Ans.**

SOLVED EXAMPLE 1.1.10 Practical Sampling Rate

Realizable filters require a nonzero bandwidth for the transition between the passband and the required out-of-band attenuation, called the transition bandwidth. Consider 20% transition bandwidth of the anti-aliasing filter used in a system for producing a high-quality digitization of a 20-kHz bandwidth music source. Determine the reasonable sampling rate.

Solution For given f_m = 20 kHz, and 20% transition bandwidth of the anti-aliasing filter,

Transition bandwidth = 20 kHz × 0.2 = 4 kHz

Therefore, the practical Nyquist sampling rate, $f_s \geq 2.2 f_m$

The practical Nyquist sampling rate $f_s \geq 44.0$ ksamples/second

The reasonable sampling rate for the digital CD audio player = 44.1 ksamples/second
Standard sampling rate for studio-quality audio = 48 ksamples/second **Ans.**

SOLVED EXAMPLE 1.1.11 Spectrum of Sampled Signal

Consider an analog signal given by $s(t) = 2 \cos(2\pi\, 100t) \cos(2\pi\, 10t)$ is sampled at the rate of 250 samples per seconds. Determine the maximum frequency component present in the signal, Nyquist rate, and cut-off frequency of the ideal reconstruction filter so as to recover the signal from its sampled version. Draw the spectrum of the resultant sampled signal also.

Solution The analog signal, $s(t) = 2 \cos(2\pi\, 100t) \cos(2\pi\, 10t)$ (Given)

Using the trigonometric identity $2 \cos A \cos B = \cos(A + B) + \cos(A - B)$, we have

$$s(t) = \cos(2\pi\, 110\, t) + \cos(2\pi\, 90\, t)$$

$\Rightarrow$ $f_1 = 110$ Hz; $f_2 = 90$ Hz

$\therefore$ the maximum frequency present in analog signal, f_m = 110 Hz **Ans.**

The sampling frequency, or Nyquist rate, $f_s \geq 2 f_m$

$\Rightarrow$ $f_s \geq 220$ Hz **Ans.**

The cut-off frequency of the ideal reconstruction filter should be more than the Nyquist rate. Therefore, $f_s > 220$ Hz. **Ans.**

Figure 1.1.6 shows the spectrum of the resultant sampled band-pass signal.

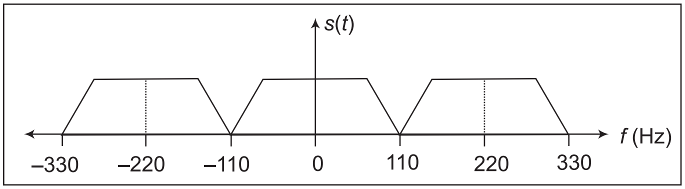


Figure 1.1.6 Spectrum of Sampled Band-pass Signal

It may be noted here that an arbitrary waveform has been chosen for simplicity purpose only.

SOLVED EXAMPLE 1.1.12 **Sampling Rate versus Nyquist Rate**

Let the maximum frequency component (f_m) in an analog signal be 3.3 kHz. Illustrate the frequency spectra of sampled signals under the following relationships between the sample frequency, f_s and the maximum analog signal frequency, f_m.

(a) $f_s = 2 f_m$
(b) $f_s > 2 f_m$
(c) $f_s < 2 f_m$

Solution

(a) For a given value of f_m = 3.3 kHz,

Given, $f_s = 2 f_m$; $\Rightarrow$ $f_s = 2 \times 3.3$ kHz = 6.6 kHz

Figure 1.1.7 illustrates the frequency spectra of sampled signals for $f_s = 2 f_m$. (In ideal sampling, spectrum will repeat for every f_s Hz, as shown).

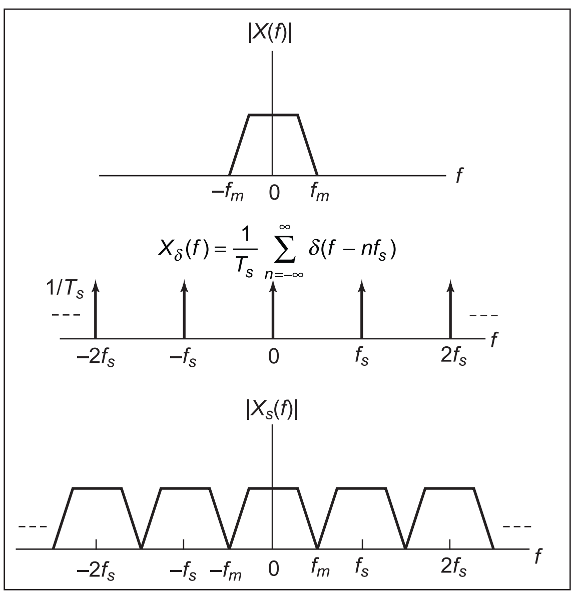


Figure 1.1.7 Frequency Spectra of Sampled Signals for $f_s = 2 f_m$

It may be noted here that any arbitrary waveform can be drawn with equal spacing between their centers.

(b) For given value of f_m= 3.3 kHz, f_s is greater than 2 f_m (i.e., 6.6 kHz).
Let f_s = 8 kHz,
Therefore, guard band = $f_s - 2f_m$ = (8 – 2 × 3.3) kHz = 1.4 kHz
Figure 1.1.8 illustrates the frequency spectra of sampled signals for $f_s > 2f_m$

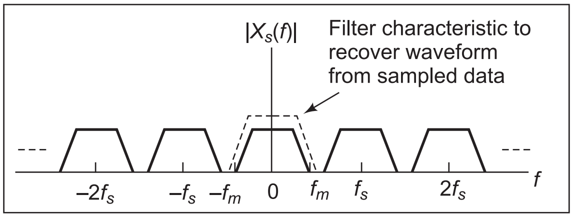


Figure 1.1.8 Frequency Spectra of Sampled Signals for $f_s > 2f_m$

(c) For given value of f_m= 3.3 kHz, f_s is less than 2 f_m (6.6 kHz).
Let f_s = 6 kHz,
Therefore, overlap band = $[f_m - (f_s - f_m)]$ = [3.3 – (6 – 3.3)] kHz = 0.6 kHz
Figure 1.1.9 illustrates the frequency spectra of sampled signals for $f_s < 2f_m$

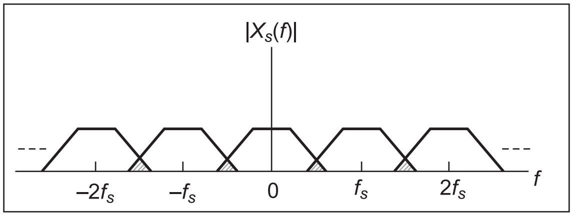


Figure 1.1.9 Frequency Spectra of Sampled Signals for $f_s < 2f_m$

The extent of overlapping between adjacent waveforms is clearly seen.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 1.1.13 An analog signal having a single frequency as 4 kHz is sampled with a 7 kHz signal. Describe its frequency spectrum. What will be the output if the sampled signals are passed through a low-pass filter having cut-off frequency at 3.5 kHz?

Ex 1.1.14 The specified voice spectrum is 300 Hz–3400 Hz. The sampling frequency used is 8 kHz. In practice, the frequency spectrum of human voice extends much beyond the highest frequency necessary for communication. Let the input analog information signal contain a 5 kHz frequency component also. What would happen at the output of the sampler? How can this problem be prevented?

Ex 1.1.15 An analog signal having a single frequency as 4 kHz is sampled with a 10 kHz signal. Describe its frequency spectrum. What will be the output if the sampled signals are passed through a low-pass filter having cut-off frequency at 5 kHz?

1.1.5 Application of Sampling Theorem — PAM/TDM

Recall

The *Sampling Theorem* allows us to replace a continuous time signal (analog signal) by a discrete sequence of pulses in the time domain. This amounts to processing of a continuous time signal by discrete sequence of number of pulses—the aspect which is quite important in signal analysis and processing of analog signals using digital filters.

Pulse Modulation

One of the most significant application of the Sampling Theorem is in the field of communication in which an analog signal is sampled by a train of pulses, and sample values are used to modify certain parameters of a periodic pulse train. This leads us to various analog pulse modulation techniques such as

- ***Pulse Amplitude Modulation*** **(PAM)** by varying the amplitude of the pulses in proportion to the instantaneous value of the analog signal.
- ***Pulse Width Modulation*** **(PWM)** by varying the width of the pulses in proportion to the instantaneous value of the analog signal.
- ***Pulse Position Modulation*** **(PPM)** by varying the position (starting point on time scale) of the pulses in proportion to the instantaneous value of the analog signal.

In all these cases, we detect the information of the pulse-modulated signal and reconstruct the original analog signal.

PAM/TDM

One significant advantage of using the PAM technique is that it permits the simultaneous transmission of several baseband signals on a time-sharing basis, known as *PAM—time division multiplexing* (PAM-TDM). In fact, PAM is suitable for both TDM and digital representation of analog signals. Multiplexing of several PAM signals is possible because various signals are kept distinct and are separately recoverable because they are sampled at different times. For example, Time-Division Multiplexing (TDM) systems use PAM for multiplexing many baseband analog signals.

Dear student... *You can read more about other forms of TDM techniques in Unit 2.*

What next

Let us now try to understand the concept and design of PAM/TDM systems with the help of following examples.

SOLVED EXAMPLE 1.1.16 **Design of PAM/TDM**

Four analog information signals are required to be transmitted by a PAM/TDM system. One out of four signals is band-limited to 3 kHz, whereas the remaining three signals are band-limited to 1 kHz each. Design a TDM scheme where each information signal is sampled at its Nyquist rate.

Solution The PAM/TDM system can be used to multiplex analog signals. The analog signals are sampled using pulse amplitude modulation (PAM), prior to time-division multiplexing. For the given data, Figure 1.1.10 shows the design of the PAM/TDM system.

The multiplexer (MUX) is a single-pole rotating mechanical or electronic switch or *commutator*, rotating at f_s (sampling frequency) rotations per second, such that $f_s \geq 2 f_m$ where f_m is the highest signal frequency present in all the channels.

We know that Nyquist rate, $f_s = 2 f_m$, where f_m is the highest frequency present in the analog information signal after band-limiting. Therefore,

- For the first band-limited signal, $s_1(t)$ of given $f_{m1} = 3$ kHz; $f_{s1} = 6$ kHz
- For the second band-limited signal, $s_2(t)$ of given $f_{m2} = 1$ kHz; $f_{s2} = 2$ kHz

- For the third band-limited signal, $s_3(t)$ of given f_{m3} = 1 kHz; f_{s3} = 2 kHz
- For the fourth band-limited signal, $s_4(t)$ of given f_{m4} = 1 kHz; f_{s4} = 2 kHz

If the sampling commutator rotates at f_s= 2000 rotations per second then the information signals $s_2(t)$, $s_3(t)$, and $s_4(t)$ will be sampled at their Nyquist rate, and the signal $s_1(t)$ has to be sampled at its f_{s1} which is three times higher than that of the other three signals. In order to achieve this, $s_1(t)$ should be sampled three times in one rotation of the commutator.

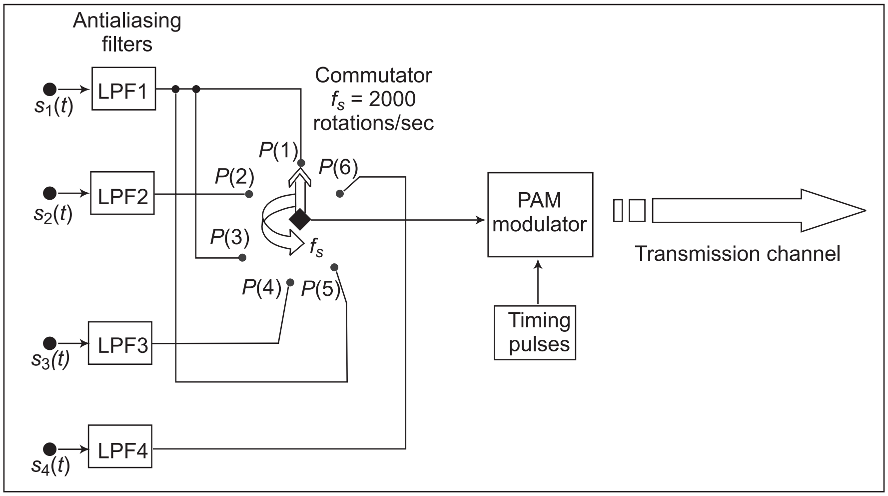


Figure 1.1.10 Design of PAM/TDM System for Example 1.1.16

So number of poles of commutator switch connected to $s_1(t)$ = 3
Thus, total number of poles of commutator switch connected to all signals = 6
Recommended speed of the commutator = 2000 rotations per second
Number of samples per second for signal $s_1(t)$ = 3 × 2000 = 6000 samples/second
Number of samples per second for signal $s_2(t)$ = 1 × 2000 = 2000 samples/second
Number of samples per second for signal $s_3(t)$ = 1 × 2000 = 2000 samples/second
Number of samples per second for signal $s_4(t)$ = 1 × 2000 = 2000 samples/second
Hence, signaling rate = 6000 + 2000 + 2000 + 2000 = 12000 samples/seccond **Ans.**
We know that minimum transmission bandwidth, B_{TDM} = (1/2) × signaling rate
Hence, minimum transmission bandwidth, B_{TDM} = (1/2) × 12000 = 6000 Hz
Alternatively, minimum transmission bandwidth, $B_{TDM} = f_{m1} + f_{m2} + f_{m3} + f_{m4}$
Hence, minimum transmission bandwidth, B_{TDM} = 6 kHz **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 1.1.17 Six analog information signals, each band-limited to 4 kHz, are required to be transmitted using PAM/TDM system. Determine the Nyquist rate, signaling rate, and minimum transmission bandwidth of the PAM/TDM channel.

Ex 1.1.18 Four independent band-limited analog information signals have bandwidths of 300 Hz, 600 Hz, 800 Hz, and 800 Hz, respectively. Each signal is sampled at Nyquist rate, and the samples are time-division multiplexed and transmitted. Determine the transmitted sample rate in Hz.

MATLAB simulation exercises on generation of sampled signal for analog signals using the sampling theorem,
Scan the QR code given here OR visit: http://qrcode.flipick.com/index.php/143

Self-Assessment Exercise linked to LO 1.1

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/144

Q1.1.1 A video signal has a bandwidth of 4.5 MHz. Determine the sampling rate and sampling intervals for ○●●
(a) minimum sampling
(b) 10% undersampling
(c) 20% oversampling

Q1.1.2 An analog signal having single frequency as 3 kHz is sampled with 5 kHz signal. Describe its frequency spectrum. What will be the output if the sampled signals are passed through a low-pass filter having cutoff frequency at 2.5 kHz? ○○●

Q1.1.3 What are the pros and cons of exceeding the sampling rate beyond the required Nyquist rate? ●●●

Q1.1.4 There are number of benefits of digital representation of analog signals such as voice, image, and video. Highlight them. Are there any drawbacks too? ○○●

Q1.1.5 Distinguish between natural and flat-top sampling. ○○●

Q1.1.6 Why is there always a defined upper limit to the analog information signal frequency that can be transmitted in a digital communication system? What is the significance of the Nyquist sampling rate? ○●●

Note ○○● Level 1 and Level 2 Category
○●● Level 3 and Level 4 Category
●●● Level 5 and Level 6 Category

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 1.1: *Understand various practical aspects of sampling such as methods of sampling, Nyquist sampling theorems, aliasing, PAM/TDM application.*

LO 1.2

1.2 PULSE CODE MODULATION (PCM)

Just a Minute...

In the conventional sense, the complete process of PCM is not a modulation technique in which the term *modulation* usually refers to the alteration of some characteristics (amplitude, frequency, or phase) of a high-frequency carrier signal (sinusoidal or pulse train) in accordance with instantaneous value of an information-bearing analog signal. However, the only part of PCM that confirms to this basic definition of modulation is the process of sampling in which the analog information signal is sampled (modulated) with a continuous train of narrow rectangular pulses, known as *impulse modulation*.

An analog signal is usually sampled to get a discrete-time signal and then quantized to get a discrete-level signal as well. Thus, the combined operation of sampling and quantizing generate a *quantized pulse-amplitude modulation* waveform, that is, a train of uniformly spaced narrow pulses whose amplitudes are restricted to a finite number of discrete levels. Then, we represent each quantized level by an equivalent decimal number (also called code number), which is further converted into its binary representation (a sequence of 1s and 0s)—the process known as *encoding*. In a nutshell, an analog signal is converted to a PCM signal through three essential processes—*sampling, quantizing,* and *encoding*. Various aspects of PCM process are illustrated here.

What We Discuss Here .....

1.2.1 Block Diagram of PCM

Figure 1.2.1 shows a simplified block diagram of a single-channel one-way PCM system.

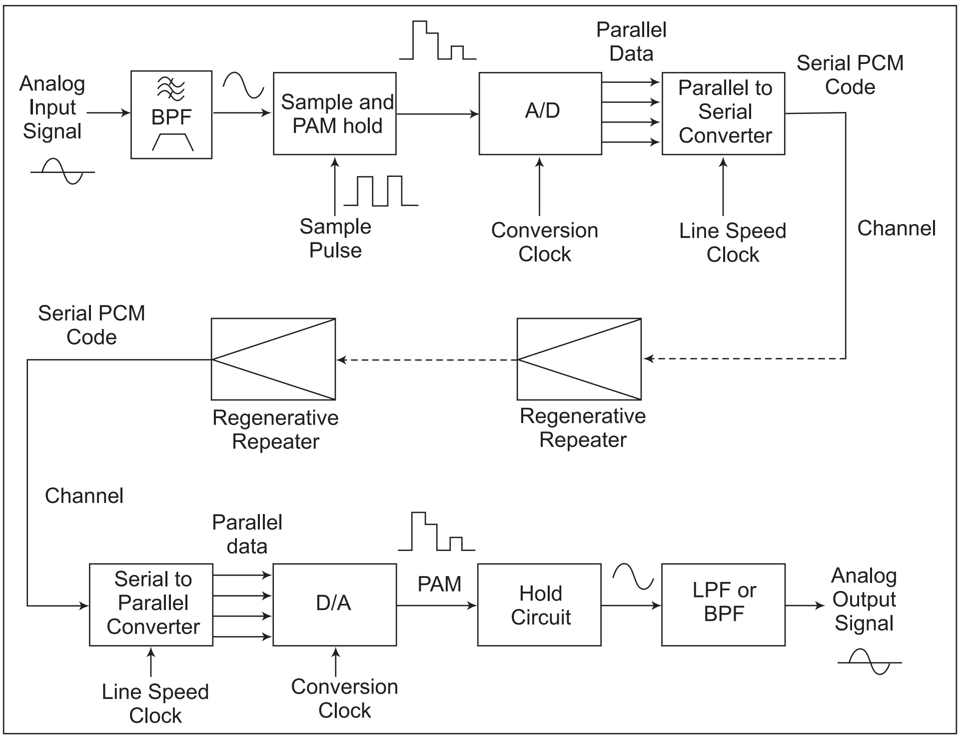


Figure 1.2.1 Block Diagram of a PCM System

Functional Description

The **band-pass filter (BPF)**, or low-pass filter, limits the frequency of the input analog signal to desired baseband signal frequency range.

- The **sample-and-hold** circuit periodically samples the analog input signal and converts these samples to a multi-level PAM signal.
- The **analog-to-digital (A/D) converter** performs the function of the quantizer and encoder. Its output is a sequence of binary symbols (also known as code words) for each sample. Each symbol consists of a train of pulses in which each pulse may represent a binary digit.

- PCM codes are then converted to serial binary data in the **parallel-to-serial converter** and then presented to the transmission medium as serial digital pulses. Thus, the signal transmitted over the communication channel in a PCM system is referred to as a digitally encoded signal.
- When the digitally encoded signal is transmitted, noise is added during the transmission along the channel.
- The transmission channel **regenerative repeaters** are placed at prescribed distances to regenerate the digital pulses and enable to remove interference, if any, due to channel noise.
- In the PCM receiver, the **serial-to-parallel converter** converts serial pulses received from the transmission line to parallel PCM codes.
- The **digital-to-analog (D/A) converter** generates sequence of quantized multi-level sampled pulses, resulting in reconstituted PAM signal.
- The **hold circuit** is basically a low-pass filter (LPF) to reject any frequency component lying outside its baseband. It converts the recovered PAM signal back to its original analog form.

Application

PCM belongs to a class of signal coders in which an analog signal is approximated by mimicking the amplitude-versus-time waveform; hence known as waveform coders. In principle, waveform coders are designed to be signal-independent. PCM is used in digital telephone systems (trunk lines) and is also the standard form for digital audio in computers and various compact disc formats, digital videos, etc. PCM is the preferred method of communications within Integrated Services Digital Network (ISDN) because with PCM it is easy to combine digitized voice and digital data into a single, high-speed digital signal and transmit it over either coaxial or optical fiber cables.

1.2.2 PCM Sampling

Let an arbitrary analog signal $s(t)$ be applied to a pulse modulator circuit (may be a high-speed transistor switching circuit) controlled by a pulse signal $c(t)$. The pulse signal consists of an infinite succession of rectangular pulses of amplitude A, pulse width T_b, and occurring with time interval T_s. Figure 1.2.2 depicts a simplified block diagram of pulse sampler.

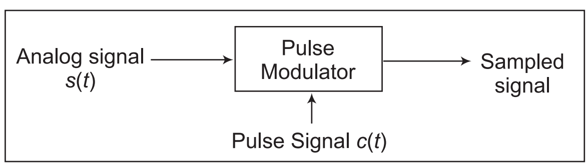


Figure 1.2.2 A Simplified Block Diagram of Pulse Sampler

The sampled signal $x(t)$ consists of a sequence of positive and negative pulses of predetermined duration T_b, taken regularly at the rate $f_s = \frac{1}{T_s}$, as shown in Figure 1.2.3.

Assume that the analog signal $s(t)$ does not contain any frequency component outside the frequency range from $-f_m$ to $+f_m$, and that the sampling rate $f_s > 2f_m$ (Nyquist criterion) so that there is no aliasing. The effect of using ordinary pulses of finite duration on the spectrum of a sampled signal is illustrated in Figure 1.2.4.

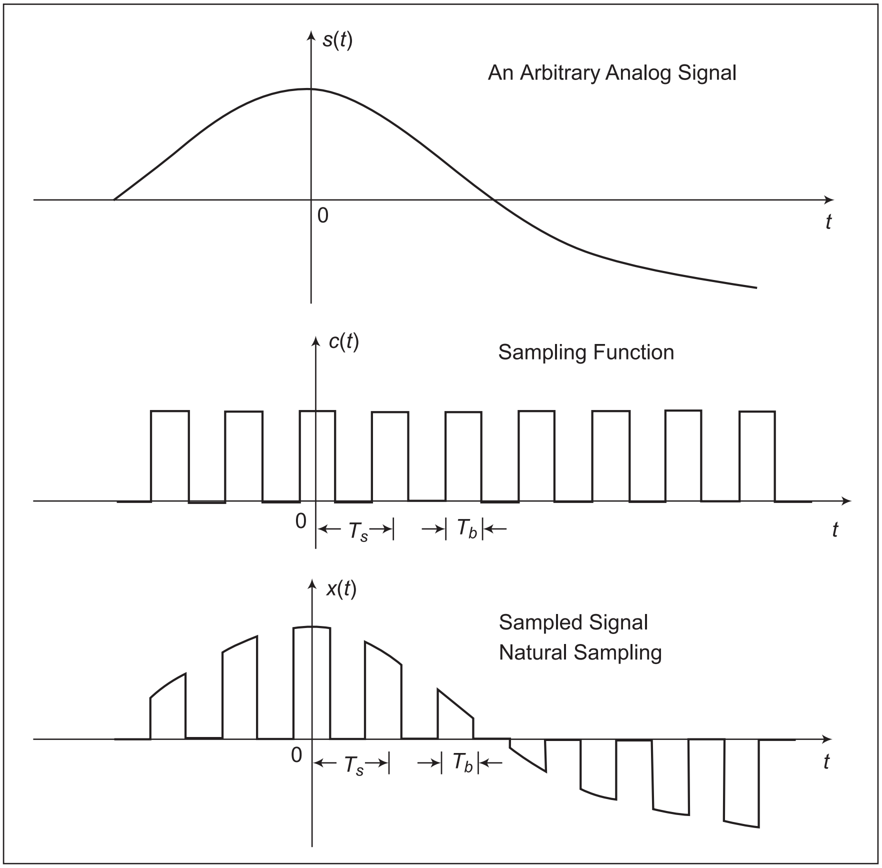


Figure 1.2.3 The Process of Natural Sampling

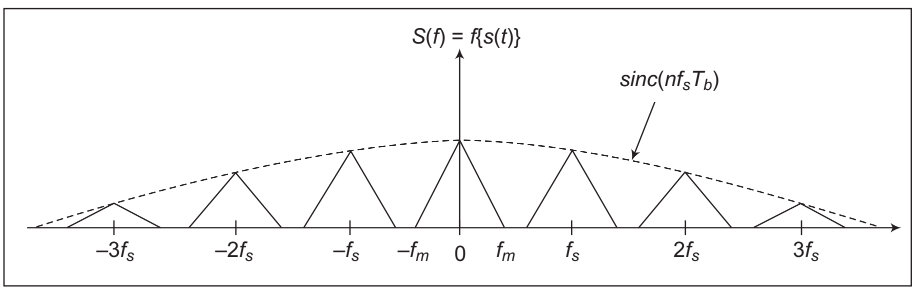


Figure 1.2.4 Spectrum of Sampled Signal using Natural Sampling

We have earlier discussed that when the sampling rate exceeds Nyquist rate, there is an increase in the width of the guard band between two adjacent samples. This makes the filtering operation easier. But it extends the minimum bandwidth required for transmitting the sampled signal.

1.2.3 Quantization of Sampled Signal

Recap

In our discussion for conversion of an analog signal into its equivalent digitized form, we have covered an important step of sampling the analog signal along its time axis, i.e., discrete-time representation of analog signal.

Quantizing—Next Step to Sampling

Now we need to discretize the amplitude axis too. The simple reason for doing this is that an analog signal is characterized by an amplitude that can take on an infinite number of values over a continuous range, whereas amplitude of a digital signal can take on only a finite number of values. The conversion of a analog signal into a discrete digital signal can be done by the process of sampling and *quantizing*.

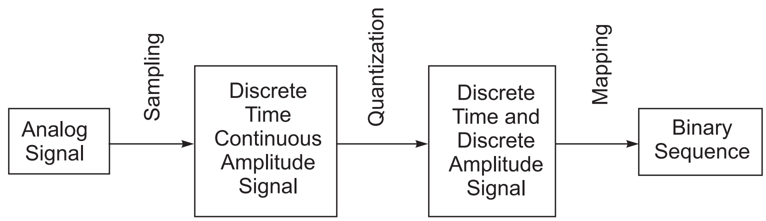


Figure 1.2.5 Quantization—Discrete Time & Discrete Amplitude Signal

The value of the sampled analog signal can be rounded off to one of the nearest permissible numbers, known as *quantized levels*. Thus, the process of converting an infinite number of possible levels of a sampled analog signal into finite number of discrete approximate levels is called *quantization*. The operation of quantization is represented in Figure 1.2.6.

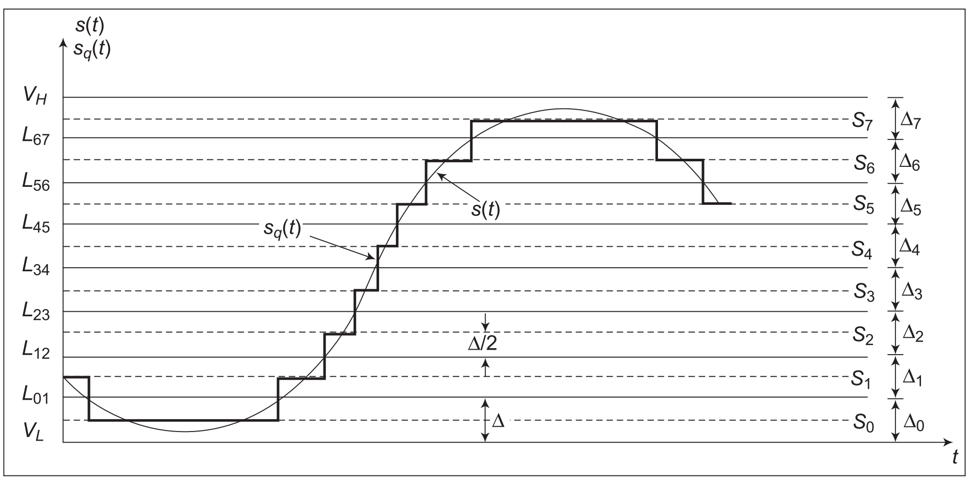


Figure 1.2.6 Operation of Quantization

Step-by-step Description

An analog signal $s(t)$ has peak-to-peak amplitude excursion from V_L (lowest amplitude level) to V_H (highest amplitude level).

- Let us divide this total range of amplitude values into L equal intervals, each interval having size Δ. Accordingly Δ, called the *step size*, is given as $\Delta = \dfrac{V_L - V_H}{L}$.

- In this figure, $L = 8$, i.e., V_L to L_{01} (marked as Δ_0), L_{01} to L_{12} (marked as Δ_1), …, L_{67} to V_H (marked as Δ_7), where L_{01}, L_{12}, …. L_{67} are termed *transition levels*.
- In the middle of each of these eight steps (i.e., Δ_0, Δ_1, … Δ_7,), we locate the *quantization levels* marked as s_0, s_1, …, s_7.
- Now we generate the quantized signal, $s_q(t)$ in such a way that whenever the amplitude of analog signal $s(t)$ is in between the range of two adjacent quantization levels, the quantized signal $s_q(t)$ maintains the previously held constant level.
  For example,
  - Whenever the amplitude of $s(t)$ lies in the range of Δ_0, the quantized signal $s_q(t)$ maintains the constant level at s_0;
  - Whenever the amplitude of $s(t)$ lies is in the range of Δ_1, the quantized signal $s_q(t)$ maintains the constant level at s_1; and so on.
- Thus, the quantized signal $s_q(t)$ will at all times be found at one of the quantized levels s_0, s_1, …, s_7. In other words, we can say that, at every instant of time, $s_q(t)$ either does not change at all with time or it changes by step size Δ only.
- Although the adjacent quantization levels are each separated by Δ, even the separation of the extremes V_L and V_H each from its nearest quantization level is only $\Delta/2$.

Quantization Noise

Therefore, we see that the quantized signal is an approximation to the original analog signal. At any instant of time, the difference in the values of analog signal $s(t)$ and its quantized signal $s_q(t)$, i.e., $s(t) - s_q(t)$ has a magnitude which is equal to or less than $\Delta/2$. This difference can be regarded as noise, known as *quantization noise* or *quantization error*. A quantizer is memoryless in the sense that the quantizer output is determined only by the value of a corresponding input sample only.

Effect of Quantization Noise

We know that the transmitted signal gets attenuated as well as corrupted by the additive noise during transmission through communication channel. As long as the channel noise has an instantaneous amplitude less than the quantization noise ($\Delta/2$), the noise will not appear at the output of the digital regenerator (a combination of a quantizer and an amplifier, also known as repeater) installed between transmitter and distant receiver. However, the statistical nature of noise is such that even if the average noise magnitude is much less than $\Delta/2$, there is always a finite probability of occurrence of error. Hence, the received signal is not an exact replica of the original signal due to errors caused by quantization noise and additive channel noise.

Uniform Quantization

So far we have discussed uniform quantization in which the step size Δ was kept uniform. Thus, *uniform quantization* is a process in which the quantization levels are uniformly spaced over the complete input amplitude range of the analog signal. That means the uniform quantizer has a linear characteristics. The largest possible quantization noise is one-half the difference between successive levels, i.e., $\pm\Delta/2$.

Need of Non-uniform Quantization

For transmission of digitized speech signals, the quantizer has to accommodate input signals with widely varying power levels. For example, the range of voltage amplitude levels covered by normal speech signals, from the peaks of loud speech levels to the lows of weak speech levels, is on the order of $10^4 : 1$ (equivalent to power ratio of 40 dB). We can say that in practice, the *signal-to-quantization noise ratio*, which is a measure of the quality of the received signal, does not remain same over the complete input range. This means that the signal-to-quantization noise ratio varies with the signal level and is higher for large signal levels. So it is highly desirable that signal-to-quantization noise ratio should remain essentially constant for a wide range of input signal levels. The amount of quantization noise can be decreased by increasing the number of levels, but it also increases the complexity at the receiver for decoding these levels precisely. The only solution to have a constant signal-to-quantization noise ratio is to adjust the step-size in accordance with the input signal amplitude levels. This is known as *non-uniform quantization*.

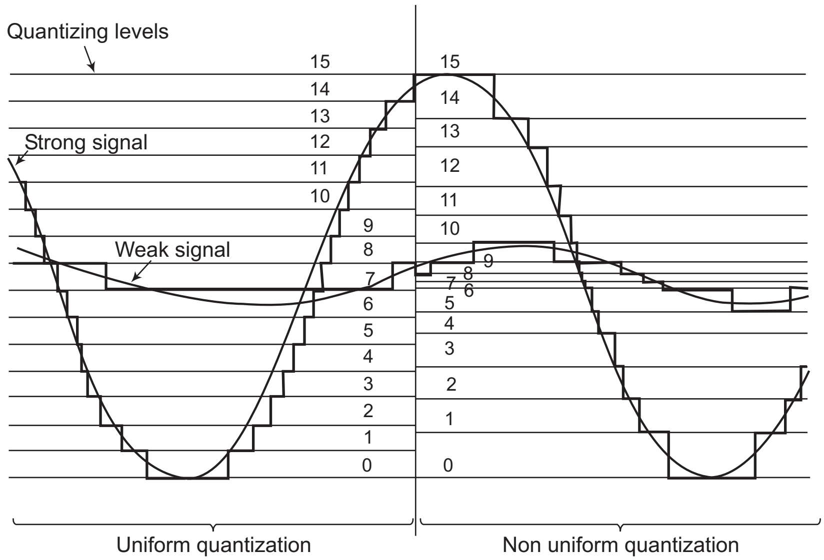


Figure 1.2.7 Uniform vs Non-uniform Quantization

Advantages of Non-uniform Quantization

Thus, in non-uniform quantization, the spacing between the quantization levels is not uniform and step size varies in accordance with the relative amplitude level of the sampled value. This results into an advantage of having higher average signal to quantization noise power ratio value than that of possible in the uniform quantizer. Moreover, the RMS value of the quantizer noise power of a non-uniform quantizer is substantially proportional to the sampled value and hence the quantization noise is also reduced. Since the signal-to-quantization noise remains essentially constant for a wide range of input power levels, the non-uniform quantization is said to be robust. In other words, *robust quantization* requires that the step-size must be small for low amplitude levels and large for high amplitude levels of analog signal. The provision for such robust performance necessitates the use of a *non-uniform quantizer*.

Companding

The non-uniform quantization technique employs an additional logarithmic amplifier before processing the sampled speech signals by a uniform quantizer. The operation of a non-uniform quantizer is equivalent to passing the analog signal through a compressor and then applying the compressed signal to a uniform quantizer at transmitter end. At the receiver, a device with a characteristic complementary to the compressor, called *expander*, is used to restore the signal samples to their correct relative level. The combination of a compressor and an expander is called a *compander*. So, *companding* is the process of compressing the signal at transmitter end and expanding it at the receiver end to achieve non-uniform quantization. An arrangement of companding or robust quantization is shown in Figure 1.2.8.

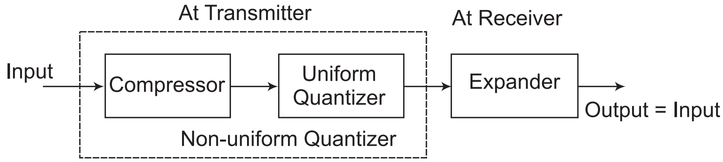


Figure 1.2.8 Block Diagram of Companding

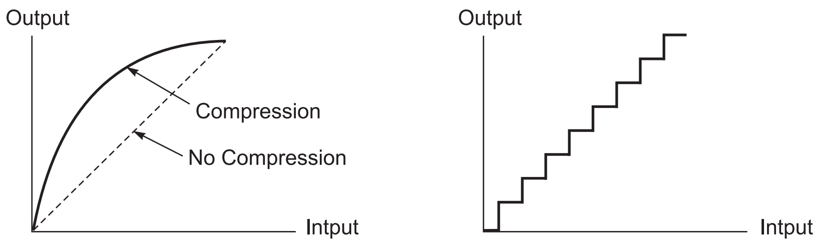


Figure 1.2.9 Characteristics of Compressor and Uniform Quantizer

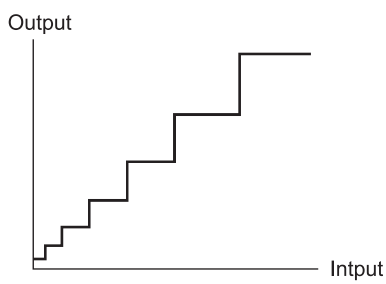


Figure 1.2.10 Characteristics of Non- Uniform Quantizer

μ-law Companding

ITU-T has recommended the *μ-law* companding standard for use in North America and Japan. The compression parameters, μ determines the degree of compression. In the μ-law companding, the compressor characteristics are continuous, approximating a linear dependence for low input levels and a logarithmic one for high input levels. The compression characteristics for μ-law (for positive amplitudes) is given by

$$V_{out} = V_{max} \frac{\ln\left(1+\mu \frac{V_{in}}{V_{max}}\right)}{\ln(1+\mu)}$$

where V_{out} is the compressed output amplitude level in volts

V_{max} is the maximum uncompressed analog input amplitude level in volts

μ is the parameter used to define the amount of compression (unit less)

V_{in} is the amplitude of input signal at a particular instant of time in volts

IMPORTANT!

It is observed that the value $\mu = 0$ corresponds to uniform quantization. For a relatively constant signal-to-quantization ratio and a 40 dB dynamic range, the value of $\mu \geq 100$ is required (suitable for 7-bit or 128-level PCM encoding). An optimum value of $\mu = 255$ has been used for all North American 8-bit or 256-level digital terminals.

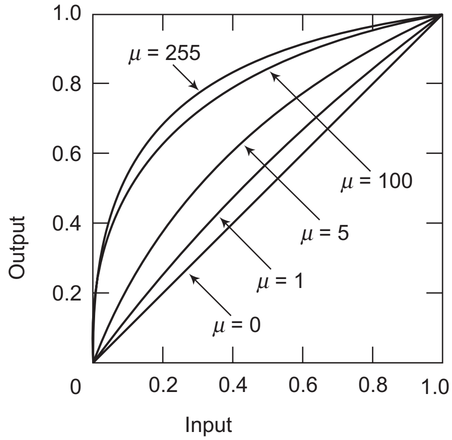


Figure 1.2.11 µ-law Characteristics

A-law Companding

ITU-T has recommended the *A-law* companding standard for use in Europe and rest of the world except North America and Japan. The compression parameters, A determines the degree of compression. The compression characteristics for A-law is given as

$$V_{out} = V_{max}\left[\frac{A\dfrac{V_{in}}{V_{max}}}{1+A}\right] \qquad \text{for} \quad 0 \le \frac{V_{in}}{V_{max}} \le \frac{1}{A}$$

$$V_{out} = \frac{1+\ln\left[A\dfrac{V_{in}}{V_{max}}\right]}{1+\ln A} \qquad \text{for} \quad \frac{1}{A} \le \frac{V_{in}}{V_{max}} \le 1$$

where V_{out} is the compressed output amplitude level in volts
V_{max} is the maximum uncompressed analog input amplitude level in volts
A is the parameter used to define the amount of compression (unit less)
V_{in} is the amplitude of input signal at a particular instant of time in volts

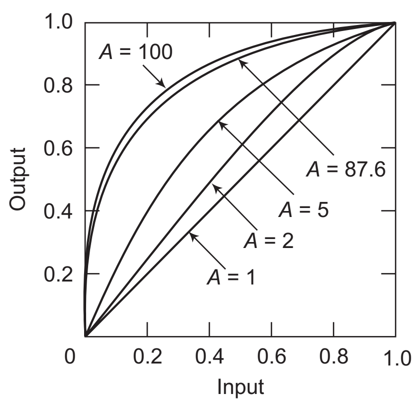
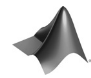


Figure 1.2.12 A-law Characteristics

The value $A = 1$ corresponds to uniform quantization. A typical value of $A = 87.6$ gives comparable results and has been standardized by the ITU-T. **IMPORTANT!**

PAM signals are never used directly for transmission, as the receiver circuits will have to handle a large number of finite voltage levels, equal to the number of quantization levels. Instead, these quantization levels are binary encoded. This gives resultant signals only two voltage levels, irrespective of the number of quantization levels. This is the pulse code modulation system which can be handled by very simple circuits. **ATTENTION**

MATLAB simulation exercises on companding,
Scan the QR code given here OR visit: http://qrcode.flipick.com/index.php/149



1.2.4 Encoding of a Quantized Sampled Signal

Recap

We have just discussed that PCM is a type of signal encoding technique in which the analog information signal is sampled (*sampling*) and the amplitude of each sample is approximated to the nearest one of a finite set of discrete levels (*quantization*), so that both amplitude and time are represented in discrete form. With PCM, the continuously varying analog signals are converted into pulses of fixed amplitude and fixed duration.

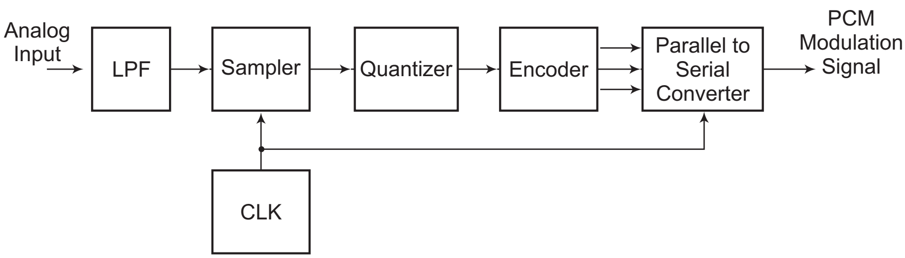


Figure 1.2.13 PCM—Functional Blocks

PCM Encoding

In fact, PCM is a binary system where a pulse or no pulse within a prescribed time slot represents either a logic 1 or logic 0 condition. For a binary PCM, we assign a distinct group of n binary digits (bits) to each of the L-quantization levels. As an instance, in a binary code, each codeword consisting of n number of bits will represent total 2^n distinct patterns. That is, for $n = 4$ bits, there will be 16 distinct patterns, ranging from 0000 to 1111, each can represent a unique quantized sample. Thus, the number of quantized levels is related to the number of bits as

$$L = 2^n \qquad \text{or} \qquad n = \log_2 L$$

where L is the number of quantized levels, and n is the number of bits per sample. It shows that L must be chosen as power of 2.[2]

Example

Let the peak-to-peak range of the analog signal is ±4V and the number of quantization levels is 8. The magnitude of seven consecutive samples is found to be 1.3 V, 3.6 V, 2.3 V, 0.7 V, –0.7 V, –2.4 V, and –3.4 V.

[2] The magnitude levels of the sampled signal are quantized which are invariable an approximation of analog signal, followed by digital encoding. The process of sampling, quantization and encoding leads us to the most important form of digital modulation technique known as ***pulse code modulation*** (PCM).

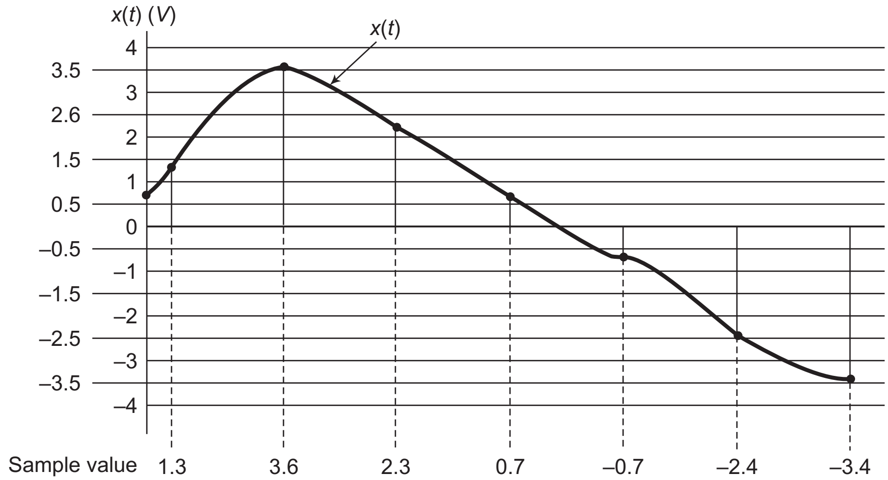


Figure 1.2.14 An Illustration of Sampling, Quantization and PCM Encoding

Explanation

For given 8 quantization levels, the step size will be 8 V/8 = 1 V. That means the quantization levels will be –3.5 V, –2.5 V, –1.5 V, –0.5 V, +0.5 V, +1.5 V, +2.5 V, and +3.5 V. To represent 8 distinct quantization levels, we require 3 bits. So, the corresponding code number and binary representation will be 0 (000), 1 (001), 2 (010), 3 (011), 4 (100), 5 (101), 6 (110), and 7 (111). Table 1.2.1 shows the determination of code number and binary code for the given quantized level of seven samples.

Table 1.2.1 Determination of Binary Code

Sample #	Quantized Level	Nearest Quantized Value	Code Number	Binary Code
1	1.3V	+1.5V	5	101
2	3.6V	+3.5V	7	111
3	2.3V	+2.5V	6	110
4	0.7V	+0.5V	4	100
5	–0.7V	–0.5 V	3	011
6	–2.4V	–2.5V	1	001
7	–3.4V	–3.5V	0	000

1.2.5 PCM System using Codec

A Fact about PCM

PCM is a digital pulse modulation technique, also known as time-domain waveform coding technique. It is the only digitally encoded modulation technique used for baseband digital transmission. It is not, in real sense, a type of modulation but rather a form of digitally encoding of analog signals.

Codec

So far we have discussed the process of converting an analog signal into a PCM signal, known as *waveform coding technique*, and the process of converting the PCM signal back from digital to analog, i.e., *decoding*. An integrated circuit that performs the PCM encoding as well as decoding functions is called a *codec* (coder + decoder). Hence, a codec is expected to perform the functions of sampling, quantization, analog-to-digital (A/D) converter as encoder on the PCM transmitter side, and digital-to-analog (D/A) converter as decoder on the PCM receiver side. One codec is used for each PCM channel. Figure 1.2.15 shows a functional block diagram of PCM system using codec.

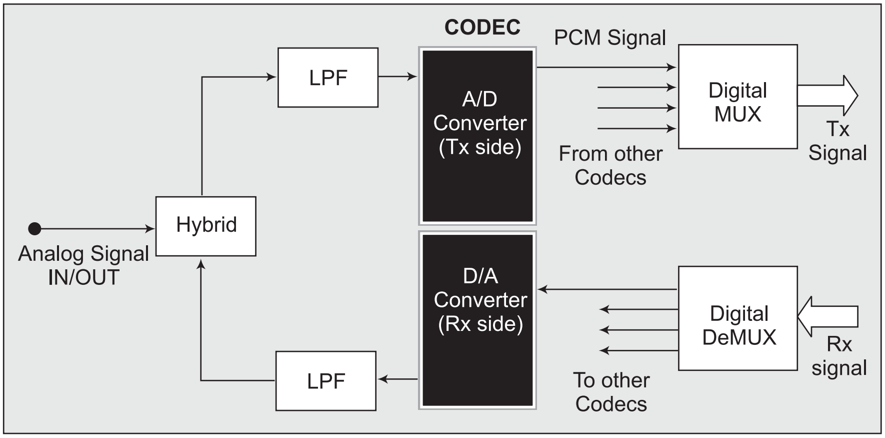


Figure 1.2.15 Functional Block Diagram of a PCM System using Codec

A Brief Description

- An analog signal is applied to transmitter side of a *codec* through a low-pass filter (used for band-limiting the analog signal).
- The codec-Tx will sample, quantize and encode this band-limited analog signal to produce a PCM signal at its output.
- It is then applied to one of many input ports of a digital multiplexer (MUX) alongwith similar PCM signals from other codecs (if available).
- The MUX combines all these PCM signals and produces a serial-bit sequence for transmission.
- The received signal is routed through digital demultiplexer (DEMUX), D/A converter and low-pass filter at the receiver side of codec in order to reconstruct the analog signal.
- The hybrid is a device which couples the analog information signal to a low-pass filter (LPF) on the Tx side and LPF on the Rx side to the source and destination, respectively. It avoids any signal coupling between transmitter and receiver.

Combo Chip

A *combo chip* can provide the codec functions, as described above, as well as the transmit and receive filtering to interface a full-duplex voice telephone circuit to the PCM channel. As stated earlier, the codec performs the functions of analog signal sampling, encoding/decoding (A/D and D/A) including digital companding. The transmit/receive low-pass filters perform functions of bandlimiting the analog information signal, noise rejection, anti-aliasing, and reconstruction of analog signals after decoding.

1.2.6 PCM System Parameters

PCM Transmission Data Rate

For a binary PCM, we assign a distinct group of n bits to each of the L quantization levels such that $n = log_2 L$. Thus, each quantized sample is encoded into n bits. As stated previously, an analog signal bandlimited to f_m Hz requires a minimum of $2f_m$ samples per second (i.e., sampling frequency, $f_s = 2f_m$ as per Nyquist rate). So, we require a total of $2nf_m$ bits per second, called *PCM transmission data rate*, or *PCM transmission speed.* In other words,

$$\text{PCM transmission data rate, } f_b = \left(\frac{\text{Samples}}{\text{Second}}\right) \times \left(\frac{\text{Bits}}{\text{Sample}}\right)$$

PCM transmission data rate, $f_b = f_s \times n$ bits/second

Substituting $f_s = 2f_m$, we get

PCM transmission data rate, $f_b = 2\ nf_m$ bits/second

PCM Bandwidth

We know that a unit bandwidth (1 Hz) can transmit a maximum of two bits per second. We can generalize that a minimum channel bandwidth B_{PCM} (Hz) $= n \times f_m$ Hz. This is the theoretical minimum *transmission bandwidth* required to transmit the PCM signal. However, for practical reasons, we may use a transmission bandwidth higher than this.

Alternately, we may state that the transmission bandwidth in a PCM system, B_{PCM}, should be greater than or equal to half of the transmission data rate, f_b. That is,

$$B_{\text{PCM}} \geq \frac{1}{2} \times f_b$$

$$\Rightarrow \qquad B_{\text{PCM}} \geq \frac{1}{2} \times f_s \times n$$

Since $f_s \geq 2f_m$ (as per Nyquist criterion),

$$B_{\text{PCM}} \geq n \times f_m$$

where f_m is the highest frequency component present in analog signal.

This expression clearly shows that increasing the number of bits per sample would result into an increase in the transmission bandwidth in PCM system. The actual transmission bandwidth will be slightly higher than calculated above. In practical PCM communication systems, additional bits would be needed to detect and correct errors as well as to ensure synchronization between transmitter and receiver, which would further increase the effective transmission bandwidth.

Dynamic Range

The *dynamic range* of a PCM system is defined as the ratio of the strongest possible signal amplitude level to the weakest possible signal level (other than 0 V) which can be decoded by the digital-to-analog converter in the receiver.

$$\text{Dynamic range,} \qquad DR(\text{dB}) = 20 \log\left(\frac{V_{\max}}{V_{\min}}\right)$$

where $V_{\max}$ is the maximum value that can be detected by the digital-to-analog converter in the receiver, $V_{\min}$ is the minimum value (also called the *resolution*).

The number of bits used for a PCM code depends on the dynamic range which are related as $DR \leq (2^n - 1)$; where n is the number of bits in a PCM code (excluding the sign bit, if any) and DR is dynamic range expressed in ratio. Note that one positive and one negative PCM code is used for 0 V, which is not considered for determining the dynamic range. It can be easily seen that the minimum number of bits required to achieve the specified value of dynamic range is given by $n = \dfrac{\log_{10}(DR+1)}{\log_{10} 2}$.

Expressing dynamic range in dB, we have

$$\boxed{DR(\text{dB}) = 20 \log (2^n - 1)}$$

For values of $n > 4$, dynamic range can be approximated as

$$DR(\text{dB}) \approx 20 \log (2^n) \approx 6\, n$$

Coding Efficiency

The *coding efficiency* of a PCM system is defined as the ratio of the minimum number of bits required to obtain the desired dynamic range to the actual number of bits used in a PCM system.

By definition, the coding efficiency (η_{PCM}) of a PCM system is given by

$$\boxed{\eta_{\text{PCM}}(\%) = \frac{\text{min\_bits}}{\text{actual\_bits}} \times 100}$$

Application

A PCM coding technique using 8 bits per sample (256 quantization levels) at a sample frequency of 8 kHz adopted for commercial telephone application at 64 kbps data rate for an acceptable signal-to-noise ratio of a telephone toll-grade quality voice is an example of uniform quantization.

Dear student... *Now we illustrate various aspects of PCM with the following examples.*

SOLVED EXAMPLE 1.2.1 **Number of Quantized Levels in PCM**

The number of bits per sample used in standard telephony voice transmission and compact disc audio storage systems is 8 and 16 respectively. Calculate the number of quantized levels in PCM.

Solution We know that the number of quantized levels, $L = 2^n$; where n is the number of bits per sample.

For telephony voice transmission system, $n = 8$ (Given)

Therefore, $L = 2^8 = 256$ levels **Ans.**

For compact disc audio storage systems, $n = 16$ (Given)

Therefore, $L = 2^{16} = 65{,}536$ levels **Ans.**

SOLVED EXAMPLE 1.2.2 **Mean-Square Quantization Error**

The quantized signal and the original signal from which it was derived, differ from each other in a random manner. As such, the process of quantization involves rounding off the sample values of an analog signal to the nearest permissible level of the quantizer. Derive an expression for the mean-square quantization error or quantization noise and show that $N_q = \Delta^2/12$, where Δ is the step-size.

Solution The quantization error is directly proportional to the difference between consecutive quantization levels. However, the quantization error is inversely proportional to the number of quantization levels for complete amplitude range. With a higher number of quantization levels, a lower quantization error is obtained. However, the performance of a quantizer is measured as the output *signal-to-quantization noise ratio*.

Now, we calculate the mean-square quantization error, $\overline{q_e^2}$, where $q_e = s(t) - s_q(t)$. Let $p(s)$ be the probability density function that $s(t)$ lies in the voltage range $\left(s - \frac{ds}{2}\right)$ to $\left(s + \frac{ds}{2}\right)$.

Then the mean-square quantization error, $\overline{q_e^2}$ is given as

$$\overline{q_e^2} = \int_{s_1-\frac{\Delta}{2}}^{s_1+\frac{\Delta}{2}} p(s)(s-s_1)^2 ds + \int_{s_2-\frac{\Delta}{2}}^{s_2+\frac{\Delta}{2}} p(s)(s-s_2)^2 ds + \cdots + \int_{s_L-\frac{\Delta}{2}}^{s_L+\frac{\Delta}{2}} p(s)(s-s_L)^2 ds$$

Usually, the number of quantization levels L is large, and the step size Δ is very small as compared to peak-to-peak range $L\Delta$ of the information signal. In this case, the probability density function $p(s)$ is constant within each quantization range. Then,

$$\overline{q_e^2} = p(s)^{(1)} \int_{s_1-\frac{\Delta}{2}}^{s_1+\frac{\Delta}{2}} (s-s_1)^2 ds + p(s)^{(2)} \int_{s_2-\frac{\Delta}{2}}^{s_2+\frac{\Delta}{2}} (s-s_2)^2 ds + \cdots + p(s)^{(L)} \int_{s_L-\frac{\Delta}{2}}^{s_L+\frac{\Delta}{2}} (s-s_L)^2 ds$$

Let $x = s - s_k$, where $k = 1, 2, \ldots L$; so that $dx = ds$, and range of integration becomes $-\frac{\Delta}{2}$ to $+\frac{\Delta}{2}$. Therefore,

$$\Rightarrow \quad \overline{q_e^2} = p(s)^{(1)} \int_{-\frac{\Delta}{2}}^{+\frac{\Delta}{2}} x^2 dx + p(s)^{(2)} \int_{-\frac{\Delta}{2}}^{+\frac{\Delta}{2}} x^2 dx + \cdots + p(s)^{(L)} \int_{-\frac{\Delta}{2}}^{+\frac{\Delta}{2}} x^2 dx$$

$$\Rightarrow \quad \overline{q_e^2} = \left[p(s)^{(1)} + p(s)^{(2)} + \cdots + p(s)^{(L)}\right] \int_{-\frac{\Delta}{2}}^{+\frac{\Delta}{2}} x^2 dx$$

$$\Rightarrow \quad \overline{q_e^2} = \left[p(s)^{(1)} + p(s)^{(2)} + \cdots + p(s)^{(L)}\right] \left|\frac{x^3}{3}\right|_{-\frac{\Delta}{2}}^{+\frac{\Delta}{2}}$$

$$\Rightarrow \quad \overline{q_e^2} = \left[p(s)^{(1)} + p(s)^{(2)} + \cdots + p(s)^{(L)}\right] \left|\frac{(\Delta/2)^3}{3} - \frac{(-\Delta/2)^3}{3}\right|$$

$$\Rightarrow \quad \overline{q_e^2} = \left[p(s)^{(1)} + p(s)^{(2)} + \cdots + p(s)^{(L)}\right] \left|\frac{2\Delta^3}{3\times 8}\right|$$

$$\Rightarrow \quad \overline{q_e^2} = \left[p(s)^{(1)} + p(s)^{(2)} + \cdots + p(s)^{(L)}\right] \frac{\Delta^3}{12}$$

$$\Rightarrow \quad \overline{q_e^2} = \left[p(s)^{(1)}\Delta + p(s)^{(2)}\Delta + \cdots + p(s)^{(L)}\Delta\right] \frac{\Delta^2}{12}$$

Since the sum of probability of all L quantization ranges is equal to unity, i.e.,

$$\left[p(s)^{(1)}\Delta + p(s)^{(2)}\Delta + \cdots + p(s)^{(L)}\Delta\right] = 1$$

Then, we get

$$\boxed{\overline{q_e^2} = \frac{\Delta^2}{12}}$$

The mean-square quantization error $\overline{q_e^2}$ is also known as quantization noise N_q in PCM and is related to the step size Δ as $N_q = \frac{\Delta^2}{12}$.

SOLVED EXAMPLE 1.2.3 **Quantization Error**

The measurement of five consecutive samples shows 1.20 V, 1.0 V, 0.95 V, 1.41 V, 1.65 V readings. The number of quantization levels is 8 in the peak-to-peak range of the analog signal of 2 V. Determine the quantization error in terms of its mean square value.

Solution For given 8 quantization levels and dynamic range of 2 V, the step size = 2 V/8 = 0.25 V. That means the quantization levels will be 0.25 V, 0.5 V, 0.75 V, 1.0 V, 1.25 V, 1.50 V, 1.75 V, and 2.0 V.

For calculating the quantization error in discrete form, each sample is considered separately, that is, the difference between measured amplitude and the quantized amplitude of the sample will give the quantization error. Table 1.2.2 shows the calculation of quantization error of each sample.

Table 1.2.2 Calculation of Quantization Error

Sample #	Measured Value	Nearest Quantized Value	Quantization Error
1	1.20 V	1.25 V	–0.05 V
2	1.0 V	1.0 V	0 V
3	0.95 V	1.0 V	+0.05 V
4	1.41 V	1.50 V	+0.09 V
5	1.65 V	1.75 V	+0.1 V

For calculating the mean square error, quantization errors of all samples must be squared, added and then averaged. That is,

$$\text{Mean square value of quantization error} = \frac{\left[(-0.05)^2 + (0)^2 + (0.05)^2 + (0.09)^2 + (0.1)^2\right]}{5}$$

$$\Rightarrow \quad \overline{q_e^2} = \frac{[0.0025 + 0 + 0.0025 + 0.0081 + 0.01]}{5} = \frac{0.0231}{5} = 0.00462 \qquad \textbf{Ans.}$$

SOLVED EXAMPLE 1.2.4 **µ-law Companding**

Consider an input analog information signal to a µ-law compressor has a positive voltage and amplitude 25% of its maximum value. Compute the output voltage as a percentage of the maximum output voltage.

Solution We know that for a µ-law compressor,

$$V_{\text{out}} = V_{\max} \frac{\ln\left(1 + \mu \frac{V_{\text{in}}}{V_{\max}}\right)}{\ln(1+\mu)}$$

For given value of $\mu = 255$, and $V_{in} = 0.25\ V_{max}$, we get

$$V_{out} = V_{max} \frac{\ln(1+255\times 0.25)}{\ln(1+255)} = 0.752\ V_{max}$$

Thus, it is observed that the output voltage is approximately **75%** of the maximum output voltage level which would be produced at the output of μ-law compressor with μ = 255.

SOLVED EXAMPLE 1.2.5 **Number of PCM Bits Per Sample**

Discrete samples of an analog signal are uniformly quantized to PCM. If the maximum value of the analog sample is to be represented within 0.1% accuracy, find the minimum number of binary digits required per sample.

Solution For a given accuracy of 0.1%, $\Delta/2 = (0.1/100) \times V$ … where Δ is the uniform step size, and V is the maximum value of the analog signal.

or, $\Delta = 0.002\ V$

We know that $\Delta = 2V/L$….. where L is the number of quantization levels, and the discrete samples are quantized from $-V$ to $+V$.

Therefore, $0.002V = 2V/L$

or, $L = 1000$

We know that $L = 2^n$ … where n is the minimum number of binary digits per sample.

Therefore, $1000 = 2^n$ or, $n = 10$

Hence, the minimum number of binary digits required per sample = 10 bits **Ans.**

SOLVED EXAMPLE 1.2.6 **Number of PCM Bits Per Sample**

Consider an analog input signal to PCM whose bandwidth is limited to 4 kHz and varies in amplitude from –3.8 V to +3.8 V, with an average signal power of 30 mW. The required signal-to-quantization error ratio is given to be 20 dB. Assuming uniform quantization, determine the number of bits required per sample.

Solution The average signal power, $S_i = 30$ mW (Given)

Signal-to-quantization error ratio, $S_i/N_q = 20$ dB or 100

Or, $(30\text{ mW})/N_q = 100;\ \rightarrow N_q = 0.3$ mW

We know that in case of uniform quantization, $N_q = \Delta^2/12$

Therefore, $\Delta^2/12 = 0.3$ mW; $\rightarrow \Delta = 0.06$

But $\Delta = (3.8 + 3.8)/2^n$

Hence, the number of bits required per sample, $n = 7$ bits **Ans.**

SOLVED EXAMPLE 1.2.7 **PCM Transmission Data Rate**

Determine the PCM transmission data rate for a single-channel PCM system employing a sample rate of 8000 samples per second and an 8-bit compressed PCM code.

Solution We know that PCM transmission data rate = $f_s \times n$

For given value of $f_s = 8000$ samples/second and $n = 8$

PCM transmission data rate = 8000 × 8 = 64 kbps **Ans.**

Dear student... *This is an example for 4 kHz voice digitization with a standard word size as 8 bits. Thus, the maximum transmission bit rate for telephony system using PCM is specified as 64 kbps.*

SOLVED EXAMPLE 1.2.8 **PCM Coding Efficiency**

A PCM system is used for an analog signal with maximum frequency of 4 kHz. If the minimum dynamic range of the quantizer used is 46 dB, and the maximum decoded voltage at the receiver is ±2.55 V, determine the minimum sampling rate, the minimum number of bits used in the PCM code, the step size, the maximum quantization error, and the coding efficiency.

Solution

(a) We know that the minimum sampling rate, $f_s = 2 f_m$ (Nyquist rate)
For given $f_m = 4$ kHz, $f_s = 2 \times 4$ kHz $= 8$ kHz **Ans.**

(b) The minimum number of bits (n) used in the PCM code depends on the dynamic range (DR). We know that for values of $n > 4$, DR (dB) $\approx 6n$
For given DR = 46 dB, we get $n \approx 46/6 \approx 7.66$.
Since the number of bits has to be a whole number, therefore, $n = 8$ bits.
But for given maximum decoded voltage of ±2.55 V, one additional sign bit is required.
Hence, the minimum number of bits used in the PCM code = 9 bits **Ans.**

(c) Step size, Δ = the maximum decoded voltage/$(2^n - 1)$
Hence, $\Delta = 2.55\text{V}/(2^8 - 1) = 0.01\text{V}$ **Ans.**

(d) Maximum quantization error $= \Delta/2 = 0.01$ V$/2 = 0.005$ V **Ans.**

(e) We know that coding efficiency is the ratio of the minimum number of bits required to obtain the desired dynamic range to the actual number of PCM bits used. Mathematically, the coding efficiency (η_{PCM}) of a PCM system is

$$\eta_{\text{PCM}}(\%) = \frac{\text{min\_bits}}{\text{actual\_bits}} \times 100$$

$$\eta_{\text{PCM}}(\%) = [(7.66 + 1 \text{ sign bit})/9 \text{ bits}] \times 100 \approx 95.9\ \%$$ **Ans.**

SOLVED EXAMPLE 1.2.9 **PCM Signal-to-Quantization Noise Ratio**

Derive an expression for signal-to-quantization noise ratio in an n-bit binary PCM system, where n is the number of bits uniquely assigned to each of the L quantization levels ($L = 2^n$).

Solution Let an analog signal has peak-to-peak range of amplitude values as $+V_m$ and $-V_m$ (i.e., $2\,V_m$). Then $\Delta = 2V_m/L$.

We know that the mean-square quantization noise, $N_q = \Delta^2/12$, where Δ is the quantization level interval or step size. By substituting $\Delta = \dfrac{2V_m}{L}$, we get

$$N_q = \frac{(2V_m/L)^2}{12} = \frac{4{V_m}^2}{12L^2} = \frac{{V_m}^2}{3L^2}$$

Let S_i be the normalized average signal power of the input analog signal. For a sinusoidal analog signal having peak amplitude V_m, the normalized average signal power (assuming that the quantization has not changed average power of the analog signal) is given as

$$S_i = \left(\frac{V_m}{\sqrt{2}}\right)^2 = \frac{{V_m}^2}{2}$$

Then signal-to-quantization noise ratio for an n-bit binary PCM signal is given by

$$\frac{S_i}{N_q} = \frac{{V_m}^2}{2} \bigg/ \frac{{V_m}^2}{3L^2} = \frac{3L^2}{2}$$

Substituting $L = 2^n$, we have $\left.\frac{S_i}{N_q}\right|_{\text{PCM}} = \frac{3(2^n)^2}{2} = \frac{3}{2}2^{2n}$

Thus, we find that for an n-bit binary PCM with a sinusoidal analog signal, the signal-to-quantization noise power ratio is given by

$$\left.\frac{S_i}{N_q}\right|_{\text{PCM}} = \frac{3}{2}2^{2n}$$

Expressing it in dB, we get

$$\left.\frac{S_i}{N_q}\right|_{\text{PCM}} (\text{dB}) = 10\log_{10}\left(\frac{3}{2}2^{2n}\right) = 10\log_{10}\left(\frac{3}{2}\right) + (10\times 2n)\log_{10} 2$$

$$\Rightarrow \qquad \left.\frac{S_i}{N_q}\right|_{\text{PCM}} (\text{dB}) = 1.76 + 6n$$

This expression shows that if one bit is added in the code word transmitted by a binary PCM system, the output signal-to-quantization noise power ratio increases by 6 dB for a sinusoidal analog signal.

SOLVED EXAMPLE 1.2.10 **Signal-to-Quantization Noise Ratio in PCM**

Compute the signal-to-quantization noise power ratio (S_i/N_q) for 4-bit binary PCM with a sinusoidal analog signal. Show that S_i/N_q increases successively by 6 dB for n = 4, 5, 6, 7 and 8.

Solution We know that for a sinusoidal analog signal input, the output signal-to-quantization noise power ratio in and n-bit binary PCM system is given by

$$\left.\frac{S_i}{N_q}\right|_{\text{PCM}} (\text{dB}) = 1.76 + 6n$$

For given n = 4, we get $\left.\frac{S_i}{N_q}\right|_{\text{PCM}} (\text{dB}) = 1.76 + 6\times 4 = 25.76$ **Ans.**

Similarly, for n = 5, we get $\left.\frac{S_i}{N_q}\right|_{\text{PCM}} = 1.76 + 6\times 5 = 31.76$ **Ans.**

Similarly, for n = 6, we get $\left.\frac{S_i}{N_q}\right|_{\text{PCM}} = 1.76 + 6\times 6 = 37.76$ **Ans.**

Similarly, for n = 7, we get $\left.\frac{S_i}{N_q}\right|_{\text{PCM}} = 1.76 + 6\times 7 = 43.76$ **Ans.**

Similarly, for n = 8, we get $\left.\frac{S_i}{N_q}\right|_{\text{PCM}} (\text{dB}) = 1.76 + 6\times 8 = 49.76$ **Ans.**

Thus, we see that for each additional bit in the code word transmitted by a binary PCM system, the output signal-to-quantization noise power ratio increases by 6 dB for a sinusoidal analog signal.

SOLVED EXAMPLE 1.2.11 **Signal-to-Quantization Noise Ratio in PCM**

An audio signal, $s(t) = 3 \cos (2 \pi 500t)$ is quantized using 10-bit PCM. Determine the signal-to-quantization noise ratio.

Solution

Average signal power, $S_i = \frac{V_m^2}{2} = \frac{3^2}{2} = 4.5$ watts (${V_m} = 3$ volts given)

Total swing of the signal $= [V_m - (-V_m)] = 2\, V_m = 2 \times 3 = 6$ volts

For 10-bit PCM, the step size, $\Delta = \frac{6}{2^{10}} = 5.86 \times 10^{-3}$

We know that quantization noise, $N_q = \frac{\Delta^2}{12}$

Therefore, $N_q = \frac{(5.86 \times 10^{-3})^2}{12} = 2.86 \times 10^{-6}$

The signal-to-quantization noise ratio, $\frac{S_i}{N_q} = \frac{4.5}{2.86 \times 10^{-6}} = 1.57 \times 10^6$

Expressing it in dB, $\frac{S_i}{N_q}(\text{dB}) = 10 \times \log(1.57 \times 10^6) = 62$ dB **Ans.**

Self-Assessment Exercise linked to LO 1.2

Q1.2.1 An analog information signal at the input to a μ-law compressor (μ = 255) is positive, with its voltage level one-half the maximum value. What proportion of the maximum output voltage level would be produced at the output of compressor?

Q1.2.2 An analog information signal at the input to A-law compressor (A = 100) is positive, with its voltage level one-half the maximum value. What proportion of the maximum output voltage level would be produced at the output of compressor?

Q1.2.3 Let the peak-to-peak range of the analog signal is specified as –4 V to +4 V and the maximum number of quantization levels is taken as 8. The magnitude of seven consecutive samples is found to be 1.3 V, 3.6 V, 2.3 V, 0.7 V, –0.7 V, –2.4 V, and –3.4 V. Determine the binary representation of each sample.

Q1.2.4 Consider a 4-bit PCM coded system. The normalized peak-to-peak input voltage range is ±16 V for a uniform quantizer. Justify that non-uniform quantization would have yielded better results.

Q1.2.5 A linear PCM system uses 12-bit quantization and encoding process for a sinusoidal analog signal. Show that the signal-to-quantization noise power ratio (S_i/N_q) is approximately 74 dB.

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/147

If you have been able to solve the above exercises then you have successfully mastered

LO 1.2: *Analyze the functional processes (sampling, quantization and binary encoding) of pulse code modulation (PCM) technique.*

Sure of what you have learnt so far? For answers, scan the QR code

OR

visit http://qrcode.flipick.com/index.php/148

Be ready for the next sections!

Mid-Chapter Check

So far you have learnt the following:

- Practical Aspects of Sampling – Methods, Nyquist Theorem, Aliasing, PAM/TDM Applications
- Pulse Code Modulation – Sampling, Quantization, Binary Encoding

Therefore, you are now skilled to complete the following tasks:

LO 1.1

MQ1.1 Define Nyquist rate and Nyquist interval.

MQ1.2 "Digital techniques are preferred over analog techniques for transmission of analog signals such as telephone voice signals". Provide key attributes to support this statement.

MQ1.3 Outline the basic principle of sampling, and distinguish between ideal sampling and practical sampling.

MQ1.4 What happens when an analog information signal is sampled at less than the Nyquist rate?

LO 1.2

MQ1.5 Find the basis for differentiating two types of quantizers — uniform quantizers and non-uniform quantizers.

MQ1.6 Hypothesize assumptions made in creating the transfer characteristics of an ideal compander. Detect the parameter due to which it cannot be realized practically.

MQ1.7 Discriminate pulse code modulation from pulse amplitude modulation as well as from quantized pulse amplitude modulation. List advantages of using binary PCM over quantized PAM signal.

MQ1.8 A linear PCM system uses 16-bit quantization and encoding process for a sinusoidal analog signal. Solve and show that the signal-to-quantization noise power ratio (S_i/N_q) is approximately 98 dB.

MQ1.9 Design a digital communication system using PCM so as to achieve a signal-to-quantization noise ratio of minimum 40 dB for an analog signal of $s(t) = 3\cos(2\pi\, 500t)$.

MQ1.10 An audio signal is required to be digitally transmitted with a sampling rate of 40 kHz and 14 bits per sample using linear PCM system. Evaluate the minimum transmission data rate needed in the communications channel.

1.3 DPCM AND ADAPTIVE DPCM

LO 1.3

Recall

We have earlier discussed that a baseband analog signal can be transmitted in the form of its sample values at each sampling time—either in the form of pulse amplitude modulation after the process of sampling, or in the form of L-ary quantized levels after the process of sampling and quantization. In these cases, the sampling is usually carried out at a rate f_s slightly higher than the Nyquist rate. Even in transmission of encoded quantized levels, i.e., PCM signal, the number of quantized levels and, hence, the number of bits n required per quantized sample are usually kept quite high in order to minimize the quantization noise. This results into higher transmission bandwidth ($f_s \times n$), which makes PCM system not very efficient.

What We Discuss Here .....

The concept of differential pulse code modulation (DPCM) that is specifically designed to take advantage of the sample-to-sample redundancies in typical analog signal waveforms.

DPCM transmitter and receiver with predictor to take advantage of sample correlation property.

Adaptive differential pulse code modulation (DPCM) in which the step size is responsive to varying level of the input analog signal.

What is DPCM?

In case of practical baseband signals such as speech waveforms, we observe that the resulting sampled signal does not usually change much from one sample value to the next one. In other words, adjacent samples carry same information with a little difference, resulting in redundant number of bits for successive sample. We can exploit this basic characteristics of the baseband signal to explore an efficient system for digital transmission of analog signal which may require less number of bits per sample.
In analog signals, we can make a good prediction and estimate about a sample value from the knowledge of past sample values. This amounts to the fact that the sample values are not completely independent, and generally there is adequate redundancy available in the Nyquist samples. Hence, the name differential pulse-code modulation (DPCM).[3]

Significance of DPCM

Instead of transmitting the sample values, we transmit only the difference between successive sample values in differential PCM.

- Let us consider the k^{th} sample having sample value as $s[k]$.
- Then we transmit the difference value $d[k] = s[k] - s[k-1]$ instead of transmitting $s[k]$ as such.
- At the receiver, $s[k]$ can be reconstructed iteratively by knowing the current difference sample value $d[k]$ and previous sample value $s[k-1]$.

Now, the range of difference values is substantially less than the range of individual sample values, the peak amplitude and, hence, the step size Δ for a given number of quantization levels L (or number of bits, n) is reduced considerably.

- This reduces the quantization noise (since $N_q = \Delta^2/12$).
- This clearly means that for a given n (or transmission bandwidth $= f_s \times n$), we can obtain better S_0/N_q, or for a given S_0/N_q, we can reduce transmission bandwidth.

---

[3] If the difference in the amplitude levels of two successive samples is transmitted rather than the absolute value of the actual sample, it is called differential PCM (DPCM). DPCM works well with data that is reasonably continuous and exhibits extremely gradual changes such as photographs with smooth tone transitions.

IMPORTANT! Hence, we conclude that fewer number of bits per quantized difference sample are required for DPCM as compared to that needed for conventional PCM. This results into reduction in overall bit rate as well as the transmission bandwidth. Thus, more efficient encoded signal is obtained.

Need of Predictor in DPCM

We know that when the sampling rate is set at the Nyquist rate, it generates unacceptably excessive quantization noise in DPCM as compared to that of in PCM. So, the quantization noise can be reduced by increasing the sampling rate considerably. In fact,

- With increased sampling rate, the difference from sample to sample becomes smaller and the rate of producing large quantization errors is reduced.
- Moreover, it results into increase in transmission bit rate (and, hence, transmission bandwidth). Generally, it exceeds that required for PCM!

On the other hand, when the signal is sampled at a rate exceeding the Nyquist rate, there is a strong *correlation* between successive samples of the baseband signal and the difference signal. Knowledge of the values of the past sample or at least the difference between values of the past successive samples enables to predict the range of the next required increment with reasonable accuracy. *So a predictor is included in DPCM system to take advantage of sample correlation property.* However, it requires provision for memory to store the difference values of past samples and to implement sophisticated algorithm to predict the next required increment.

1.3.1 DPCM Transmitter with Predictor

Figure 1.3.1 shows a simplified block diagram of a DPCM transmitter with a predictor in the feedback loop.

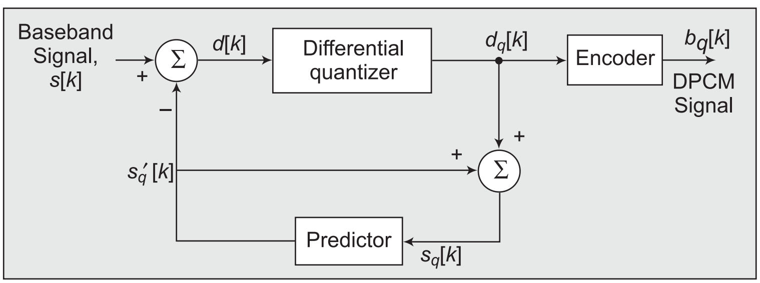


Figure 1.3.1 DPCM Transmitter with Predictor

As mentioned earlier, in DPCM we transmit the quantized version $d_q[k]$ of the difference value $d[k]$ between the present sample value $s[k]$ and the estimate of the quantized sample $s_q'[k]$, also known as the predicted quantized value of $s_q[k]$.

Derivation for Prediction Error

The difference value $d[k] = s[k] - s_q'[k]$ is quantized in the differential quantizer.

Its output is given as $d_q[k] = d[k] - q[k]$, where $q[k]$ is the quantization error and is given as $q[k] = d[k] - d_q[k]$.

The output of the predictor $s_q'[k]$ is fed back to its input after adding the output of the quantizer. That is,

$$s_q[k] = s_q'[k] + d_q[k]$$

Re-writing the expression, $d[k] = s[k] - s_q'[k]$

or, $$s_q'[k] = s[k] + d[k]$$

Substituting $s_q'[k] = s[k] + d[k]$ in the expression $s_q[k] = s_q'[k] + d_q[k]$, we have

$$s_q[k] = s[k] - d[k] + d_q[k]$$

or, $$s_q[k] = s[k] + d_q[k] - d[k]$$

or, $$s_q[k] = s[k] + q[k] \qquad \{\because q[k] = d_q[k] - d[k]\}$$

This clearly shows that $s_q[k]$ is the quantized version of original baseband signal $s[k]$. In fact, the quantized signal $s_q[k]$ is the input of the predictor, as assumed. The difference between the output and input of the predictor $s_q'[k] - s[k]$ is known as *prediction error*.

Interpretation of the Results

It is desirable that the variance of the prediction error should be as small as possible for a good predictor. In that case, a quantizer with a given number of representation levels can be adjusted to produce a quantization error with a small variance of the prediction error. The output signal is encoded and DPCM signal is obtained for transmission.

1.3.2 DPCM Receiver with Predictor

Figure 1.3.2 shows a simplified block diagram of a DPCM receiver with a predictor in the feedback loop.

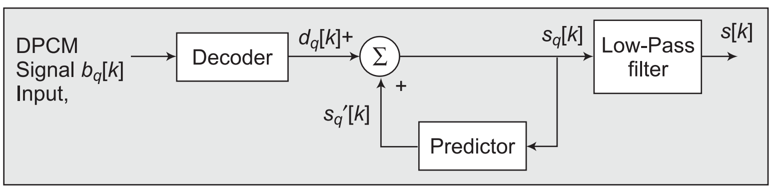


Fig. 1.3.2 DPCM Receiver with Predictor

How does a DPCM Receiver Work?

The DPCM receiver consists of a decoder to reconstruct the quantized error signal. The quantized version of the original input signal is reconstructed using exactly a similar type of predictor as used in DPCM transmitter. In the absence of channel noise, the receiver output differs from the input signal by quantizing prediction error.

Interpretation of the Results

From the above analysis, it is quite clear that by using a predictor as well as increasing sampling rate, the transmission bandwidth requirement in DPCM is much less than that of required in PCM. Moreover, the quality of signal transmission using DPCM can be made comparable to that of PCM.

Application

By using a second-order predictor processor for speech signals, the improvement in signal-to-quantization noise ratio (*S*/*N*) is typically 5.6 dB. Alternately, for the same *S*/*N*, the bit rate for DPCM could be lower than that for PCM by 3–4 bits per sample. Thus, telephone systems using DPCM can often operate at 32 kbps or even 24 kbps, instead of 64 kbps in PCM.

1.3.3 Adaptive Differential PCM

What is ADPCM?

The term *adaptive* means being responsive to changing level of the input analog signal. We use an adaptive quantizer at the encoder of *Adaptive DPCM (ADPCM)* in which the step size Δ is varied in accordance with the current change in the amplitude level of the analog signal, while keeping the number of quantization level L fixed. When Δ is fixed then either it is too large which results into larger quantization error, or it is too small that the quantizer is unable to cover the desired signal range. Therefore, the step size can be made adaptive so that Δ is large or small, depending on whether the prediction error for quantizing is correspondingly large or small. In practice, the combined use of adaptive quantization and adaptive prediction is used to achieve the best performance.

Significance of ADPCM

It is obvious that the quantized prediction error $d_q[k]$ can decide the prediction error and hence the appropriate step-size Δ. For example,

- If the quantized prediction error samples are very small then the prediction error is also very small and Δ needs to decrease.
- Similarly, when the quantized prediction error samples are very close to the largest positive or negative value, it indicates that the prediction error is large and Δ needs to increase.

IMPORTANT!

It is necessary that both ADPCM encoders and decoders have access to the same quantized samples. Therefore, the adaptive quantizer and ADPCM decoder can apply the same algorithm to adjust Δ exactly in the same way.

Performance Comparison of ADPCM

As compared to DPCM, ADPCM can further reduce the number of bits needed for encoding an analog signal waveform.

In practice, an 8-bit PCM codeword is encoded into a 4-bit ADPCM codeword at the same sampling frequency. This results into 2:1 reduction in transmission bandwidth without any degradation in the quality.

Thus, the use of adaptive quantization and adaptive prediction in ADPCM is useful in many practical applications because speech signals are inherently non-stationary but do not change rapidly from one sample to next sample.

This implies that the design of predictors for such input signals should likewise be time-varying, that is, adaptive.

Application

Adaptive quantization is an efficient quantization technique because usually the speech signals have a large dynamic range, as large as 40 dB or more. The ITU-T standard G.726 specifies an ADPCM codec @ 32 kbps for speech signals sampled at 8 kHz and is widely used in the DECT (Digital Enhanced Cordless Telecommunications) system and PHS (Personal Handy-Phone System).

Self-Assessment Exercise linked to LO 1.3

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/142

Q1.3.1 Differentiate the key features of differential PCM and conventional PCM.

Q1.3.2 Check the purpose of including an accumulator and a predictor in generating differential PCM.

Q1.3.3 Compare and contrast the major features of differential PCM and adaptive differential PCM.

Q1.3.4 If the DPCM transmitter predictor uses previous input samples for prediction then paraphrase the problem encountered at the DPCM receiver and give the reasons.

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 1.3: *Generate variants of PCM such as differential PCM (DPCM) and adaptive differential PCM (ADPCM).*

1.4 DELTA MODULATION

LO 1.4

Recall

In the previous sections, we have seen that in PCM, the analog signal samples (i.e., sampling) are quantized in L levels (i.e., quantization), and then binary encoded (n bits per sample, where $n = \log_2 L$). As discussed in differential PCM, fewer number of bits ($<n$) are used to represent the prediction error between the adjacent samples because there is always some correlation among successive samples. This leads to higher encoding efficiency.

What We Discuss Here .....

We begin with the illustration of the basic concept of delta modulation (DM) which is a single-bit version of differential PCM technique.

Then, we describe briefly the mathematical model of DM encoder and decoder.

The encoding process in delta modulation gives rise to slope overload and granular noise. These are elaborated.

Finally, we discuss adaptive delta modulation in which step size is automatically varied in order to minimize the effects of slope overload and granular noise.

Essence of Delta Modulation (DM)

If we sample a baseband signal such as voice or video, with a sampling rate much higher than the Nyquist rate (oversampling) then there is substantial increase in the correlation between adjacent samples of the analog signal. The difference between highly correlated adjacent samples will then have a much smaller range of quantized values (dynamic range). This means that we can further explore the possibility of adopting a simple quantizing and encoding strategy for obtaining the encoded signal with just 1-bit per sample! This is the essence of *delta modulation*.

DM—A Single Bit Version of DPCM

In its simplest form, delta modulation (DM) is a single-bit version of DPCM technique. DM quantizes the difference between the present and the previous sample value with just two-levels ($L = 2$) and encodes it by using just a single bit. The single bit, providing for two possibilities, binary logic 1 and 0, is used to increase or decrease the estimated or approximated waveform of the baseband signal.

Algorithm to Implement DM

The simple algorithm is followed to generate a delta-modulated signal waveform. The present sample value is compared with the previous approximated sample value. Then, there may be one of two possibilities only—either the value of the present sample is smaller, or larger than the value of the previous sample. Accordingly, a binary logic '0' is assigned in the former case, whereas logic '1' is assigned in the latter case. Hence, for each sample, only one binary bit is transmitted. The difference between the actual input signal and the approximation signal is quantized into two representation levels only: $+\Delta$ and $-\Delta$, corresponding to positive and negative difference respectively.

In its basic form, delta modulation provides a staircase approximation to the oversampled version of an input baseband signal, as shown in Figure 1.4.1.

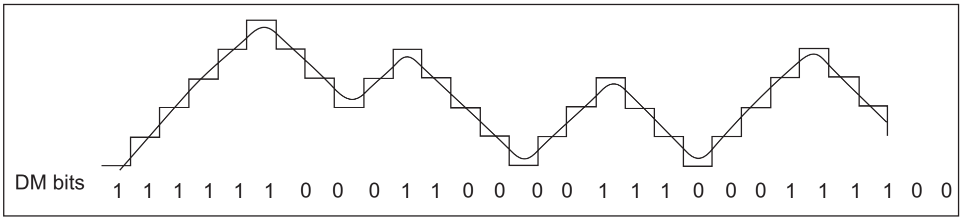


Figure 1.4.1 An Ideal Delta Modulation Waveform

How does DM differ from DPCM?

Delta modulation (DM) is, in fact, a differential PCM (DPCM) scheme in the sense that in both schemes it is the difference signal between the current sample and its previous sample which is encoded into binary digits. The difference between DM and DPCM lies in the number of bits which represents the difference signal. In DM, the difference signal is encoded into just a single bit which provides just two possibilities – either to increase or decrease the estimated signal (i.e., quantized approximation signal having just two levels only). Whereas in DPCM, fewer levels will be required to quantize the magnitude of the difference signal and correspondingly, fewer number of bits (but certainly more than one bit) will be needed to encode the quantized levels. Accordingly, a comparator is used in DM in place of difference amplifier and quantizer in DPCM because we need to know only whether current sample is larger or smaller than the quantized approximation value of the previous sample and not the magnitude of the difference signal. The up-down counter used in DM serves as the accumulator in DPCM.

How does DM Functions?

We can say that in PCM, the information of each quantized sample is transmitted by an *n*-bit codeword, whereas in DM the information of the difference between successive samples is transmitted by a 1-bit codeword. As mentioned earlier, the difference could be positive or negative. Accordingly, a positive or a negative pulse is generated in the delta modulated signal. Basically, DM carries the information about the derivative of baseband signal, hence, the name *delta modulation*.

Why to Transmit Every Sample?

In delta modulation (DM), we compare the present sample of the analog signal with an approximation of the previous sample, and their difference is applied to a single-bit quantizer, or the hard-limiter (comparator). So, if the comparator output is positive, whatever may be its actual magnitude, it gives an output of $+\Delta$ (step-size) and if the comparator output is negative, irrespective of its actual magnitude, it gives an output of $-\Delta$. This output goes to the encoder (generating binary 1 for $+\Delta$ and binary 0 for $-\Delta$) and through it, to the channel for transmission. Also, this signal serves as the approximation to the next sample by giving it as input to the accumulator. Initially, the approximate signal starts building up from zero to catch-up with the analog signal, thereby producing a *staircase approximation* to the information signal by trying to track the analog signal.

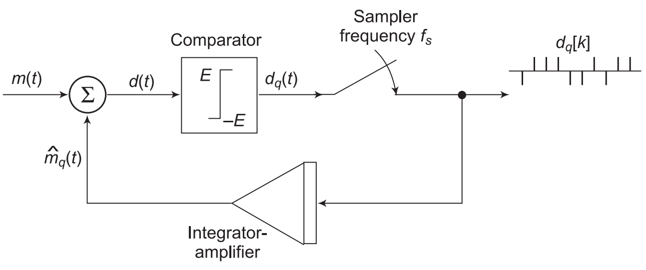


Figure 1.4.2 Basic Concept of Linear DM Encoding

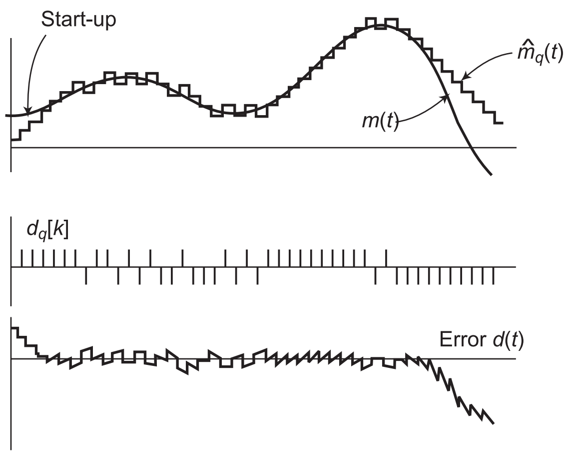


Figure 1.4.3 DM Encoder Waveforms

Decoding DM Waveform

When DM encoded waveform propagates through the channel, it will be corrupted by additive noise and so it will be distorted. An amplifier-integrator circuit followed by a low-pass filter will produce the estimate of the original analog signal. The amplifier-integrator serves the purpose of regenerating the received DM pulses as well as the decision-making and pulse generator part of DM receiver to decide, during each time slot, whether a positive or negative pulse is received. Accordingly, the output will be $+\Delta$ or $-\Delta$ volts at each sampling instant. This sequence is then fed to an accumulator (part of amplifier-integrator circuit) in the receiver which gives the staircase approximation signal. The low pass filter having the cutoff frequency as the bandlimited frequency of the original analog signal removes the high frequency out-of-band noise components and gives at its output a waveform that approximated the original analog signal waveform. Figure 1.4.4 shows the basic concept of DM decoder functions.

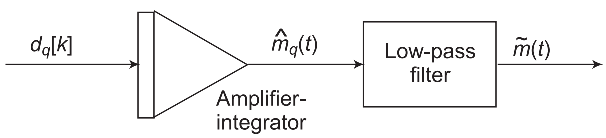


Figure 1.4.4 Basic Concept of DM Decoding

Application

As compared to PCM and DPCM, DM is a very simple and inexpensive method of digitization of analog signal (or A/D conversion). It implies that a 1-bit codeword in DM does not require any additional framing bit for synchronization of transmitter and receiver.

1.4.1 Slope Overload and Granular Noise

What is Slope Overload Noise?

As clearly observed from the encoding process in delta modulation, if the variations in adjacent samples of the analog signal are quite fast as compared to the step size (also known as *threshold of coding*), overloading occurs. In other words, if the derivative of the analog signal waveform is too high such that the approximate quantized signal waveform cannot follow the original signal waveform then the resulting slope overload gives rise to the *slope overload noise*. This is the basic limiting factor in the performance of DM.

When does it Occur?

Therefore, in delta modulation, we must choose the step size Δ and the sampling frequency f_s carefully. We know that the baseband signal has a specified maximum frequency and the fastest rate at which it can change. To account for fastest possible change in the signal, the step size and/or sampling frequency must be increased. Assuming that the input signal does not change too rapidly from sample to sample, the staircase approximation remains within ±Δ/2 of the input signal. Increasing the sampling frequency results in the delta modulation waveform that requires a large bandwidth. Hence, when the slope of the analog signal is greater than the delta modulator can maintain, it is called *slope overload distortion*.

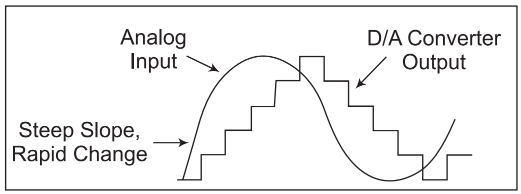


Figure 1.4.5 Slope Overload Distortion

Interpretation Thus, we can say that slope-overload distortion occurs when the input analog signal changes at a faster rate than the D/A converter in the delta encoder can maintain.

- It is more prevalent in analog signals that have steep slopes or whose amplitude levels vary rapidly.
- In fact, the slope overload occurs when the delta-modulated quantized signal cannot follow the slope of the analog signal.
- During the sample interval T_s, the delta-modulated quantized signal can change at the most by step size Δ.
- Hence, the maximum slope that the quantized signal can follow is (Δ/T_s), or $(\Delta \times f_s)$, where f_s $(= 1/T_s)$ is the sampling frequency.

Therefore, in DM, the step size is related to the sampling frequency.

In order to avoid slope-overload distortion, the maximum slope of the analog signal must be equal to or less than the maximum slope of the staircase approximation.

Mathematical Analysis We can analyze this situation for the case of a given sinusoidal analog signal $s(t) = V_m \cos(2\pi f_m t)$, where V_m is its peak amplitude, and f_m is its frequency.

Let this analog signal waveform is sampled with sampling frequency $f_s (= 1/T_s$, where T_s is the sampling interval) which is related to frequency of the analog signal by $f_s >> 2f_m$ (i.e., oversampling, as desired in case of DM).

The condition for no slope overload distortion states that slope of the input analog signal should be less than or equal to the maximum slope which the quantized sample signal can follow. Mathematically,

$$\left|\frac{d}{dt}s(t)\right|_{\max} \le \frac{\Delta}{T_s}; \quad \Rightarrow \left|\frac{d}{dt}V_m\cos(2\pi f_m t)\right|_{\max} \le \frac{\Delta}{T_s}$$

$$V_m(2\pi f_m) \le \frac{\Delta}{T_s}; \quad \Rightarrow \; V_m(2\pi f_m) \le \Delta f_s \quad \left(\because \; \frac{1}{T_s} = f_s\right)$$

$$\Rightarrow \qquad f_s \geq \frac{2\pi f_m V_m}{\Delta}$$

$$\Rightarrow \qquad V_m\Big|_{\max} = \frac{\Delta \times f_s}{2\pi f_m}$$

Thus, we see that the overload amplitude of the sinusoidal analog signal is inversely proportional to its frequency. This also means that for higher sinusoidal signal frequency, the slope-overload distortion occurs for smaller amplitude. Thus, we conclude that in order to reduce slope-overload distortion, an increase in the magnitude of the minimum step size is required. But a large step-size is required to accommodate wide dynamic range of the input signal. Moreover, the sample frequency should also be increased. So, we must choose the step size Δ and the sampling frequency f_s carefully for optimum performance of DM system.[4]

Possible Solution

In fact, there are two major sources of noise in delta modulation systems, namely, slope-overload distortion and granular noise. If we increase the step size to overcome the slope-overload distortion then quantization noise, known as *granular noise* in delta modulation, increases. Granular noise is more prevalent in analog signals that have gradual slopes and whose amplitude levels vary only by a small amount.

What is Granular Noise?

When the input analog signal has a relatively constant amplitude, the reconstructed signal has variations that were not present in the original signal. This is known as *granular noise*. It is similar to quantization noise in conventional PCM.

When does it Occur?

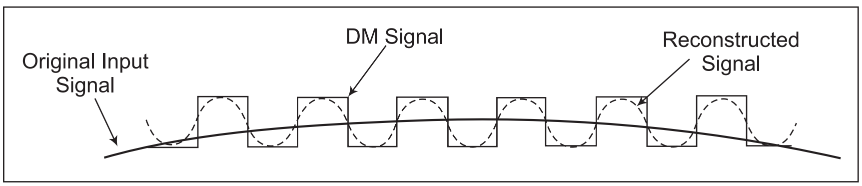


Figure 1.4.6 Granular Noise

Interpretation It is seen that to reduce granular noise, a decrease in the magnitude of the minimum step size and a small resolution is required. These are contrary to the requirement of increasing the step size to reduce the slope overload distortion. Thus, there is an optimum value of Δ, which yields the best compromise giving the minimum overall noise (slope overload distortion and granular noise). As indicated earlier, the optimum value of Δ depends on the sampling frequency and the slowly varying or fast-changing nature of the baseband analog signal.[5]

[4] The maximum amplitude of the voice signal that can be used without causing slope overload distortion is the same as the maximum amplitude of a sinusoidal signal of reference frequency 800 Hz that can be used without causing slope overload distortion in the same system. In practice, a first order predictor along with a single integrator up to 2 kHz and a double integrator beyond 2 kHz is used.

[5] For voice signals, DM system operating at 4 kbps data rate is equivalent to a standard PCM system operating with a sample rate of 8 kHz and 5 bits/sample that means at 40 kbps data rate—a clear cut advantage of using DM over PCM. DM is well suited for voice and television signals because the voice or video spectrum decays with frequency and closely follows overload characteristics.

1.4.2 DM Encoder and Decoder

DM Encoder

As stated previously, delta modulation is a special case of DPCM. In DM, we use a first-order predictor which is just a time delay of the sampling interval T_s. Thus, the DM encoder (modulator) of DM transmitter as well as decoder (demodulator) of DM receiver are identical to those of the DPCM except that there is a time delay for the predictor in accumulator. Figure 1.4.7 shows a typical model of DM system encoder (modulator). The discrete-time relationship between input signal, error signal and their quantized versions are also indicated.

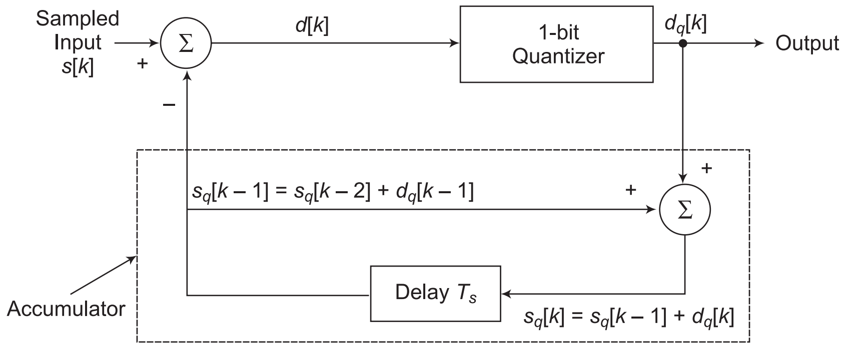


Figure 1.4.7 Mathematical Model of a DM Encoder

Mathematical Analysis From the figure, we can write

$$s_q[k] = s_q[k-1] + d_q[k] \quad \text{and} \quad s_q[k-1] = s_q[k-2] + d_q[k-1]$$

Therefore,
$$s_q[k] = s_q[k-2] + d_q[k-1] + d_q[k]$$

This means we can proceed iteratively in the same way for each sample. Assuming the initial condition, $s_q[0] = 0$, we obtain the generalized form as

$$s_q[k] = \sum_{s=0}^{k} d_q[s]$$

DM Decoder

This shows that the feedback part of the DM encoder is just an accumulator which may be simply realized by an integrator. Figure 1.4.8 shows the corresponding model of DM system decoder (demodulator).

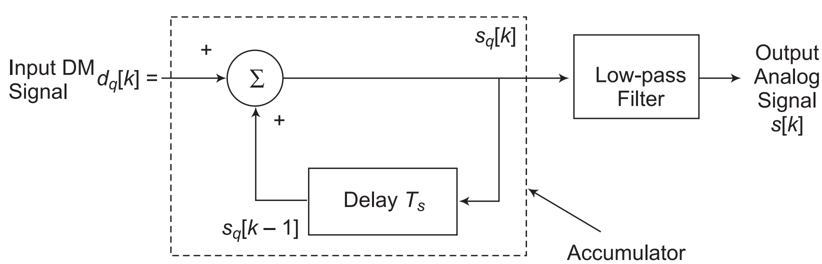


Figure 1.4.8 Mathematical Model of DM Decoder

Functional Description As shown in the figure, the decoder (demodulator) of DM receiver is just an accumulator. The incoming delta-modulated data sequence of positive and negative pulses is passed through an accumulator in a similar manner as used in DM encoder. The output is passed through a low-pass filter which yields the desired signal reconstructed from the quantized samples.

In delta modulation, a single transmission bit error may result in an offset error of twice the step size in all later values. This is quite serious. In PCM, a single transmission bit error causes an error in reconstructing the associated sample value only.

Signal-to-Quantization Noise Ratio in DM

The amount of quantization noise in delta modulation process is given by

$$\delta(t) = m(t) - \hat{m}(t)$$

where $m(t)$ represents the analog signal waveform, and $\hat{m}(t)$ represents the delta-modulated waveform.

As long as the slope overloading is avoided, the error signal $\delta(t)$ is always less than the step size Δ. Assuming that $\delta(t)$ takes on all values between $-\Delta$ to $+\Delta$ with equal probability, then the probability density of $\delta(t)$ is given as

$$p(\delta) = \frac{1}{2\Delta}; -\Delta \le \delta \le +\Delta$$

The normalized power of the error waveform, $\delta(t)$ is given by

$$\overline{|\delta(t)|^2} = \int_{-\Delta}^{+\Delta} \delta^2 p(\delta) d\delta = \int_{-\Delta}^{+\Delta} \delta^2 \frac{1}{2\Delta} d\delta = \frac{1}{2\Delta}\int_{-\Delta}^{+\Delta} \delta^2 d\delta$$

$$\Rightarrow \quad \overline{|\delta(t)|^2} = \frac{1}{2\Delta}\left[\frac{\Delta^3}{3} - \frac{(-\Delta)^3}{3}\right] = \frac{1}{2\Delta}\left[\frac{\Delta^3}{3} + \frac{\Delta^3}{3}\right]$$

$$\Rightarrow \quad \overline{|\delta(t)|^2} = \frac{1}{2\Delta} \times \frac{2\Delta^3}{3} = \frac{\Delta^2}{3}$$

The spectrum of $\delta(t)$ extends continuously from normally zero up to transmission bit rate f_b, where $f_b = \frac{1}{\tau}$; τ being the duration of the step.

The *output noise power* is given by

$$\boxed{N_q\Big|_{DM} = \frac{\Delta^2}{3} \times \frac{f_M}{f_b} = \frac{\Delta^2 f_M}{3 f_b}}$$

where f_M is the upper limit of the analog signal frequency range.

The two-sided power spectral density (PSD) of $\delta(t)$ is then given as

$$S_\Delta(f) = \frac{\Delta^2/3}{2 f_b} = \frac{\Delta^2}{6 f_b}; -f_b \le f \le f_b$$

The maximum *output signal power* which may be transmitted is given as

$$S_0 = \frac{\Delta^2 f_b^2}{2(2\pi f_M)^2} = \frac{\Delta^2 f_b^2}{8\pi^2 f_M^2}$$

The output signal-to-quantization ratio for DM is computed as

$$\left.\frac{S_o}{N_q}\right|_{DM} = \frac{\Delta^2 f_b^{\,2}}{8\pi^2 f_M^{\,2}} \div \frac{\Delta^2 f_M}{3 f_b} = \frac{\Delta^2 f_b^{\,2}}{8\pi^2 f_M^{\,2}} \times \frac{3 f_b}{\Delta^2 f_M}$$

$$\left.\frac{S_o}{N_q}\right|_{DM} \approx \frac{3}{80}\left(\frac{f_b}{f_M}\right)^3$$

In order to avoid overload distortion, f_b has to be increased by same factor as f_M.[6]

SOLVED EXAMPLE 1.4.1 — To Avoid Slope Overload

A sinusoidal analog signal is applied to a DM encoder having a fixed step-size of 0.5V. If the sampling frequency is 20 times the Nyquist rate, then determine the maximum allowed amplitude of the input analog signal so as to avoid slope overload distortion.

Solution We know that the maximum allowed amplitude of the input analog signal,

$$\left.V_m\right|_{\max} = \frac{\Delta \times f_s}{2\pi f_m}$$

For analog frequency of f_m Hz, the Nyquist rate $f_{s(\min)} = 2f_m$

It is specified that sampling frequency, $f_s = 20\, f_{s(\min)} = 20 \times 2f_m = 40\, f_m$

For given $\Delta = 0.5$V, we have $\left.V_m\right|_{\max} = \dfrac{0.5\ V \times 40 f_m}{2\pi f_m} = 3.18$ V **Ans.**

SOLVED EXAMPLE 1.4.2 — Minimum Sampling Frequency in DM

A linear delta modulator is fed with an analog signal $m(t) = 4 \sin (2\pi\ 15t)$. Determine the minimum sampling frequency required to prevent slope overload distortion, assuming that the step size of the modulator is 0.1π.

Solution We know that the minimum sampling frequency of the input analog signal required to prevent slope overload distortion in DM encoder,

$$f_s = \frac{2\pi f_m V_m}{\Delta}$$

The given analog signal $m(t) = 4 \sin (2\pi\ 15t)$ contains the single frequency component as

$$2\pi f_m t = 2\pi\ 15t \quad \Rightarrow f_m = 15 \text{ Hz}$$

For specified $V_m = 4$V and $\Delta = 0.1\pi$, we have

$$f_s = \frac{2\pi 15 \times 4V}{0.1\pi} = 1200 \text{ Hz}$$ **Ans.**

Dear student... *For analog frequency of f_m = 15 Hz, the Nyquist rate $f_{s(min)} = 2f_m$ = 30 Hz. Thus we see that in this case the minimum sampling frequency of the input analog signal required to prevent slope overload distortion in DM encoder is 40 times the Nyquist rate.*

[6] In PCM, the value of each quantized sample is transmitted by an n-bit code word, whereas in DM the value of the difference between successive samples is transmitted by a 1-bit code word.

SOLVED EXAMPLE 1.4.3 **Maximum Allowable Amplitude in DM**

A delta modulator system is designed to operate at five times the Nyquist rate for an analog information signal band-limited to 3 kHz. The quantizing step-size is 40 mV. Determine the maximum allowable amplitude of input analog signal for which the delta modulator does not experience slope-overload distortion.

Solution Let the analog information signal is represented as $v_m(t) = V_m \sin(2\pi f_m t)$. We have to determine its maximum allowable amplitude $V_m(\max)$ for which the delta modulator does not experience slope-overload distortion.

We know that the slope of the analog information signal $V_m(t)$ will be maximum when its derivative $\frac{d}{dt}\left[v_m(t)\right]$ will be maximum, that is,

Slope of the analog signal, $\frac{d}{dt}\left[V_m \sin(2\pi f_m t)\right] = V_m 2\pi f_m \cos(2\pi f_m t)$ is maximum.

Taking maximum value of $\cos(2\pi f_m t) = 1$, $\left.\frac{d}{dt} v_m(t)\right|_{\max} = V_m 2\pi f_m$

Maximum slope of delta modulator is given as the ratio of its step size and sampling period, that is, $\frac{\Delta}{T_s}$, where sampling period $T_s = \frac{1}{f_s}$ (f_s is the sampling frequency).

We know that slope overload distortion will take place if slope of analog information signal is greater than maximum slope of delta modulator. To avoid slope overload distortion,

$$\left.\frac{d}{dt} v_m(t)\right|_{\max} = \frac{\Delta}{T_s}, \text{ or } V_m(\max) 2\pi f_m = \Delta \times f_s$$

Or, $$V_m(\max) = \frac{\Delta \times f_s}{2\pi f_m}$$

For given values of $\Delta = 40$ mV, $f_m = 3$ kHz, and $f_s = 5 \times 2 \times f_m$, we get

$$V_m(\max) = \frac{40 \text{ mV} \times 5 \times 2 \times f_m}{2\pi f_m} = 63.7 \text{ mV}$$ **Ans.**

SOLVED EXAMPLE 1.4.4 **Signal-to-Quantization Noise Ratio in DM**

An audio signal comprising of a single sinusoidal term, $s(t) = 3 \cos(2\pi 500t)$ is quantized using DM. Determine the signal-to-quantization noise ratio.

Solution

The average signal power, $S_o = \frac{V_m^{\,2}}{2} = \frac{3^2}{2} = 4.5$ watts ($V_m = 3$ volts given)

Since $f_m = 500$ Hz (Given)

Therefore, the Nyquist rate, $f_s = 2f_m = 1000$ samples/second

Let the sampling frequency is 8 times the Nyquist rate, that is, $f_b = 8000$ samples/second, because delta modulation sampling takes place at a rate well above Nyquist rate.

Let the maximum amount that the function can change in 1/8000 seconds is approximately 1 V. Generally, a step size is chosen which is equal to or greater than the maximum slope of the analog signal. This is required to avoid slope overload distortion. Let us choose a step size of 1 V.

The quantization noise power is then given by

$$N_q\Big|_{DM} = \frac{\Delta^2 f_m}{3 f_b} = \frac{1^2 \times 500}{3 \times 8000} = 20.83 \times 10^{-3} \text{ W}$$

Hence, $$\frac{S_o}{N_q}\bigg|_{DM} = \frac{4.5}{20.83 \times 10^{-3}} = 216 \Rightarrow 23.3 \text{ dB}$$ **Ans.**

SOLVED EXAMPLE 1.4.5 S_o/N_q for PCM and DM system

Compute the Signal-to-quantization noise ratio (S_o/N_q) for PCM and DM system for given 8 number of bits (*n=8*) in a code-word. Comment on the result.

Solution

We know that for PCM, $\frac{S_o}{N_q} = 2^{2\times n}$

For given value of n=8, $\frac{S_o}{N_q}(PCM) = 2^{2\times 8}$

Therefore, $\frac{S_o}{N_q}(PCM) = 10\log(2^{16}) \simeq 48 \text{ dB}$ **Ans.**

We also know that for DM, $\frac{S_o}{N_q} = \frac{3}{80}(2\times n)^3$

For given value of n=8, $\frac{S_o}{N_q}(DM) = \frac{3}{80}(2\times 8)^3 = 153.6$

Therefore, $\frac{S_o}{N_q}(DM) = 10\log(153.6) \simeq 22 \text{ dB}$ **Ans.**

Comment on the result: For a fixed channel bandwidth (that is for same number of bits, say 8), the performance of DM in terms of Signal-to-quantization noise ratio (S_o/N_q) is 22 dB only as compared to that for PCM which is as high as 48 dB.

SOLVED EXAMPLE 1.4.6 S_o/N_q Comparison

Compute the Signal-to-quantization noise ratio (S_o/N_q) for PCM and DM system for the speech signal which has a bandwidth of 3200 Hz. Assume *n=2* in a code-word. Comment on the result.

Solution In case of speech signal, the given bandwidth of 3200 Hz is adequate, and the voice frequency spectrum has a pronounced peak at 800 Hz (that is one-fourth of specified bandwidth).

We know that for PCM, $\frac{S_o}{N_q} = 2^{2\times n}$

For given value of n=2, $\frac{S_o}{N_q}(PCM) = 2^{2\times 2}$

Therefore, $\frac{S_o}{N_q}(PCM) = 10\log(2^4) \simeq 12\,\text{dB}$ **Ans.**

We also know that for DM, $\frac{S_o}{N_q} \approx 0.6(2\times n)^3$

For given value of n=2, $\frac{S_o}{N_q}(DM) = 0.6\times(2\times 2)^3$

Therefore, $\frac{S_o}{N_q}(DM) = 10\log(38.4) \simeq 15.8\,\text{dB}$ **Ans.**

Comment on the result: For a speech signal, the performance of DM in terms of Signal-to-quantization noise ratio (S_o/N_q) is better than that offered by PCM by about 3.8 dB.

1.4.3 Delta-Sigma Modulation

Drawbacks of DM

As per our discussion on slope-overload distortion and granular noise in case of delta modulation, we emphasized that the conventional DM quantizes and encodes the derivative of the analog signal. This implies that the regenerative repeaters or the receiver of DM requires an integrator. We know that the transmitted signal inevitably is subjected to random channel noise. The channel noise will also be integrated and accumulated at the output of the DM receiver. This degrades the overall performance of the communication which is a major drawback of DM systems.

Delta-Sigma Modulation—A Solution

In order to overcome this drawback of DM, we combine the operation of transmitter integrator (in the accumulator part) and the receiver integrator, and then shift it prior to the encoder in the transmitter. It can be easily analyzed that this modification does not affect the overall modulator-demodulator response of DM communication link. This modified system is known as the *delta-sigma modulation* ($\Delta - \Sigma$).

What is Delta-Sigma Modulation?

In *delta-sigma modulation* ($\Delta - \Sigma$), the input to the delta modulator is actually the difference between the integral of the analog signal and the integrated output pulses. One of the advantage of shifting the receiver integrator is that the channel noise is not accumulated and demodulated at the receiver, thereby improving the performance of the system. Moreover, the low-frequency components of the analog signal are pre-emphasized (boosted) by the integrator which is helpful in speech signals whose low-frequency components are more important. In this way, the correlation between adjacent samples of delta modulator is also increased, which in turn reduces the variation of the error signal at the input of quantizer. The integrator effectively smoothens the analog signal for encoding, thereby reducing overloading.

Application

High oversampling is employed in sigma-delta modulation systems, they are mostly useful in low-frequency applications such as digital telephony, digital audio encoders (compact disc) and digital spectrum analyzers.

1.4.4 Adaptive Delta Modulation (ADM)

Why ADM?

As discussed in conventional delta modulation, the dynamic range of amplitude level of the difference between adjacent sample values is very small. Moreover, the step size cannot be made large due to the slope overload effect. This problem can be overcome by *adaptive delta modulation* in which the step size is automatically varied, depending on the level of the derivative of the input analog signal. For example,

- If the input signal is varying very fast, the step size is increased suitably during this period with which the slope overload could also be avoided.
- Similarly, if the input signal is varying slowly, the step size is reduced which, in turn, will reduce granular noise.

This amounts to discrete changes in the value of step size in accordance with changes in the input signal levels.

Complexity of ADM

Of course, the receiver must be able to adapt step sizes in exactly the same manner as the transmitter. Since the transmission consists of a series of binary digits, the step size must be derived from this bit sequence.

For example, the slope overload causes the quantized difference signal to have several successive pulses of the same polarity. The increased step size will avoid this situation to happen. Similarly, small-level variations will cause the successive pulses to alternate continuously in polarity. This requires a reduction in step size to minimize this effect.

In ADM, we detect such type of pulse patterns and automatically adjust the required step size. This results in a much larger dynamic range and thereby improved performance.

A Common Algorithm

A common algorithm followed for an ADM is that when three consecutive 1s or 0s occur, the step size is increased or decreased by a factor of 1.5. However, various algorithms may be used for ADM, depending on the requirement of the particular system. After the next sample, if the output level is still below the current sample level, the next step size increases even further until the output is closer to that of current sample value of analog signal. Slope-overload distortion reduces but granular noise (quantization error) increases. Therefore, when an alternative sequence of 1s and 0s occur, ADM automatically reverts to its minimum step size.

An Alternative to ADM—CVSDM

The performance of ADM can be further improved by adjusting the step size Δ continuously according to the input signal characteristics.

The continuous variable slope delta modulation (CVSDM) is a delta modulation technique which employ continuously varying step size. CVSDM may be viewed as a special case of adaptive delta modulation, which also encodes at the rate of 1 bit per sample so that the baseband signal is sampled at 8 kHz which results into transmission encoded data rate of 8 kbps.

Application

CVSDM is quite robust under high bit-error rate conditions. Usually bit rates achieved are 9.6 kbps to 128 kbps. 12 kbps CVSDM is used for digitally encrypted two-way radio systems. 16 kbps CVSDM is used for military digital telephones to provide voice recognition quality audio systems. 64 kbps CVSDM is one of the options used to encode voice signals in Bluetooth devices.

1.4.5 Comparison of PCM and DM Techniques

Now, we are in a position to compare various parameters associated with digitization of an analog signal with respect to traditional PCM, Differential PCM, DM, and Adaptive DM techniques as summarized in Table 1.4.1.

Table 1.4.1 Comparison of PCM and DM Techniques

S. No.	Parameter	PCM	DPCM	DM	ADM
1.	Number of bits per sample	4/8/16 bits	More than one bit but less than PCM	One bit	One bit
2.	Number of levels, L	Depends on number of bits, $L = 2^n$	Fixed number of levels	Two levels, $L = 2$	Two levels, $L = 2$
3.	Step size	Fixed or variable	Fixed or variable	Fixed	Variable
4.	Transmission bandwidth	More bandwidth needed	Lesser than that of PCM	Lowest	Lowest
5.	Feedback	Does not exist	Exists	Exists	Exists
6.	Quantization noise/distortion	Quantization noise depends on step-size	Quantization noise and slope overload distortion	Slope overload distortion and granular noise	Quantization noise only
7.	Complexity of implementation	Complex	Simple	Simple	Simple

Self-Assessment Exercise linked to LO 1.4

Q1.4.1 Delta modulation provides a staircase approximation to the oversampled version of an input baseband signal. Interpret the meaning of over sampling of a baseband signal.

Q1.4.2 Delta modulation needs a simple circuit as compared to pulse code modulation. Outline the comparative benefits and drawbacks of delta modulation.

Q1.4.3 Hypothesize the possible situations in which the use of delta modulation scheme is recommended.

Q1.4.4 A sinusoidal modulating signal $m(t) = V_m\ (2\pi f_m t)$ is applied as input to a linear delta modulator. Find the expression for the signal power to quantization noise power under the condition that no slope overload distortion occurs. Assume that the quantization error has uniform distribution with zero mean.

Q1.4.5 A linear delta modulator is fed with an analog signal having a single tone frequency of 1 kHz and maximum amplitude level of 1V. Determine the minimum sampling frequency required to prevent slope overload distortion, assuming that the step size of the modulator is 0.1π. Find the corresponding number of samples to be transmitted if 2 seconds duration sampled signal is to be transmitted without slope overload distortion.

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/145

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 1.4: *Implement delta modulation and adaptive delta modulation techniques.*

Key Concepts

- adaptive delta modulation
- adaptive differential PCM
- aliasing
- codec
- companding
- delta modulation
- decoding
- differential PCM
- encoding
- foldover distortion
- granular noise
- pulse amplitude modulation
- pulse code modulation (PCM)
- sampling
- quantizing
- quantization error
- quantization noise
- sample-and-hold circuit
- slope overload

Learning Outcomes

LO 1.1

- Modern digital communication systems have better performance and use less bandwidth than equivalent analog communication systems.
- For digital transmission of an analog information signal, it must be sampled at least twice per cycle of its highest frequency component; else it will result into undesirable aliasing distortion.
- Pulse amplitude modulation is mainly used for time-division multiplexing (TDM) of several analog signals, and for generating the most popular pulse code modulation.

LO 1.2

- PCM system essentially requires that the amplitude of each sample of an analog signal be converted to a binary code.
- PCM essentially involves sampling (PAM), quantizing, and binary encoding.
- More number of bits used per sample in PCM, greater is the accuracy but the higher bit rate (and transmission bandwidth) will be required.
- The signal-to-noise ratio (SNR) can often be improved by using companding for PCM.

LO 1.3

- In differential PCM (DPCM), only the difference between the amplitude of the previous and present samples is encoded.
- In DPCM, fewer levels and, hence, fewer bits will be required to quantize and encode the difference between successive samples.
- If the sampling frequency in DPCM is same as that used in PCM, the quantization noise will be increased because only the differences in samples are encoded.

LO 1.4

- Delta modulation system transmits only one bit per sample corresponding to whether the signal level is increasing or decreasing from its previous sample value.
- DM needs a higher sampling rate than PCM for desirable results.

Hands-on Projects

1.1 Design and implement PCM modulator using IC CW6694 from CONWISE alongwith circuits for associated functional blocks such as 2048 kHz square wave generator, IC μA 741C Op-amp for voltage controlled voltage source low-pass filter.

1.2 Search through the web to enlist various types of codec and combo ICs used in generation and reception of PCM signals. Prepare a comparative chart of their key features and applications. Design a PCM link selecting any one of these chips. Implement it on PCB, apply a composite analog signal and evaluate the received signal.

1.3 Maxim's DS2164Q ADPCM processor chip is a dedicated digital-signal-processing (DSP) chip that has been optimized to perform adaptive differential pulse-code modulation (ADPCM) speech compression at three different rates. The chip can be programmed to compress (expand) 64kbps voice data down to (up from) either 32kbps, 24kbps, or 16kbps. Study the data sheet and evolve algorithm to design ADPCM technique.

1.4 Realize a practical circuit to implement delta modulator and demodulator using functions such as comparator, digital-to-analog converter, up/down counter, digital-to-analog converter along with other components. Compare its performance with that of conventional PCM link.

Objective-Type Questions

1.1 Identify which process of sampling in pulse code modulation is known as
- (a) analog modulation
- (b) impulse modulation
- (c) digital modulation
- (d) quadrature amplitude modulation

1.2 Choose the sampling rate at which the amplitude of the analog speech signal is sampled in standard digital voice communication systems.
- (a) 1600 samples/second
- (b) 3200 samples/second
- (c) 6400 samples/second
- (d) 8000 samples/second

1.3 Select the appropriate Nyquist sampling rate when an analog signal having highest frequency of 1630 Hz and lowest frequency of 250 Hz is sampled.
- (a) 1630 Hz
- (b) 1880 Hz
- (c) 2760 Hz
- (d) 3260 Hz

1.4 A continuous time signal is represented by $s(t) = 8 \cos(200\,\pi t)$. The minimum sampling rate to avoid aliasing should be
- (a) 100 Hz
- (b) 200 Hz
- (c) 400 Hz
- (d) 800 Hz

For Interactive Quiz with answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/165

1.5 *Statement I:* In uniform quantizers, step size remains same throughout the amplitude range of the input analog signal.
Statement II: The process of companding allows binary codes to be compressed, thereby reducing transmission time and bandwidth requirements.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

1.6 In ____________________, the code used for each sample is based on the difference between successive samples rather than samples themselves.
(a) differential pulse code modulation
(b) delta modulation
(c) delta-sigma modulation
(d) adaptive delta modulation

1.7 To avoid slope-overload distortion in delta modulation, which one of the following relationships between the sampling rate f_s, the maximum frequency f_m, the maximum amplitude of the input sinusoidal signal V_m, and the step size Δ is correct?

(a) $f_s \geq \dfrac{2\pi f_m \Delta}{V_m}$

(b) $f_s \geq \dfrac{2\pi f_m V_m}{\Delta}$

(c) $f_s \geq \dfrac{2\pi V_m}{\Delta f_m}$

(d) $f_s \geq \dfrac{2\pi f_m}{\Delta V_m}$

1.8 *Statement I:* The S_o/N_q performance of DM is better than PCM by more than 3 dB.
Statement II: For same quality of transmission, the bit rate needed for DM is much higher than by PCM.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

1.9 Slope overload is a large difference between the original and replicated delta modulation signals. It can be avoided by using
(a) uniform step size
(b) non-uniform step size
(c) adaptive step size
(d) non-adaptive step size

1.10 The use of PCM at the standard transmission bit rate of 64 kbps
(a) requires 4 kHz channel bandwidth for transmission of digitized voice signals

(b) requires 8 kHz channel bandwidth for transmission of digitized voice signals
(c) requires 32 kHz channel bandwidth for transmission of digitized voice signals
(d) requires 64 kHz channel bandwidth for transmission of digitized voice signals

Short-Answer-Type Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/150

1.1 What is the significance of Nyquist criterion for the process of sampling an analog signal? Give an account of performance of natural and flat-top sampling with suitable illustrations.
1.2 Define Nyquist rate. What happens when an analog signal is sampled at less than the Nyquist rate?
1.3 How do you justify that pulse code modulation is a modulation as well as encoding technique?
1.4 State three criteria for proper signal encoding so as to determine how successful a PCM receiver would be to decode the signal.
1.5 Voice-quality telephone signals require a relatively constant signal-to-quantization noise ratio performance over a wide dynamic range. Can it be achieved with compander characteristics?
1.6 Bring out clearly the similarities and dissimilarities between μ-law and A-law companding.
1.7 Discuss critically the benefit of using companding. Give at least one technical reason for use of different companding schemes in the US and Europe.
1.8 List the essential operations performed at the PCM receiver.
1.9 What is meant by quantization range and quantization error? How does it depend upon the step size?
1.10 Outline the basic principle of differential PCM technique.
1.11 How is adaptive quantization useful in adaptive differential PCM?
1.12 Specify the algorithm that is followed to generate delta-modulated signal waveform.
1.13 Delta modulation is a special case of differential PCM. Carry out the comparative study of similarities among these two techniques.
1.14 What happens when the variations in adjacent samples of the analog signal are quite fast as compared to the step-size in delta modulation? Draw an appropriate illustration to depict this phenomenon.
1.15 Specify two unique features offered by delta modulation.

Discussion Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/146

1.1 The analog signal such as speech or video signals is sampled with a pre-determined train of narrow rectangular pulses in such a way so as to closely approximate the instantaneous sampling process. State and discuss the criteria for faithful representation of analog signal with discrete samples. **[LO 1.1]**

1.2 The sampling rate must be fast enough so that at least two samples are taken during the period corresponding to the highest frequency spectral component present in the analog signal. Critically analyze the impact of oversampling and under sampling. **[LO 1.1]**

1.3 PCM is used in digital telephone systems (trunk lines) and is also the standard form for digital audio in computers and various compact disc formats, digital videos, etc. How can an analog voice signal be digitized and transmitted? **[LO 1.2]**

1.4 It is highly desirable that signal-to-quantization noise ratio should remain essentially constant for a wide range of input signal levels such as speech. Suggest a practical solution so as to obtain a constant signal-to-quantization noise ratio and elaborate the implementation scheme. **[LO 1.2]**

1.5 In case of practical baseband signals such as speech waveforms, we observe that the resulting sampled signal does not usually change much from one sample value to the next one. Show that differential PCM is more bandwidth efficient than conventional PCM. **[LO 1.3]**

1.6 In order to reduce slope-overload distortion in delta modulation (DM), an increase in the magnitude of the minimum step size is required. But a large step size is required to accommodate wide dynamic range of the input signal. Moreover, the sample frequency should also be increased. In view of these aspects, establish a relationship among various parameters for optimum performance of DM system. **[LO 1.4]**

1.7 In conventional delta modulation, the dynamic range of amplitude level of the difference between adjacent sample values is very small. Moreover, the step size cannot be made large due to the slope overload effect. Plan an alternate delta modulation technique to overcome the problem of slope overload distortion. **[LO 1.4]**

Problems

For answer keys, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/153

1.1 For the analog signal $s(t) = 2\cos(100\,\pi t) + 18\cos(2000\,\pi t)$, determine the allowable sampling rate and sampling interval. ○○●

1.2 The signal $x(t) = 2\cos(400\,\pi t) + 6\cos(640\,\pi t)$ is ideally sampled at $f_s = 500$ Hz. If the sampled signal is passed through an ideal low-pass filter with cut off frequency of 400 Hz, then calculate the frequency component that will appear in the filter output. ○●●

1.3 A binary channel with bit rate of 36 kbps is available for PCM voice transmission. Evaluate the Nyquist sampling rate, recommended sampling rate, the number of quantization levels, and the number of binary digits in PCM encoder. ○○●

1.4 Audio signals having bandwidth up to 15 kHz are recorded digitally using PCM. If the audio signal is sampled at a rate 20% more than the Nyquist rate for practical considerations and the samples are quantized into 65, 536 levels, determine the minimum bandwidth required to transmit the PCM signal. ○●●

1.5 Consider a PAM signal whose amplitude varies from –0.5 V to +7.5 V. This range is divided into eight uniform quantization levels. The pulses from –0.5 V to +0.5 V are approximated (quantized) to a value 0 V, the pulses from +0.5 V to +1.5 V are quantized to 1 V, and so on. Determine the 3-bit binary code corresponding to three pulses having amplitudes 3.8 V, 1.2 V, and 5.7 V respectively. ●●●

1.6 The information in analog signal having maximum frequency of 3 kHz is required to be transmitted using 16 quantization levels in PCM system. Find the number of bits per sample and minimum sampling rate required for PCM transmission. ●●●

1.7 A 6-bit single-channel PCM system gives an output transmission data rate of 60 kbps. What could be the maximum possible analog frequency for this system? ○●●

1.8 Consider the maximum frequency of an analog information signal is 3.2 kHz. A binary channel of bit rate 36 kbps is available for PCM voice transmission. Compute the minimum sampling frequency, number of bits required per sample, and number of quantized levels. ●●●

1.9 Consider an analog signal $x(t) = 3 \cos (500\ \pi t)$.Evaluate the signal-to-noise power ratio using 10 bit PCM system. ●●●

1.10 Determine the step size for a quantized PAM signal whose peak voltages are +16.8 V and –22.4 V, assuming 16-bit PCM code for each sample. ○●●

1.11 A PCM system uses a uniform quantizer followed by n-bit encoder. Show that rms signal to quantization noise ratio is approximately given as $(1.8 + 6\ n)$ dB. ●●●

1.12 Given maximum voltage amplitude of 3.6 V, determine the compressed value for an input voltage of 1.26 V using (i) μ-law compander; (ii) A-law compander. ○●●

1.13 A delta modulator system is designed to operate at five times the Nyquist rate for an analog information signal band limited to 3 kHz. The quantizing step size is 40 mV. Determine the maximum allowable amplitude of input analog signal for which the delta modulator does not experience slope-overload distortion. ○○●

1.14 Compare the signal-to-quantization noise ratio (S_o/N_q) for PCM and DM system for given 8 number of bits ($n = 8$) in a code word. Comment on the result. ○○●

1.15 Compute the signal-to-quantization noise ratio (S_o/N_q) for PCM and DM system for the speech signal which has a bandwidth of 3200 Hz. Assume $n = 2$ in a code word. ○●●

Critical Thinking Questions

1.1 A significant advantage of digital transmission over analog transmission against the background noise is that it is not necessary to know the exact amplitude of the received pulse waveform. Consider an arbitrary analog signal waveform. Illustrate the process of natural sampling and depict its spectrum. **[LO 1.1]**

1.2 For conversion of an analog signal into its equivalent digitized form, we sample the analog signal along its time axis, followed by discretization of its amplitude-axis too. Why is it necessary to quantize the pulse amplitude modulated signal? Compare and contrast two methods of quantization—uniform and robust, and bring out clearly the benefit of robust quantization. **[LO 1.2]**

1.3 Differential PCM works well with data that is reasonably continuous and exhibits extremely gradual changes such as photographs with smooth tone transitions. With DPCM, more efficient encoded signal can be obtained. Analyze the mathematical models of DPCM encoder and decoder and highlight the utility of predictor in its feedback path. **[LO 1.3]**

1.4 Delta modulation quantizes the difference between the present and the previous sample value with just two-levels and encodes it by using just a single bit. Illustrate the simple algorithm to implement delta modulation. Also critically analyze the problem encountered when the variations in adjacent samples of the analog signal are quite fast as compared to the step size. **[LO 1.4]**

References for Further Reading

[Cou00] Couch, L., *Digital and Analog Communication Systems*. Upper Saddle River, NJ: Prentice Hall, 2000.

[SI12] Singal, TL; *Analog and Digital Communication*. Tata McGraw-Hill, 2012.

2 Baseband Transmission and Digital Multiplexing

Learning Objectives

To master baseband transmission and digital multiplexing, you must cover these milestones:

LO 2.1 Know how to convert digital data to digital signal using various line codes for reliable baseband transmission

LO 2.2 Discuss intersymbol interference (ISI) problem in digital transmission

LO 2.3 Hypothesize Nyquist pulse-shaping criteria and outline equalizer schemes to minimize the effects of ISI

LO 2.4 Illustrate the concept of digital multiplexing and design TDM systems

LO 2.5 Integrate TDM–PCM encoded digital carrier systems and calculate their data rates

Essence of Baseband Transmission and Digital Multiplexing

Baseband transmission means sending a digital signal directly over a communications channel. We know that the information such as text, computer files, and programs, have a natural digital (usually binary) representation. This information needs to be converted to a digital signal for digital transmission. Digital information or data can be represented, in binary form, by two voltage levels. Generally, binary data usable to an application are not in a form that can be transmitted over the desired communications channel. To improve propagation characteristics, however, the digital data are often encoded into a more complex form of digital signals. Line coding is the process of converting digital data to digital signals with certain properties. Digital data have a broad spectrum with a significant low-frequency content. Baseband transmission of digital data, therefore, requires the use of a low-pass channel with a bandwidth large enough to accommodate the essential frequency contents of the data stream. In this chapter, we study the transmission of digital data over a baseband channel. Digital multiplexers enable simultaneous transmission of many line-encoded digital signals over a common communications channel.

INTRODUCTION

Recall

Refer Chapter 1: PCM and Delta Modulation and Demodulation

We have discussed in the previous chapter that analog signal waveforms are transformed into a sequence of 1s and 0s using *waveform coding technique* such as pulse code modulation, delta modulation, and other digital pulse modulation techniques. These binary digits are just abstractions—a way to describe the analog information signal. The digital output of these source encoders is required to be converted into electrical pulses for the purpose of baseband transmission over the channel.

Question

How can you convert digital data, usually available in the form of 0s and 1s, to digital signals that must represent digital data in the form of appropriate voltage levels on a wireline?

Solution

Probably the simplest process is ***line coding***, also known as ***transmission coding***, or ***digital baseband signaling***. You will find the details of line codes and its types in Section 2.1.

Good to Know!

- There are *several* line codes available for electrical representation of binary data stream. These line codes differ from one another with respect of desirable *line properties* which are necessary for efficient transmission of baseband digital signals.
- Due to dispersive nature of the channel, each received pulse is affected by an adjacent pulse. This results into *Intersymbol Interference* (*ISI*) which is a major source of bit errors in the reconstructed data stream at the receiver.
- *Pulse shaping* and *equalization* techniques employed in the baseband signal receiver help to control bit errors.
- For simultaneous transmission of many baseband digital signals over a common channel, digital multiplexing techniques such as *time-division multiplexing* (TDM) is employed.

A PRIMER

Recap

The term 'baseband' is used to designate the frequency band of the original message (information) signal directly from the source or from the input transducer of an electronic communication system. For example, in telephony, the baseband is the audio band (band of voice signal) of 300 Hz–3400 Hz. In television, the video baseband is the video band 0 to 4.5 MHz. For digital data or pulse code modulation (PCM) that uses bipolar line code signaling at a rate of R pulses per second, the baseband is approximately 0 to R Hz.

IMPORTANT!

In baseband communication, information signals are directly transmitted without any change in its frequency band (i.e., that do not use carrier modulation). Bandwidth utilization means the efficient use of available bandwidth which can be achieved by the process of multiplexing.

In this chapter...

- We begin with a description of the need and properties of *line codes*, followed by discussion on various *line coding techniques* and their *power spectra*. **LO 2.1**
- Next, we describe an undesirable phenomenon associated with baseband digital data transmission, known as *intersymbol interference (ISI)*. **LO 2.2**
- The discussion is carried forward to describe different methods to minimize the effect of ISI on the performance of digital communication, by employing special techniques such as *pulse shaping* and *equalization*. **LO 2.3**
- For simultaneous transmission of several baseband digital signals, we employ *digital multiplexers* known as Time-Division Multiplexers (TDM). **LO 2.4**
- Finally, North American T1 and European E1 digital carrier systems that are capable of carrying several PCM-encoded multiplexed digital data (or digitized analog data) on common channel are discussed. **LO 2.5**

2.1 LINE CODING AND ITS TYPES

LO 2.1

We know how to convert an analog signal to a digital data (sequence of binary symbols 1's and 0's) either by using waveform encoders such as PCM, DM, etc. The digital output comprising of a long sequence of binary symbols 1's and 0's is neither uniform nor suitable for direct transmission over the communications channel. These binary symbols are required to be converted into electrical pulses or waveforms having finite voltage levels so as to make it compatible for transmission over the communications channel. It is in a *line code* that a binary stream of data takes on an electrical representation.[1]

What is Line Coding?

Line coding is the process by which digital symbols are transformed into waveforms that are compatible with the characteristics of the baseband channel. Thus, line coding involves converting (encoding) the binary data output of a waveform encoder or a source encoder into electrical pulses (waveforms) for the purpose of baseband digital signal transmission. It is obvious that the communication channel such as wireline can propagate digital (usually binary) pulse waveforms directly.

Figure 2.1.1 depicts baseband digital transmission of digitized data over communication channel such as a wireline.

[1] We know how to represent binary data, i.e., in terms of two symbols 0 and 1. But how do we assign electrical voltages to them that suits communication over wireline media? This issue is addressed in line coding. That is why, line coding is sometimes referred to as *digital carrier line encoding* or *baseband digital modulation*.

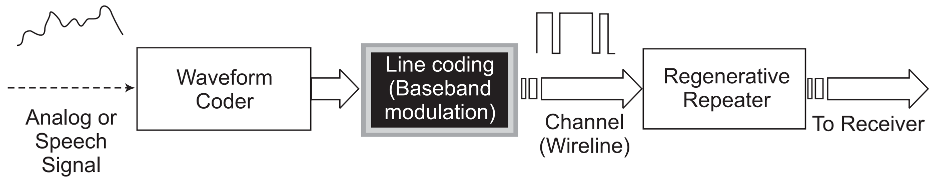


Figure 2.1.1 Baseband Digital Transmission

Thus, line coding converts a sequence of information data bits to a digital signal. We assume that data, in the form of text, numbers, graphical images, audio, or video, are stored in computer memory as sequences of bits in computer communications. At the sender, digital data are encoded into a digital signal; and at the receiver, the digital data are reconstructed by decoding the digital signal.

Figure 2.1.2 shows the process of line coding (*Encoder*) and decoding (*Decoder*).

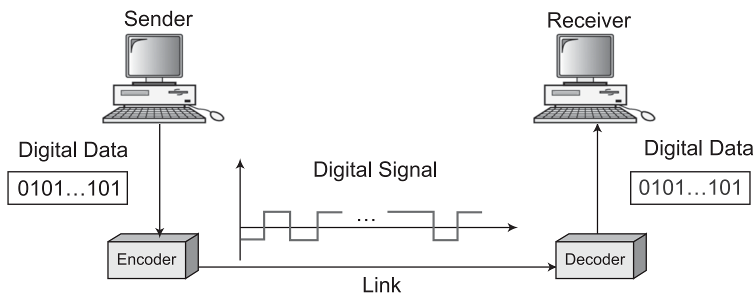


Figure 2.1.2 Process of Line Coding

Line coding, in general, looks into factors such as

(i) Power and bandwidth required for transmission
(ii) Ability to extract timing information
(iii) Presence of dc or low-frequency components which is not suitable for ac-coupled circuits
(iv) Error-detecting ability

What Next?

Before we proceed further, let us understand the basic terminology associated with transmission of line-encoded digital data, i.e., digital signal.

Data Element and Data Rate

Define

A ***data element*** is the smallest entity that can represent a piece of information, that is, the bit. It is data elements which we need to send through a digital communication system. The ***data rate***, or the ***bit rate***, is defined as the number of data elements (bits) sent in one-second time duration. The unit of data rate is ***bits per second (bps)***.[2]

Signal Element and Signal Rate

Define

A *signal element* is the shortest unit (time wise) of a digital signal which can be sent through a a digital communication system. In digital data communications, a signal elements carries data elements. The *signal rate* is defined as the number of signal elements sent in one-second time duration. The unit of signal rate is the *baud*, sometimes called the *baud rate*, the *pulse rate*, or the *modulation rate*.

[2] Data elements are being carried; signal elements are the carriers. In other words, a signal element carries data element in digital data communications. Our goal is to increase the data rate while decreasing the signal rate. Increasing the data rate increases the speed of data transmission; decreasing the signal rate decreases the transmission bandwidth requirement.

Relationship between Data Rate and Signal Rate

We now discuss the relationship between data rate (or, bit rate) and signal rate (or, baud rate). Let us define a ratio r which is given as the number of data elements carried by each signal elements. For example,

- One data element per one signal element, i.e., $r = 1$. Time duration of a signal element is exactly the same as that of a data element.
- One data element per two signal elements, i.e., $r = 1/2$. Time duration of a signal element is one-half that of a data element. In other words, there are two signal elements (two transitions) to carry each data element.
- Two data elements per one signal element, i.e., $r = 2$. This means a signal element carries two data elements.
- Four data elements per three signal elements, i.e., $r = 4/3$. This means a group of 4 bits is carried by a group of three signal elements, and so on.

The relationship between data rate and signal rate (i.e., bit rate and baud rate), of course, depends on the value of r as well as the data pattern. For example, if we have a data pattern of all 1s or all 0s, the signal rate may be different from a data pattern of alternating 0s and 1s.

From the above discussion, we can formulate the relationship between data rate and signal rate as

$$S = k \times f_b \times \frac{1}{r} \ \textbf{baud}$$

What do these Variables Represent?

S is the number of signal elements in one second, expressed in baud;

k is the factor which depends on the requirement of minimum or maximum signal rate (k varies from 0 to 1, with an average value taken as ½);

f_b is the data rate, expressed in bps;

r is ratio of the number of data elements carried by each signal elements.

Example

Let us consider that a signal is carrying data in which one data element is encoded as one signal element ($r = 1$). The average baud rate to transmit 100 kbps data rate will be $S = \left(\frac{1}{2}\right) \times (100000) \times \left(\frac{1}{1}\right) = 50000$ baud; taking an average value of k as ½.

However, in practice, most digital signals have a finite bandwidth (range of frequencies). In other words, the actual bandwidth of a digital signal is theoretically infinite, but many of its frequency components have such a small amplitude that they can be ignored.

So, we can say that the effective bandwidth is finite.

IMPORTANT!

It is, in fact, the baud rate, not the bit rate, which determines the required transmission bandwidth for a digital signal. For the moment, we can say that the bandwidth of the channel is directly proportional to the signal rate (baud rate).

Therefore, the minimum bandwidth required can be given as

$$B_{\min} = k \times f_b \times \frac{1}{r} \ \ \textbf{Hz}$$

k, f_b, and r have same meanings, as defined previously.

Recall

We know that a signal with L number of levels actually can carry $\log_2 L$ bits per level, i,e., $r = \log_2 L$.

If each level corresponds to one signal element then for an average case $k = ½$, we have

$$f_{b(\max)} = 2 \times (B) \times (\log_2 L) \quad \text{bps}$$

It may be recalled that this is, in fact, Nyquist formula which is used to determine the maximum data rate of a channel.

Example

Let us consider that a channel has a bandwidth of 200 kHz. If we use four levels of digital signaling then the maximum data rate of the channel will be

$$f_{b(\max)} = 2 \times (200 \text{ kHz}) \times (\log_2 4) = 800 \quad \text{kbps}$$

Synchronization of Digital Signals

Digital transmission systems are usually ***synchronous***, that is, they operate in conjunction with a clock waveform, which is a regular periodic waveform whose time period is equal to the **bit duration** T_b.

- The clock signal also serves the purpose to mark the beginning and end of the time duration allocated to each bit.
- In a digital transmission system, there must be a clock waveform at both the transmitting end as well as at the receiving end.
- These clocks must be synchronized with one another.
- This implies that the receiver's bit duration must correspond exactly to the sender's bit duration.
- Then, only the digital signals received by the receiver from the sender can be correctly interpreted.

Define

The ***signaling rate*** or ***transmission data rate***, f_b is defined as the rate at which the digital data is transmitted, and is related to the bit duration as $f_b = \dfrac{1}{T_b}$ bps.

2.1.1 Desirable Properties of Line Codes

The line codes often use the terminology non-return-to-zero (NRZ) or return-to-zero (RZ). As the name suggests,

- ***Non-Return-to-Zero (NRZ)*** indicates that the binary pulse used to represent the bit does not necessarily return to zero or neutral level during the bit period.
- ***Return-to-Zero (RZ)*** implies that the binary pulse used to represent the bit always returns to zero or neutral level usually at the middle of the bit period.

Before we discuss different types of line-coding techniques, we address their common characteristics and desirable properties of a good line code for a specified application. Upon reading this subsection, you will be able to answer how to optimize the performance of digital data transmission.

List of Desirable Properties of Line Codes

There are certain desirable properties which must be considered for line coding.

- Transmission power efficiency
- Duty cycle
- The dc components
- Baseline wandering
- Bandwidth considerations
- Self-clocking capability or self-synchronization
- Immunity to noise and interference
- Error detection capability
- Ease of detection and decoding

All these desirable properties of line-coding signaling formats are now discussed for better understanding of different types of line codes in the next sub-section.

Property 1 Transmission Power Efficiency

Transmission power efficiency, or transmission voltage levels, can be categorized as either unipolar (UP), or polar, as given below:

- In *unipolar voltage levels*, only one nonzero voltage level is specified. For example, positive voltage level is used to represent binary data 1 and zero (ground) voltage level is used to represent binary data 0.
- In *polar voltage levels*, two distinct nonzero symmetrical but opposite voltage levels are specified. For example, positive voltage level is used to represent binary data 1 and negative voltage level is used to represent binary data 0.

Dear student... *After reading this material, you should be able to answer how power efficiency can be increased by using polar voltage levels. This is illustrated with the help of the following example.*

SOLVED EXAMPLE 2.1.1 **Transmission Power Efficiency—Polar *versus* Unipolar**

Show that there is 200% improvement in power efficiency in case of a polar line code using symmetrical voltage levels as compared to that of a unipolar line code for the same voltage level. Assume a load resistor of 1 Ω.

Solution

Case I: Unipolar Line-Coding Waveform

Step 1: Let binary data 1 be represented by +5 V (peak) and binary data 0 is represented by 0 V. [Assume]

Step 2: Therefore, average power in case of unipolar waveform,

$$P_{\text{av}}(\text{UP}) = \frac{V_{\text{rms}}^{2}(1)}{R} + \frac{V_{\text{rms}}^{2}(0)}{R} = \frac{1}{R}\left[V_{\text{rms}}^{2}(1) + V_{\text{rms}}^{2}(0)\right]$$

Substituting the values as $R = 1\ \Omega$, $V_{\text{rms}} = V_p/\sqrt{2}$; we have

$$P_{\text{av}}(\text{UP}) = \frac{1}{1\ \Omega}\left[\left(\frac{5}{\sqrt{2}}\right)^{2} + 0\right] = 12.5\ \text{W} \qquad \text{[Compute it carefully]}$$

Case II: Polar Symmetrical Line-Coding Waveform

Step 3: Let binary data 1 be represented by +5 V (peak) and binary data 0 is represented by –5 V (peak). [Assume]

Step 4: Therefore, average power in case of polar waveform,

$$P_{\text{av}}(\text{UP}) = \frac{V_{\text{rms}}^{2}(1)}{R} + \frac{V_{\text{rms}}^{2}(0)}{R} = \frac{1}{R}\left[V_{\text{rms}}^{2}(1) + V_{\text{rms}}^{2}(0)\right]$$

Substituting the values as $R = 1\ \Omega$, $V_{\text{rms}} = V_p/\sqrt{2}$; we have

$$P_{\text{av}}(\text{Polar}) = \frac{1}{1\ \Omega}\left[\left(\frac{+5}{\sqrt{2}}\right)^{2} + \left(\frac{-5}{\sqrt{2}}\right)^{2}\right] = 25\ \text{W} \qquad \text{[Compute it carefully]}$$

Step 5: Improvement in Transmission Power Efficiency

$$\frac{P_{\text{av}}(\text{Polar})}{P_{\text{av}}(\text{UP})} \times 100 = \frac{25}{12.5} \times 100 = 200\% \qquad \textbf{[Hence, Proved]}$$

Thus, there is an improvement in transmission power efficiency by 200% in case of polar symmetrical voltage levels ($+V$ and $-V$) as compared to that of unipolar line code using the same voltage level ($+V$).

BASED ON THE ABOVE EXAMPLE,
you are now ready to attempt problems related to transmission power efficiency of unipolar, polar, and bipolar voltage levels.

Property 2: Duty Cycle

Duty cycle is defined as the ratio of the bit duration for which the binary pulse has defined transmission voltage to the entire bit duration.

For example,

Define

- In **non-return-to-zero (NRZ) line-encoding format**, the binary pulse is maintained high for binary data 1 for the entire bit duration, or the binary pulse is maintained low for binary data 0 for the entire bit duration. So, the duty cycle is 100%. We will see the details of NRZ line-encoding format in Section 2.1.2.
- In **return-to-zero (RZ) line-encoding format**, the binary pulse is maintained high for binary data 1 for 50% of the entire bit duration only, and the binary pulse is maintained low for binary data 0 for the entire bit duration. So, the average duty cycle is less than 100% of specified bit duration. We will see the details of RZ line-encoding format in Section 2.1.2.

Property 3: DC Components

Some communication systems like a telephone system (a telephone line cannot pass frequencies below 200 Hz), and a long-distance link using transformers (to isolate different parts of the line electrically) do not allow transmission of frequencies around zero, called direct-current (dc) components. This situation occurs when the voltage level in a digital signal is constant for a while, the frequency spectrum shows very low-frequency components. Therefore, we need a line-coding technique with no dc component. Figures 2.1.3 and 2.1.4 show a signal without dc component and with dc component respectively.

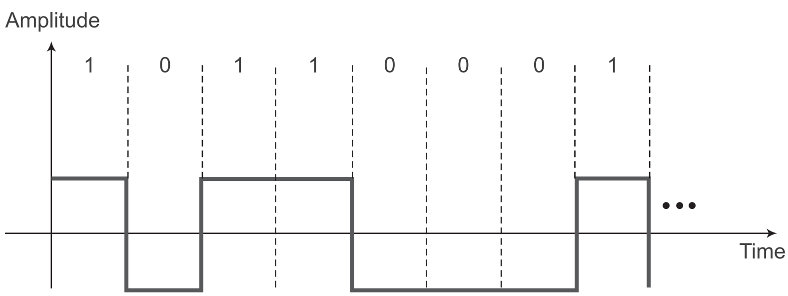


Figure 2.1.3 A Digital Signal without dc component

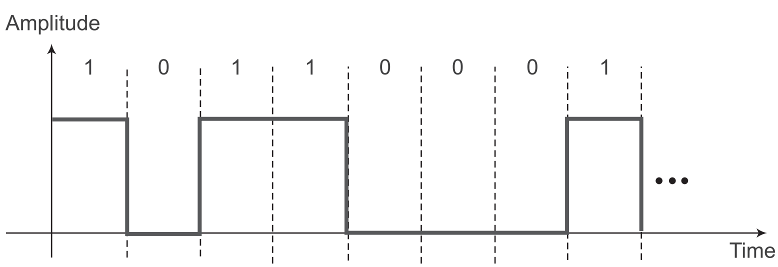


Figure 2.1.4 A Digital Signal with dc component

Property 4: Baseline Wandering

In digital transmission, a long sequence of either 1s or 0s produces a condition in which a receiver may lose its amplitude reference. This reference is needed for optimum determination of received 1s and 0s with clear-cut discrimination between them. Similar conditions may also arise when there is a significant imbalance in the number of 1s and 0s transmitted. This condition causes a drift in the baseline (a running average received signal power calculated by the receiver while decoding a digital signal), called *baseline wandering*. This makes it difficult for the receiver to decode received data correctly. Therefore, a good line-coding scheme needs to prevent baseline wandering.

Property 5: Bandwidth Considerations

Obviously, it is desirable that the bandwidth of a line code should be as small as possible. This allows more information to be transmitted per unit channel bandwidth.

Property 6: Self-clocking Capability or Self-synchronization

A self-synchronizing digital signal includes timing information in the data being transmitted. This can be achieved if there are transitions in the line-encoded transmitted digital signal that alert the receiver to the beginning, middle, or end of the pulse waveform. If the receiver's clock is out of synchronization, these transitions can reset the clock. We will discuss later that there are some line coding techniques which exhibit self-clocking capability.[3]

Figure 2.1.5 shows the effect of lack of self-synchronization in which the receiver has a shorter bit duration.

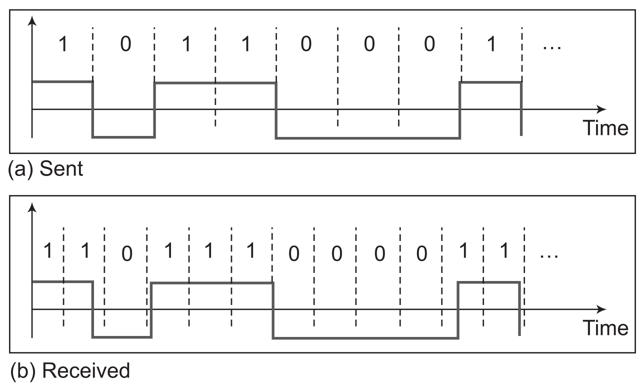


Figure 2.1.5 Need of Synchronization between Transmitted and Received Data

As it can be seen from the figure, the receiver receives extra bits (110111000011) in the same time duration as the transmitter transmits (10110001). Thus, we can say that due to mismatch of the sender and receiver clock *intervals*, the receiver might misinterpret the received digital signals.

So, it is imperative to say that the receiver's bit intervals must correspond exactly to the sender's bit intervals for correct interpretation of digital bit stream.

SOLVED EXAMPLE 2.1.2 Significance of Synchronization

A digital transmission system transmits data at the data rate of 1 Mbps.

(a) How many extra bits per second are received by the receiver if the receiver clock is 0.1% faster than the sender clock?

[3] In synchronous digital transmission systems, digital data is transmitted serially over a communication channel. If the receiver clock is faster or slower, the bit intervals are not matched and the receiver might misinterpret the signals.

(b) What happens if the data rate of sender is 1 kbps only?

Solution

(a) In one second, the sender sends 1,000,000 bits (i.e., data rate of 1 Mbps).
Now, the receiver clock is 0.1% faster than the sender clock.
This means that the receiver receives 1,000,000 × 0.1/100 = 1,000 extra bits in one second.
Hence, at 1 Mbps, the data rate at which the receiver receives is 1,001,000 bps. **Ans.**

(b) At 1 kbps data rate, the receiver receives 1 extra bit in 1 second.
Hence, at 1 kbps, the data rate at which the receiver receives is 1,001 bps. **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.3 In a digital transmission, the sender clock is 0.2 percent faster than the receiver clock. How many extra bits per second does the sender send if the data rate is 1 Mbps?

Property 7: Immunity to Noise and Interference

It is desirable that a line-encoding format should be capable to minimize the effects of noise and interference. This will enable to have minimum errors introduced in transmitted data due to external noise and interference. We will see later that there are some line coding techniques which exhibit this capability.

Property 8: Error-Detection Capability

It is desirable that line codes should have a built-in error-detection capability which should enable to detect some or all errors that occurred during transmission. There should not be any need to introduce additional bits into the transmitted data sequence for this purpose. Some line-coding techniques do have this capability.

Property 9: Ease of Detection and Decoding

In order to make receiver design simple and reliable, line-coding techniques should be such so as to have easy detection and decoding of digital data information. Moreover, a complex line-coding technique is more costly to implement than a simple one. For example, a technique that uses four signal levels is more difficult to interpret than one that uses only two levels.

LET'S RECONFIRM OUR UNDERSTANDING!!

- Define bit rate and baud rate.
- What do you understand by line coding?

2.1.2 Types of Line-Coding Techniques

Recap

As mentioned earlier, a line code converts a binary stream of data to an appropriate representation of electrical waveform, as depicted in Figure 2.1.6 in its simplest form.

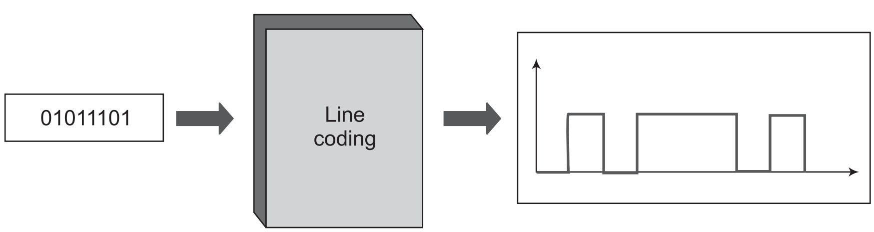


Figure 2.1.6 A Simplest Form of Line Coding

Various line-coding techniques can also be distinguished based on polarity of voltage levels used to represent the binary digits. Accordingly, line-coding techniques can be broadly classified in three types, as depicted in Figure 2.1.7.

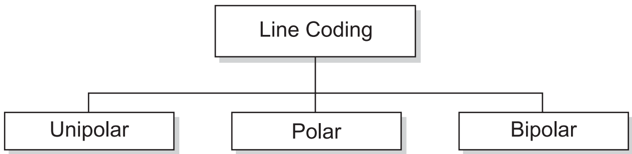


Figure 2.1.7 Classification of Line-Coding Techniques

- ***Unipolar*** (represented by only one level +V or –V). Thus, Unipolar line coding can be Unipolar Non-Return-to-Zero **(UP-NRZ)** and Unipolar Return-to-Zero **(UP-RZ)**.
- ***Polar*** (represented by two distinct non-zero symmetrical but opposite voltage levels, +V and –V).[4]
- ***Bipolar*** (also known as pseudo-ternary +V, –V, and 0 V or alternate mark inversion).

Polar line codes can be further classified as depicted in Figure 2.1.8.

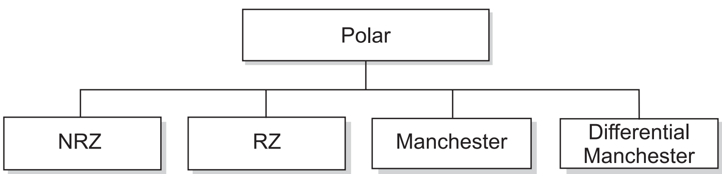


Figure 2.1.8 Classification of Polar Line Codes

List of Types of Line Codes

Any one of several line coding techniques can be used for the electrical representation of a binary data stream:

- Unipolar Non-Return-to-Zero (NRZ) Line Code
- Unipolar Return-to-Zero (RZ) Line Code
- Polar Non-Return-to-Zero (NRZ) Line Code
- Manchester Polar Line Code
- Differential Manchester Polar Line Code
- Bipolar Non-Return-to-Zero Alternate Mark Inversion (BP-NRZ-AMI) Line Code
- Bipolar RZ Line Code
- Bipolar RZ-AMI Line Code
- High-Density Bipolar (HDB) NRZ-AMI Line Code
- Binary Eight Zeros Substitution (B8ZS) RZ-AMI Line Code

Now, we describe each one of these line codes briefly in order to develop better understanding.

Type 1: Unipolar NRZ Line Code

In *unipolar NRZ line code*, a binary symbol 0 is represented by 0 V for the entire bit interval T_b and the binary symbol 1 is represented by a constant voltage level, say $+V$ or $-V$, during its entire bit interval T_b (where bit rate is $1/T_b$), and, hence, called non-return-to-zero (NRZ).

[4] The polar line coding is the most efficient one because it requires the least power for a given error probability. The bipolar line code has the advantage that one single error can be detected because the received pulse sequence will violate the bipolar rule.

Mathematically, unipolar NRZ waveform can be expressed as

For symbol '0'; $v(t) = 0$ during $0 \le t < T_b$ interval

For symbol '1'; $v(t) = +V$ or $-V$ during $0 \le t < T_b$ interval

SOLVED EXAMPLE 2.1.4 **Unipolar NRZ Line Code**

Illustrate the unipolar NRZ line coding waveform for the binary data sequence given as 0 1 0 0 1 1 1 0 1 0.

Solution We know that in unipolar NRZ line code, a binary symbol 0 is represented by 0 V for the entire bit interval T_b and the binary symbol 1 is represented by a constant voltage level, say $+V$ or $-V$ during its entire bit interval T_b (where the bit rate is $1/T_b$).

Figure 2.1.9 depicts the unipolar NRZ line coding waveform for the given binary data sequence 0 1 0 0 1 1 1 0 1 0.

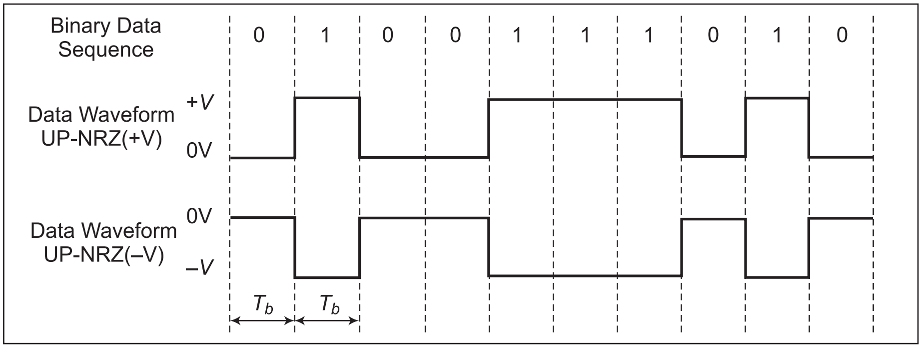


Figure 2.1.9 Unipolar NRZ Line Coding Waveform for Example 2.1.4

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.5 Draw the unipolar NRZ line coding waveform for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

It can be easily seen that in unipolar NRZ line code, the following holds good:

- Long strings of 1s and 0s produces high dc as well as low-frequency components which do not carry any information. For example, in case of an equal number of 1s and 0s, the average dc voltage level is equal to one-half of the nonzero voltage, that is, $V/2$ with 100% duty cycle.
- The clock recovery is difficult because there is no separation between the pulses and thereby causes synchronization problem.
- With NRZ line coding, a long sequence of either 1s or 0s produces a condition in which a receiver may lose its amplitude reference. This reference is needed for optimum determination of received 1s and 0s with clear-cut discrimination between them. Similar conditions may also arise when there is a significant imbalance in the number of 1s and 0s transmitted. This typical condition is called *dc wandering*.
- The unipolar NRZ line code format contains substantial power in its dc component. Usually, the dc component does not carry any useful information, so the power in it gets wasted.

If unipolar NRZ line code is used then the communications channel should be able to pass dc components which requires dc-coupled circuits. So the clock signal cannot be extracted from this code.

- NRZ line code has more energy of the pulse since pulse width extends for the complete bit period.

Where to use Unipolar NRZ Line Code?

Application

The unipolar NRZ is the simplest form of commonly used line code in internal computer waveforms. Unipolar NRZ format is very simple to generate—it requires only one dc power supply. Standard TTL and CMOS circuits can be used to implement it.

Type 2: Unipolar RZ Line Code

In *unipolar RZ line code*, the binary symbol 0 is represented by 0 V for the entire bit interval T_b and the binary symbol 1 is represented by a pulse of voltage level $+V$ which returns to zero (RZ) after one-half bit period ($T_b/2$).

Mathematically, unipolar RZ waveform can be expressed as

For symbol '0'; $v(t) = 0$ during $0 \le t < T_b$ interval

For symbol '1'; $v(t) = \begin{cases} +V \\ 0 \end{cases}$ for $\begin{cases} 0 \le t < (T_b/2) \\ (T_b/2) \le t < T_b \end{cases}$ interval

SOLVED EXAMPLE 2.1.6 **Unipolar RZ Line Code**

Illustrate the unipolar RZ line coding waveform for the binary data sequence 0 1 0 0 1 1 1 0 1 0.

Solution We know that in unipolar RZ line code, the symbol 0 is represented by 0 V for the entire bit interval T_b and a symbol 1 is represented by a pulse of voltage level $+V$ which returns to zero (RZ) after one-half bit period ($T_b/2$).

Figure 2.1.10 depicts a unipolar RZ line encoding waveform for the given binary data sequence 0 1 0 0 1 1 1 0 1 0.

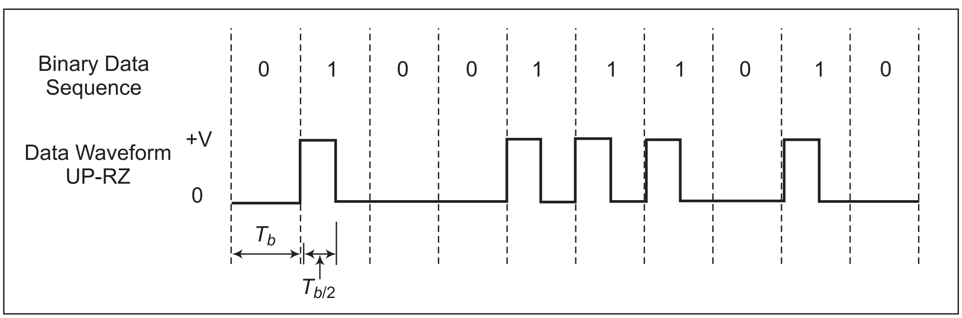


Figure 2.1.10 Unipolar RZ Line Coding Waveform for Example 2.1.6

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.7 Draw the unipolar RZ line coding waveform for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

It can be seen that in unipolar RZ line code, the following holds good:

- Since only one nonzero voltage level is used, each pulse is active for only 50% of a bit period.
- Assuming an equal probability of 1s and 0s occurring in a data sequence, the average dc voltage of a UP-RZ waveform is one-fourth of the nonzero voltage (i.e., $V/4$).
- The dc component is lower compared to that of unipolar NRZ.
- The synchronous data sequence with RZ line encoding signaling format has transitions of state for consecutive logic 1 data bits that are not present in NRZ formats. These transitions help in clock recovery for synchronous receivers. So, the clock recovery is better.
- Unipolar RZ line-code format contains a discrete impulse train at every clock frequency. This gives desirable self-clock extracting capability at the receiver. However, to maintain the same probability of error, it requires 3 dB more power as compared to that needed in binary signaling.

Application

Now where to use Unipolar RZ Line Code?

Unipolar RZ line codes are used in baseband data transmission and in magnetic tape-recording applications.

Type 3: Polar NRZ Line Code

In *polar NRZ-Level (NRZ-L) line code*, a binary symbol 0 is represented by one voltage level (say, $-V$) for the entire bit interval T_b and the binary symbol 1 is represented by another voltage level (say, $+V$) for its entire bit interval T_b (where bit rate is $1/T_b$), and, hence, called polar non-return-to-zero (polar NRZ-L).

Mathematically, polar NRZ-L waveform can be expressed as

For symbol '0'; $v(t) = -V$ during $0 \le t < T_b$ interval

For symbol '1'; $v(t) = +V$ during $0 \le t < T_b$ interval

SOLVED EXAMPLE 2.1.8 **Polar NRZ-L Line Code**

Illustrate the polar NRZ-L line coding waveform for the binary data sequence 0 1 0 0 1 1 1 0 1 0.

Solution We know that in polar NRZ-Level (NRZ-L) line code, a binary symbol 0 is represented by one voltage level (say, $-V$) for the entire bit interval T_b and the binary symbol 1 is represented by another voltage level (say, $+V$) for its entire bit interval T_b.

Figure 2.1.11 depicts the polar NRZ-L line-coding waveform for the given binary data sequence 0 1 0 0 1 1 1 0 1 0.

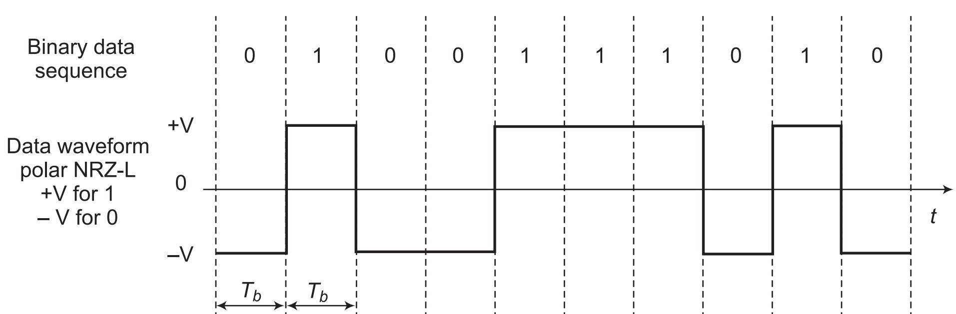


Figure 2.1.11 Polar NRZ-L Line-Coding Waveform for Example 2.1.8

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.9 Draw the polar NRZ-L line coding waveform for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

It can be seen that in polar NRZ-L line coding, the following holds good:

- A problem can arise when the binary data contains a long string of 0s or 1s. The receiver clock signal may or may not be synchronized with the source clock signal.
- The average dc value is minimum, and if the probability of occurrence of symbols '0' and '1' are same then it would be even zero.
- As compared to unipolar NRZ (signal level change between 0 V and +*V*), polar NRZ-L has higher transmission power requirement (due to signal level change between –*V* and +*V*) although the power spectral density (PSD) is same.
- Polar NRZ-L has better noise immunity, easy detection of binary levels 0 and 1 at receiver and low probability of error.

Where to use Polar NRZ-L Line Code?

Application

Polar NRZ-L line code is extensively used in digital logic circuits. We must remember that polar line code uses two distinct nonzero symmetrical but opposite voltage levels (+***V*** and –***V***) to represent the symbols 1 and 0, respectively, and there is no 0 V level. Accordingly, there is only polar NRZ-L line coding technique, not polar RZ.

ATTENTION

Differential line-coding techniques such as polar NRZ-M (M stands for Mark), a binary data 1 (or mark) is represented by a change in voltage level from its previously held voltage level, and a binary data 0 is represented by no change in voltage level from its previous one (that is, it remains same). Polar NRZ-M is primarily used in magnetic tape recording. Polar NRZ-S (S stands for Space) is complement of NRZ-M.

Type 4: Manchester Polar Line Code

In *Manchester polar line code*, a binary symbol 0 is represented by a –*V* pulse during first half of the bit period followed by a +*V* pulse during the second half of the bit period, and a binary symbol 1 is represented by a +*V* pulse during the first half of the bit period followed by a –*V* pulse during the second half of the bit period. Due to this reason, this type of line code is also known as *split phase* or *biphase*.

Mathematically, Manchester polar line coding waveform can be expressed as

For symbol '0'; $$v(t) = \begin{cases} -V \\ +V \end{cases} \quad \text{for} \begin{cases} 0 \le t < (T_b/2) \\ (T_b/2) \le t < T_b \end{cases}$$

For symbol '1'; $$v(t) = \begin{cases} +V \\ -V \end{cases} \quad \text{for} \begin{cases} 0 \le t < (T_b/2) \\ (T_b/2) \le t < T_b \end{cases}$$

SOLVED EXAMPLE 2.1.10 **Manchester Polar Line Code**

Illustrate the Manchester polar line-coding waveform for the binary data sequence given as 0 1 0 0 1 1 1 0 1 0.

Solution We know that in *Manchester polar line code*, a binary symbol 0 is represented by a –*V* pulse during the first half of the bit period followed by a +*V* pulse during the second half of the bit period, and a binary symbol 1 is represented by a +*V* pulse during the first half of the bit period followed by a –*V* pulse during the second half of the bit period.

Figure 2.1.12 depicts the Manchester polar line-coding waveform for the given binary data sequence 0 1 0 0 1 1 1 0 1 0.

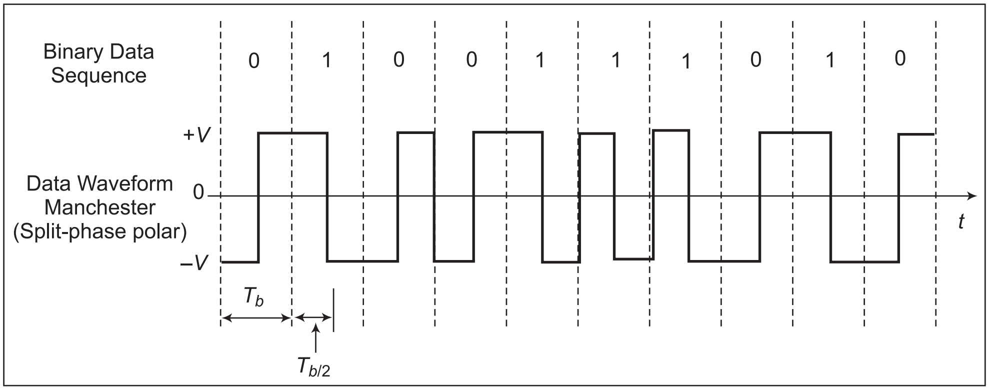


Figure 2.1.12 Manchester Polar Line-Coding Waveform for Example 2.1.10

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.11 Draw the Manchester polar line-coding waveform for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

Thus, for the Manchester line encoding signaling format, a data bit is split into two parts: The voltage level during the first half of the data bit interval is the inverse of the voltage level during the second half of the data bit interval. This clearly means that at the midway point, the level is inverted. This midway transition occurs at the middle of each data bit interval and serves as the recovered clock signal at the receiver. The level transition in the middle of every Manchester encoded data bit is used as a clock signal.

It is seen that in the Manchester line code, the following holds good:

- The transition occurs from $+V$ to $-V$ or from $-V$ to $+V$ at the middle of each bit period (that is, the split-phase signal passes through zero at least once per bit duration or clock cycle), the recovery of clock information at the receiver becomes easier.
- Since the biphase signal voltage changes level more frequently than does the NRZ signal, we naturally expect that its spectrum will have higher frequency components than are present in the NRZ signal and does not cause unnecessary dc level.
- Manchester line encoding has a distinct advantage of facilitating easy clock recovery because it has a built-in clock capability at the center of each data bit interval.
- Its disadvantage is that the interpretation of the data is dependent on the level of the first half of the bit period.
- The bandwidth occupied is approximately double that of polar NRZ-L line code.

Where is Manchester line code used?

Application

Manchester line code is specified in IEEE 802.3 standard for Ethernet Local Area Network (LAN). It is mostly used in satellite telemetry communication links, optical communications, and magnetic tape-recording systems.

Type 5: Differential Manchester Polar Line Code

In *differential Manchester polar line code*, the binary symbol (1 or 0) is represented by a transition for symbol 0 and no transition for binary 1 at the beginning of the bit, in addition to transition at the middle of the bit interval.

SOLVED EXAMPLE 2.1.12 Differential Manchester Polar Line Code

Illustrate the differential Manchester polar line-coding waveform for the binary data sequence given as 0 1 0 0 1 1 1 0 1 0.

Solution We know that in the differential Manchester polar line code, the binary symbol 0 is represented by a transition at the beginning of the bit, followed by transition at the middle of the bit interval too, and the binary symbol 1 is represented by no transition at the beginning of the bit, but followed by transition at the middle of the bit interval.

Figure 2.1.13 depicts the differential Manchester polar line-coding waveform for the given binary data sequence 0 1 0 0 1 1 1 0 1 0.

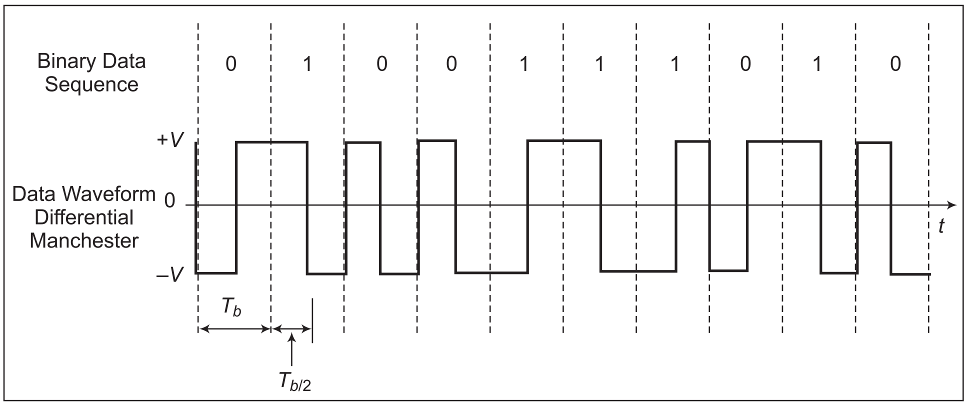


Figure 2.1.13 Differential Manchester Polar Line-Coding Waveform for Example 2.1.12

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.13 Draw the Differential Manchester polar line-coding waveform for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

It may be noted here that

- Differential Manchester line code requires only one transition to represent binary 1 but two transitions to represent binary 0 in order to achieve better synchronization, and
- It is more complex to design and implement.

SOLVED EXAMPLE 2.1.14 Differential Manchester vs Manchester Line Code

What are the similarities and dissimilarities between differential Manchester line code and Manchester line code? Illustrate them with the help of their respective line-coding waveforms for the binary data sequence given as 0 1 0 0 1 1 1 0 1 0.

Solution Both differential Manchester and Manchester line codes provide a clock transition at the midpoint of each data bit period.

Differential Manchester line code moves the detection of the data level to the leading edge of a data bit interval. It accomplishes this by using a comparison of the data levels at the beginning of a data bit to determine if the bit is a 1 or a 0. If there is a change of state at the beginning of the data bit interval then the data bit is 0. If there is no change of state, then the data bit is a 1.

Manchester line code detects the data based on transition from low to high (for bit 0), or high to low (for bit 1) at the middle of a data bit interval.

This is illustrated in Figure 2.1.14.

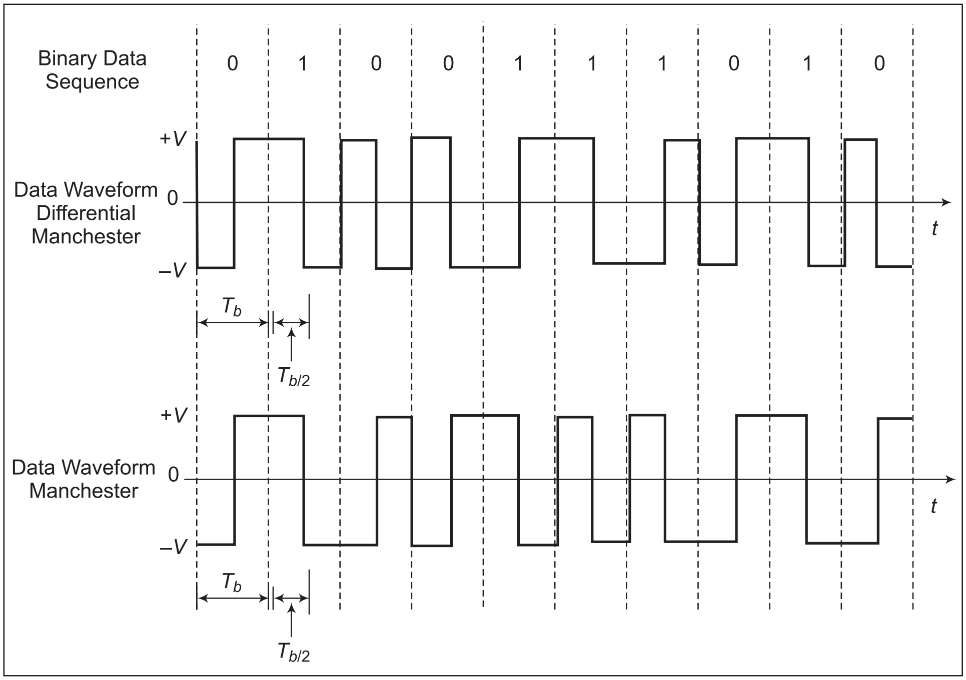


Figure 2.1.14 Manchester and Differential Manchester Line-Coding Waveforms

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.15 Illustrate the similarities and dissimilarities between differential Manchester and Manchester line codes with the help of their respective line-coding waveforms for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

Bipolar Line Codes

Many line codes use three voltage levels (+*V*, 0, and –*V*), instead of two voltage levels (+*V* and 0, or 0 and –*V*, or +*V* and –*V*) as used in polar line codes, to encode the binary data. These are known as *bipolar line codes* or *multilevel binary line codes* or *dicodes* or *duobinary signaling*. Bipolar NRZ Alternate Mark Inversion (BP-NRZ-AMI), High-Density Bipolar (HDB) NRZ-AMI, Bipolar RZ (BP-RZ), and BP-RZ-AMI type of line-coding techniques belong to this category of line codes.

Type 6: Bipolar NRZ-AMI Line Code

In *bipolar NRZ-AMI line coding*, the following holds good:

- The binary symbol 1s are represented alternatively by +*V* and –*V* voltage levels, and the binary symbol 0 is represented by 0 V.
- It means that if first binary symbol 1 is represented by +*V*, then the next binary symbol 1 will be represented by –*V*, and the third binary symbol 1 will be represented again by +*V*, and so on.
- Thus, the alternate transitions occur even when consecutive binary data 1s occur.
- The binary symbol 0 is always represented by 0 V level.

SOLVED EXAMPLE 2.1.16 **Bipolar NRZ-AMI Line Code**

Illustrate the bipolar-NRZ-AMI line-coding waveform for the binary data sequence specified as 0 1 0 0 1 1 1 0 1 0.

Solution Figure 2.1.15 depicts the bipolar NRZ-AMI line-encoding waveform for the given binary data 0 1 0 0 1 1 1 0 1 0.

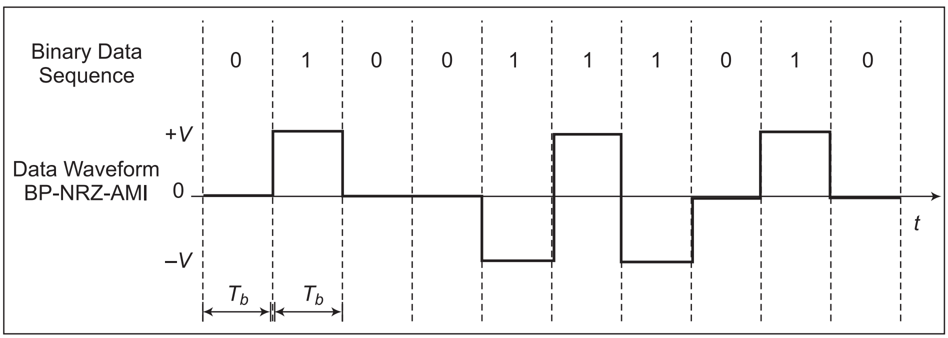


Figure 2.1.15 Bipolar NRZ-AMI Line-Coding Waveform for Example 2.1.16

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.17 Draw the Bipolar NRZ-AMI line coding waveform for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

It can be seen that in bipolar NRZ-AMI line code, the following holds good:

- Assuming an equal probability of the occurrence of 1s and 0s, the average dc level is 0 V with 100% duty cycle.
- However, there is a problem in synchronization when a long sequence of binary symbol 0s is present.
- Bipolar NRZ line code format contains substantial power in the dc component and there is no possibility of clock extraction from this code. Moreover, it requires two distinct power supplies.
- In BP-NRZ-AMI signaling format, the polarity of the encoded logic alternates between +*V* and –*V* voltage levels. As long as the logic 1 levels continue to alternate polarity, it is assumed that the received data is without any errors. If two consecutive logic 1 bits appear at the same polarity then a polarity violation seems to have occurred and the data is believed to be having errors.

- In BP-RZ-AMI signaling format, the voltage level returns to zero at the center of every logic 1 bit period. This adds more transitions between $+V$, 0 V, and $-V$ levels and helps in recovering clock signals for synchronous data transmissions. Alternate mark inversion uses polarity violation to detect the possibility of corrupted data.
- Bipolar AMI-RZ line-code format is more prone to error. To maintain the same probability of error, it requires additional 3 dB power as compared to other line-code formats. A long sequence of zeroes may cause a loss of the clock signal. The receiver also requires distinguishing three distinct voltage levels.[5]

Type 7: Bipolar RZ Line Code

In *bipolar RZ line code*, the binary symbol 0 and 1 are represented by opposite-level pulses ($-V$ and $+V$) during the first half of the bit period, respectively, and 0 V during the second half of the bit period.

Mathematically, in a bipolar RZ waveform,

For binary '0'; $$v(t) = \begin{cases} -V & \text{for } 0 \le t < (T_b/2) \\ 0 & (T_b/2) \le t < T_b \end{cases}$$

For binary '1'; $$v(t) = \begin{cases} +V & \text{for } 0 \le t < (T_b/2) \\ 0 & (T_b/2) \le t < T_b \end{cases}$$

SOLVED EXAMPLE 2.1.18 **Bipolar RZ Line Code**

Illustrate the bipolar RZ line coding waveform for the binary data sequence given as 0 1 0 0 1 1 1 0 1 0.

Solution We know that in bipolar RZ line code, the binary symbol 0 is represented by $-V$ voltage level during the first half of the bit period and 0 V during its second half, whereas the binary symbol 1 is represented by $+V$ voltage level during the first half of the bit period and 0 V during the second half of the bit period. Figure 2.1.16 depicts bipolar RZ line-coding waveform for given binary data sequence 0 1 0 0 1 1 1 0 1 0.

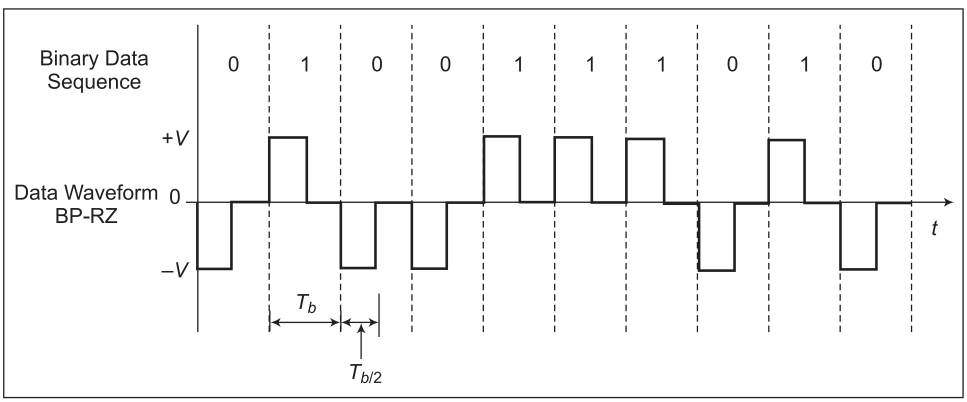


Figure 2.1.16 Bipolar RZ Line-Coding Waveform for Example 2.1.18

[5] Coded Mark Inversion (CMI) is a two-level NRZ code which has higher clock content. In CMI, logic 0 is represented by a negative pulse followed by a positive pulse, each of half-bit duration, and logic 1 is represented by $+V$ and $-V$ for complete bit duration.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.19 Draw the bipolar RZ line-coding waveform for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

It can be easily seen that in bipolar RZ line code, the following holds good:

- The transition in every bit period ensures better synchronization.
- Each pulse is active only for 50% of a bit period, so the average dc level of a BP-RZ waveform is 0 V (assuming an equal probability of 1s and 0s occurring in binary data sequence).
- Moreover, it performs better in the presence of noise.
- However, it has higher transmission power requirement although the power spectral density is similar to the polar RZ line-coding format.
- The main disadvantage of bipolar RZ line coding is that it occupies more bandwidth.

Type 8: Bipolar-RZ-AMI Line Code

In *Bipolar Return-to-Zero Alternate-Mark-Inversion (BP-RZ-AMI) line code*, the binary symbol 0 is represented by 0 V for the entire bit interval T_b, and the binary symbol 1 is represented by alternating pulses of $+V$ and $-V$ voltage levels which returns to zero (RZ) after one-half bit period ($T_b/2$).

Mathematically, bipolar RZ-AMI waveform can be expressed as

For symbol '0'; $v(t) = 0$ during $0 \le t < T_b$ interval

For symbol '1'; $v(t) = \begin{cases} +V, \text{ or } -V \text{ alternatively} \\ 0 \end{cases}$ for $\begin{cases} 0 \le t < (T_b/2) \\ (T_b/2) \le t < T_b \end{cases}$

SOLVED EXAMPLE 2.1.20 **Bipolar RZ AMI Line Code**

Illustrate the BP-RZ-AMI line coding waveform for the binary data sequence given as 0 1 0 0 1 1 1 0 1 0.

Solution We know that in the bipolar return-to-zero alternate-mark-inversion (BP-RZ-AMI) line code, the binary symbol 0 is represented by 0 V for the entire bit interval T_b, and the binary symbol 1 is represented by alternating pulses of $+V$ and $-V$ voltage levels which returns to zero (RZ) after one-half bit period ($T_b/2$).

Figure 2.1.17 depicts BP-RZ-AMI line-coding waveform for given binary data sequence 0 1 0 0 1 1 1 0 1 0.

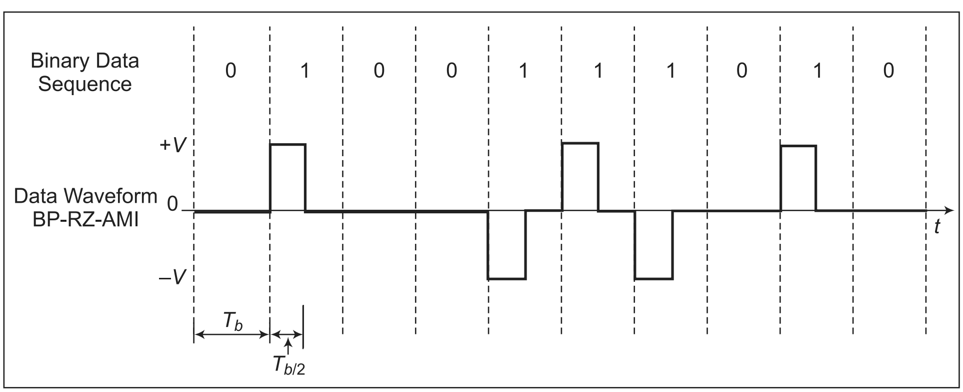


Figure 2.1.17 BP-RZ-AMI Line-Coding Waveform for Example 2.1.20

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.21 Draw the bipolar RZ-AMI line-coding waveform for the binary data sequence given as 1 0 1 1 0 0 0 1 0 1.

It is seen that in the bipolar RZ-AMI line code, the following holds good:

- Due to alternating $+V$ and $-V$ voltage levels for representing binary symbol 1, and 0V for representing binary symbol 0, the average dc voltage level is approximately 0V, regardless of the presence of number of 1s and 0s in binary data sequence.
- Bipolar RZ-AMI line code format is the most efficient for ac coupling as its power spectra is null at dc.
- Due to alternating binary 1s, a single error can be easily detected.
- The clock signal can be easily extracted by converting RZ-AMI to unipolar RZ.

Application

Where does bipolar RZ-AMI line code find applications?

BP-RZ-AMI line-coding technique is used in telephone systems as signaling scheme, and T-carrier lines with +3 V, 0 V, and –3 V voltage levels to represent binary data.

Type 9: High Density Bipolar (HDB) NRZ-AMI Line Code

Contrary to linear line codes described so far, High Density Bipolar (HDB) is a scrambling-type technique of line code, which provides synchronization without increasing the number of bits. In HDB-NRZ-AMI line coding,

- Some pre-defined number of pulses are added when the number of consecutive binary symbol 0s exceeds an integer value n.
- It is denoted as HDB_n, where $n = 1, 2, 3, ...$
- In HDB_n coding, when the input data sequence contains consecutive $(n + 1)$ zeros, this group of zeros is replaced by special $(n + 1)$ binary digit sequence.
- These special data sequences consist of some binary 1s so that they may be detected at the receiver reliably.

For example, when $n = 3$, the special binary sequences used are 000V and *B* 00V, where *B* (*B* stands for Bipolar) and *V* (*V* stands for Violation to AMI rule of encoding) are considered 1s. This means that four consecutive 0s are replaced with a sequence of either 000V or B00V, as explained in the following solved example.

SOLVED EXAMPLE 2.1.22 **Bipolar HDB_3 NRZ-AMI Line Code**

Illustrate the bipolar HDB_3 NRZ-AMI line-coding waveform for the binary data sequence given as 1 1 1 0 0 0 0 1 0 1 1 0 1 0 0 0 0 0 0 0 0 1.

Solution Figure 2.1.18 depicts the bipolar HDB_3 NRZ-AMI line coding waveform for the given binary data 1 1 1 0 0 0 0 1 0 1 1 0 1 0 0 0 0 0 0 0 0 1.

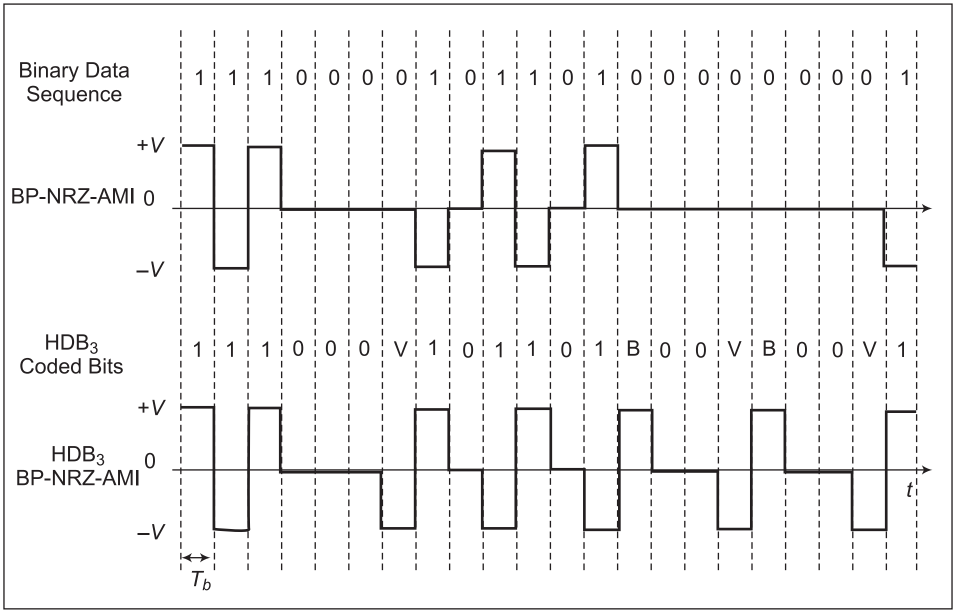


Figure 2.1.18 HDB$_3$ NRZ-AMI Line-Coding Waveform for Example 2.1.22

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.23 Draw the bipolar HDB$_3$ NRZ-AMI line-coding waveform for the binary data sequence given as 1 0 1 0 0 0 0 1 0 1 1 1 0 1 0 0 0 0 0 0 0 0 0 1 1.

It is seen that in High Density Bipolar (HDB)[6] NRZ-AMI line code, the following holds good:

- The problem of synchronization in bipolar NRZ-AMI line code is eliminated in HDB$_3$ NRZ-AMI line code.
- Despite deliberate bipolar violations, HDB$_n$ line coding retains error detecting capability.
- The dc wandering condition is avoided.
- The dc null condition in its power spectral density is maintained, as in case of conventional bipolar line codes.

Type 10: Binary Eight Zeros Substitution (B8ZS) RZ-AMI Line Code

Just like HDB$_3$, B8ZS is another scrambling-type technique of line code, which provides synchronization without increasing the number of bits. In B8ZS-RZ-AMI line coding, whenever eight consecutive binary 0s appear in given binary data sequence, one of two special bit sequence – + – 0 – + 0 0 0 or – + 0 + – 0 0 0 (where + and – represent bipolar logic 1 conditions) is substituted for eight consecutive 0s.

SOLVED EXAMPLE 2.1.24 **B8ZS-RZ-AMI Line Code Type I**

Illustrate the B8ZS-RZ-AMI line-coding waveform for the binary data sequence given as 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0. Use special bit sequence as + – 0 – + 0 0 0.

[6] High Density Bipolar (HDB) NRZ-AMI line code is a scrambling-type technique of line code, which provides synchronization without increasing the number of bits. HDB$_3$ line code is used in E1 digital carrier systems.

Solution Figure 2.1.19 illustrates waveforms for B8ZS-RZ-AMI line code using special bit sequence + – 0 – + 0 0 0 for the given binary data 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0.

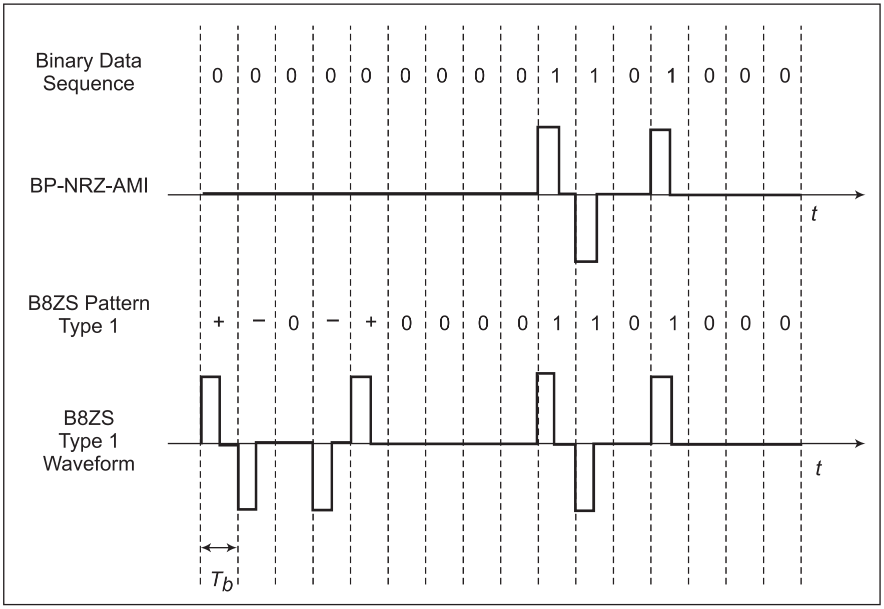


Figure 2.1.19 B8ZS Line-Coding Waveform for Example 2.1.24

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.25 Draw the B8ZS-RZ-AMI line-coding waveform for the binary data sequence given as 1 1 0 0 0 0 0 0 0 0 0 1 1 0 1 0. Use special bit sequence as + – 0 – + 0 0 0 1.

SOLVED EXAMPLE 2.1.26 **B8ZS-RZ-AMI Line Code Type II**

Illustrate B8ZS-RZ-AMI line-coding waveform for the binary data sequence given as 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0. Use special bit sequence as – + 0 + – 0 0 0.

Solution Figure 2.1.20 illustrates waveforms for B8ZS signaling format type II using alternate bit sequence – + 0 + – 0 0 0 for the given binary data 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0.

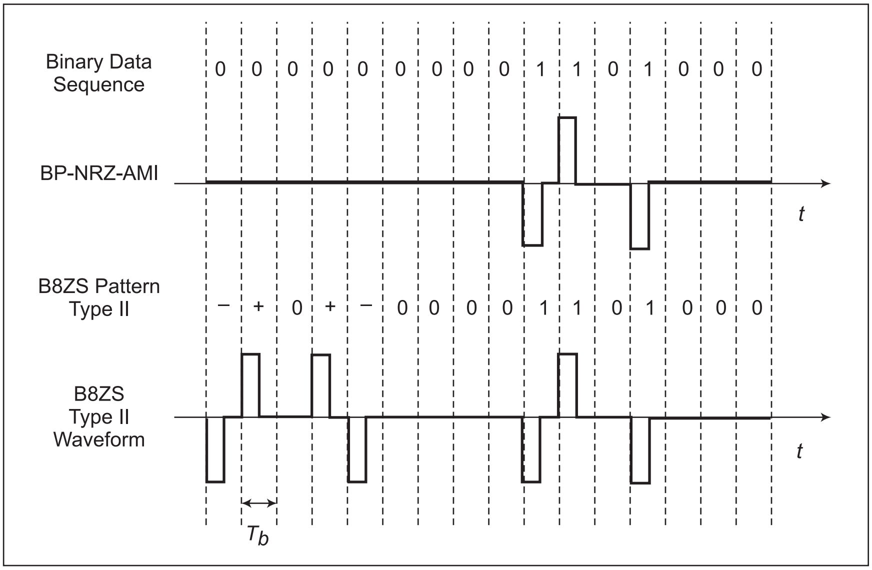


Figure 2.1.20 B8ZS Line-Coding Waveform for Example 2.1.26

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.1.27 Draw the B8ZS-RZ-AMI line-coding waveform for the binary data sequence given as 1 1 0 0 0 0 0 0 0 0 0 1 1 0 1 0. Use special bit sequence as – + 0 + – 0 0 0.

It may be noted that the following holds good:

- B8ZS line code ensures that sufficient transitions occur in the data to maintain clock synchronization.
- The receiver will detect the bipolar violations in the fourth and seventh bit positions and then substitute the eight 0s back into the data sequence.

Name typical application areas of B8ZS line Code. **Application**

Bipolar violation means two consecutive binary logic 1s with the same voltage polarity. B8ZS encoding is used to avoid conventional bipolar encoding of 8 consecutive 0s. Therefore, the B8ZS line-coding technique is widely used in T1 digital carrier systems.[7]

In B6ZS RZ-AMI line-code, one of two special bit sequences 0 – + 0 + –, or 0 + – 0 – + (where + and – represents bipolar logic 1 condition) is substituted for six consecutive 0s in a given binary data sequence. It ensures adequate number of transitions necessary for proper clock synchronization, and is used in T2 digital carrier systems. Similarly, in B3ZS RZ-AMI line code, one of four special bit sequences 0 0 –, – 0 –, 0 0 +, or + 0 + is substituted for three consecutive 0s. It is used in T3 digital carrier systems.

[7] There is a block-type line code, 2BIQ, abbreviated for 2 binary digits encoded into quaternary symbol, having the block size of 2. The block to symbol transformation is as follows: 00 → –3 V, 01 → –1 V, 10 → +1 V, 11 → +3 V. This results in 4-level PAM signal having zero dc component, less bandwidth and better performance with respect to intersymbol interference and cross talk as compared with linear codes.

2.1.3 Power Spectra of Line Codes

Recap

The term 'Power Spectral Density (PSD)' $P_X(f)$ of a signal represents the power per unit bandwidth (in Hz) of the spectral components at the frequency f. The PSD is a positive, real, and an even function of f. It is expressed in volts squared per Hz. Since the autocorrelation function for random binary pulse signals (as given by various line codes) can be determined from their statistical information, we can directly apply the relationship between the PSD (spectral information) and the autocorrelation function. The power spectral density of various line codes depends on the signaling pulse waveform.

What We Discuss Here...

Power spectral densities of the unipolar NRZ, polar NRZ, bipolar NRZ-AMI and Manchester line codes. A comparison between them is also presented.

Note *For derivation of their expressions, the readers may refer to Appendix C—Power Spectra of Discrete PAM Signals.*

PSD of Unipolar NRZ Line Code

The expression for *PSD of unipolar NRZ line code* is given as

$$\boxed{P_X(f) = \frac{V^2 T_b}{2} \operatorname{sinc}^2(f T_b) + \frac{V^2}{4}\delta(f)}$$

It is observed that the presence of the Dirac delta function $\delta(f)$ in the second term accounts for one half of the power contained in the unipolar NRZ data format. Specifically, the power spectral density $P_X(f)$ is normalized with respect to $V^2 T_b$, and the frequency f is normalized with respect to the bit rate $1/T_b$.

Figure 2.1.21 shows the power spectral density of unipolar NRZ line-code waveform.

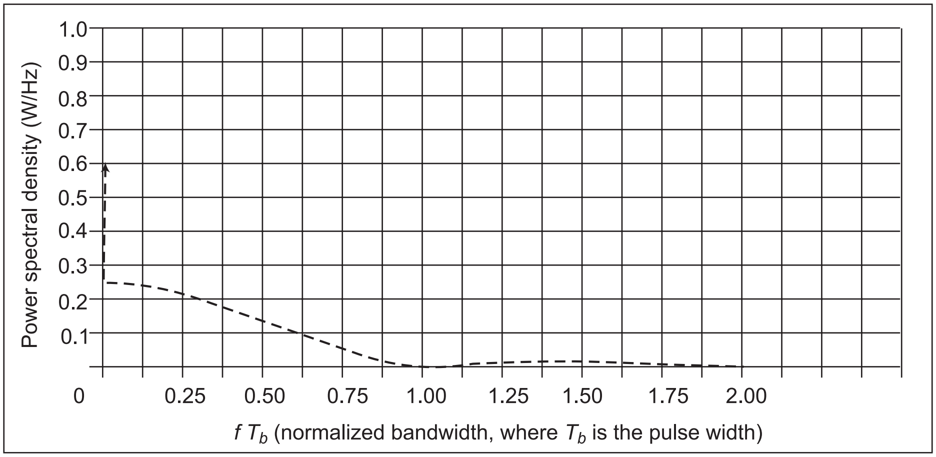


Figure 2.1.21 PSD of Unipolar NRZ Line Code

The PSD of unipolar NRZ line code has the following properties:

- The value of $P_X(f)$ is maximum at $f = 0$.
- The value of $P_X(f)$ is significant at all multiples of the bit rate $f_b = 1/T_b$.
- The peak value of $P_X(f)$ between f_b and $2f_b$ occurs at $f = 1.5 f_b$, and is 14 dB lower than the peak value of $P_X(f)$ at $f = 0$.

- The main lobe centered around $f = 0$ has 90% of the total power.
- When the NRZ signal is transmitted through an ideal low-pass filter with cut-off frequency at $f = f_b$, the total power is reduced by 10% only.
- The NRZ signal has no dc component.
- The power in the frequency range from $f = 0$ to $f = \pm \Delta f$ is $2G(f)\,\Delta f$.
- In the limit as Δf approaches to zero, the PSD becomes zero.

The expression for *PSD of Polar NRZ line code* is given as

PSD of Polar NRZ Line Code

$$\boxed{P_X(f) = \frac{V^2 T_b}{2} \sin c^2 (f\,T_b)}$$

Specifically, the power spectral density $P_X(f)$ is normalized with respect to $V^2 T_b$, and the frequency f is normalized with respect to the bit rate $1/T_b$.

Figure 2.1.22 shows the power spectral density of polar NRZ line-code waveform.[8]

The PSD of Polar NRZ line code has the following properties:

- Most of the power of the polar NRZ line code lies within the main lobe of the sine-shaped curve, which extends up to the bit rate $1/T_b$.
- In general, the polar signaling format has most of its power concentrated near lower frequencies.
- Theoretically, the power spectrum becomes very small as frequency increases but never becomes zero above a certain frequency.
- The first non-dc null frequency, known as *essential bandwidth*, is R_b Hz where R_b is the clock frequency. The Nyquist bandwidth required to transmit R_b pulses per second is $R_b/2$.

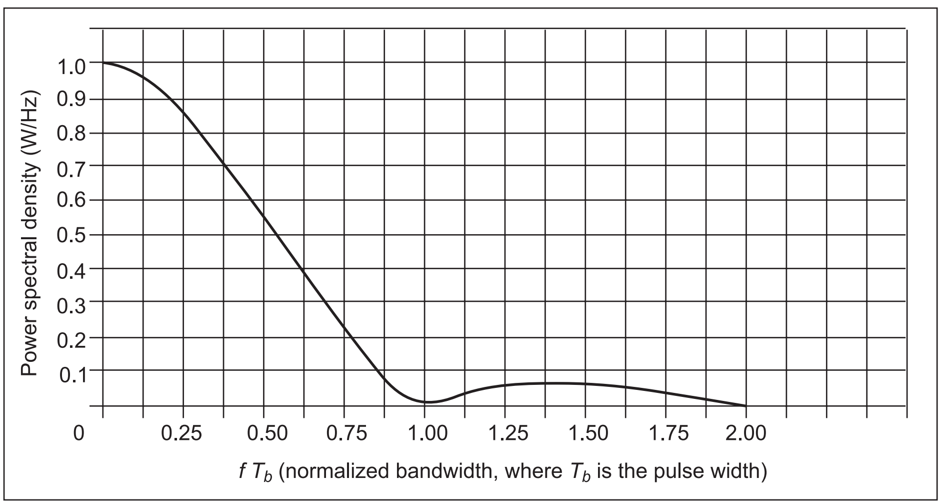


Figure 2.1.22 PSD of Polar NRZ Line Code

[8] Polar line code is the most efficient from the power requirement viewpoint. Although they are not the most bandwidth efficient. For a given power, the error-detection probability is the lowest among all line codes. There is no discrete clock-frequency component in the spectrum of the polar line code, yet the rectification of the polar RZ code yields a periodic signal of clock frequency which can be used to extract timing and synchronization purpose.

IMPORTANT! This shows that the essential bandwidth of polar NRZ is twice the Nyquist bandwidth. So we can say that the essential bandwidth of polar RZ line code will be $2R_b$ Hz which is four times the Nyquist bandwidth required to transmit R_b pulses per second.

PSD of Bipolar NRZ-AMI Line Code

In bipolar NRZ-AMI line code (also known as pseudo-ternary signaling), $+V$ and $-V$ levels are used alternately for transmission of successive 1s, and no pulse for transmission of binary 0 (that is, 0V). With consecutive pulses alternating, we can avoid dc wandering condition and thus cause a dc null in the PSD.

Assume The occurrence of binary 0s and 1s of a random binary sequence is equally probable.

The expression for PSD of *bipolar NRZ-AMI line code* is given as

$$\boxed{P_X(f) = V^2 T_b \operatorname{sinc}^2(f T_b) \sin^2(\pi f T_b)}$$

Specifically, the power spectral density $P_X(f)$ is normalized with respect to V^2T_b, and the frequency f is normalized with respect to the bit rate $1/T_b$.

Figure 2.1.23 shows the power spectral density of a bipolar NRZ-AMI line code.

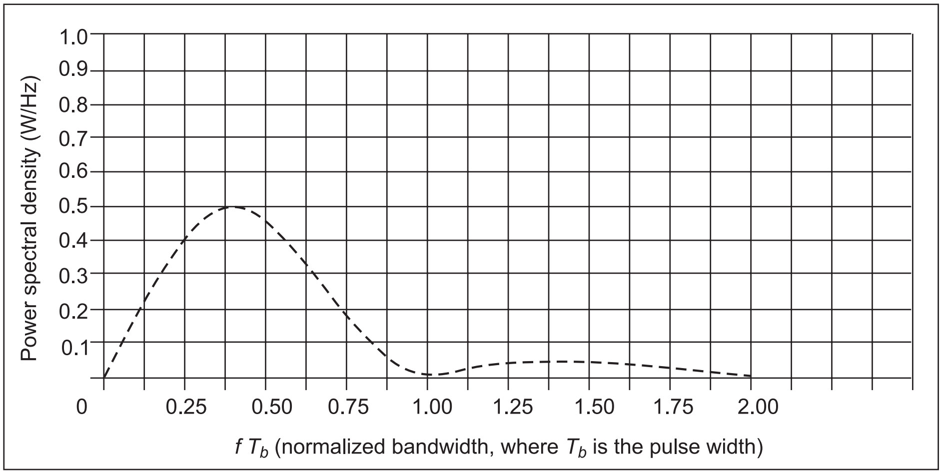


Figure 2.1.23 Power Spectra of Bipolar NRZ-AMI Line Code

The PSD of bipolar NRZ-AMI line code has the following properties:

- The value of $P_X(f)$ is zero at $f = 0$.
- The main lobes extend from $f = 0$ to $f = 2f_b$.
- The main lobes have peaks at approximately $\pm\dfrac{3f_b}{4}$.
- When the bipolar NRZ signal is transmitted through an ideal low-pass filter, with cut-off frequency at $f = 2f_b$, 95% of the total power will be passed.
- When the bipolar NRZ signal is transmitted through an ideal low-pass filter, with cut-off frequency at $f = f_b$, then only approximately 70% of the total power is passed.[9]

[9] The spectrum of the bipolar NRZ line code has higher frequency components than are present in the unipolar or polar NRZ signal.

This shows that the essential bandwidth of the bipolar NRZ-AMI as well as bipolar RZ signal is R_b, which is half that of the polar RZ signal and twice the Nyquist bandwidth ($R_b/2$). Although most of the power lies within a bandwidth equal to the bit rate $1/T_b$, the spectral content is relatively small around zero frequency or dc.

IMPORTANT!

- Assume that the input binary data consisting of independent and equally likely symbols.
- The autocorrelation function of Manchester line code is same as that for the polar RZ line code.

PSD of Manchester Line Code

The expression for *PSD of Manchester line code* is given as

$$\boxed{P_X(f) = V^2 T_b \operatorname{sinc}^2\left(\frac{fT_b}{2}\right)\sin^2\left(\frac{\pi f T_b}{2}\right)}$$

Specifically, the power spectral density $P_X(f)$ is normalized with respect to V^2T_b, and the frequency f is normalized with respect to the bit rate $1/T_b$. It can be easily seen that the PSD of the Manchester line code has a dc null. The PSD of Manchester line code is shown in Figure 2.1.24 along with the power spectral density of unipolar NRZ, polar NRZ, and bipolar NRZ line codes for comparison purpose.

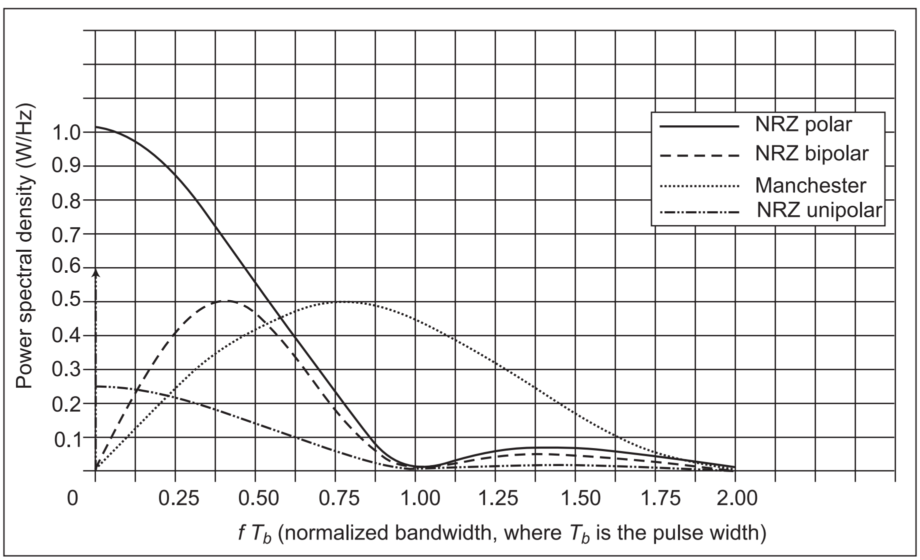


Figure 2.1.24 Power Spectra of different Line Codes

- In case of bipolar NRZ, most of the power lies within a bandwidth equal to the bit rate $1/T_b$.
- The spectral content of a bipolar NRZ line code is relatively small around zero frequency as compared to that of unipolar NRZ and polar NRZ line-code waveforms.
- In case of Manchester line code, most of the power lies within a bandwidth equal to the bit rate $2/T_b$, which is twice that of unipolar NRZ, polar NRZ, and bipolar NRZ line codes.

IMPORTANT!

2.1.4 Attributes of Line Coding Techniques — A Comparison

Table 2.1.1 shows the comparison of advantages and disadvantages of unipolar line codes, i.e., UP-NRZ and UP-RZ.

Table 2.1.1 Comparison of Unipolar Line Codes

Parameter	UP-NRZ	UP-RZ
Advantages	• Simplicity in implementation. • Doesn't require a lot of bandwidth for transmission.	• Simplicity in implementation. • Presence of a spectral line at symbol rate which can be used as symbol timing clock signal.
Disadvantages	• Presence of dc level (indicated by spectral line at 0 Hz in its PSD). • Contains low-frequency components. • Does not have any error correction capability. • Does not possess any clocking component for ease of synchronization. • Is not transparent. Long string of zeros causes loss of synchronization.	• Presence of dc level (indicated by spectral line at 0 Hz in its PSD). • Continuous part is non-zero at 0 Hz. • Does not have any error correction capability. • Occupies twice as much bandwidth as Unipolar NRZ. • Is not transparent.

Figures 2.1.25 and 2.1.26 shows PSD waveforms of unipolar non-return-to-zero (UP- NRZ) and unipolar return-to-zero (UP-RZ) line codes, respectively, for comparison.

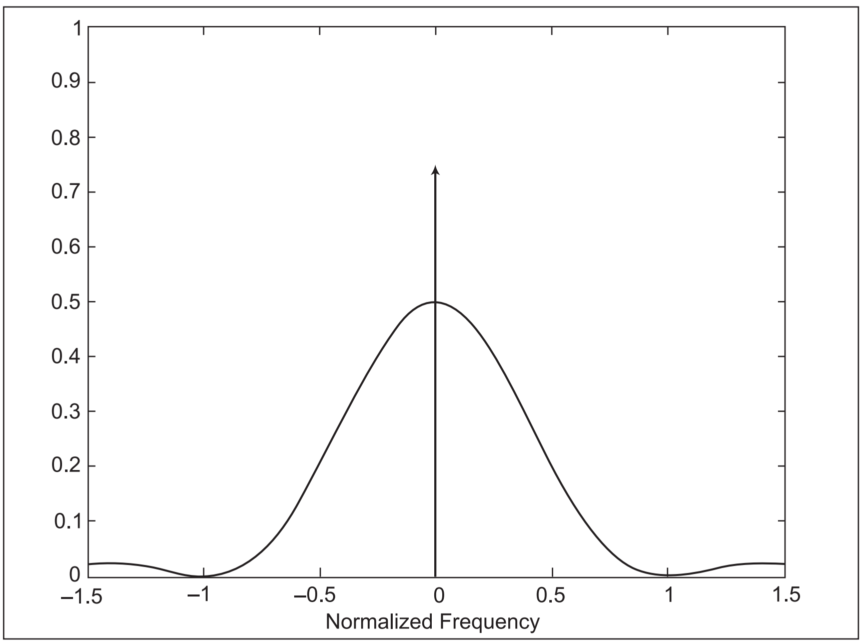


Figure 2.1.25 PSD of Unipolar NRZ Line Code

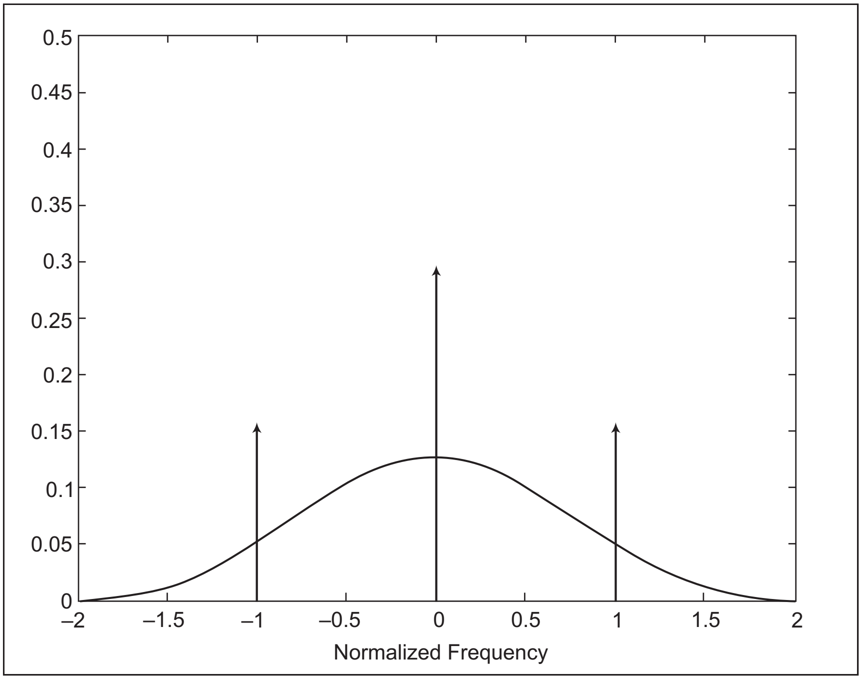


Figure 2.1.26 PSD of Unipolar RZ Line Code

Note *When unipolar NRZ signals are transmitted over links with either transformer or ac coupled repeaters, the dc level is removed converting them into a polar format. The continuous part of the PSD is also non-zero at 0 Hz (i.e., contains low-frequency components). This means that ac coupling will result in distortion of the transmitted pulse shapes. The ac coupled transmission lines typically behave like high-pass RC filters and the distortion takes the form of an exponential decay of the signal amplitude after each transition. This effect is referred to as "Signal Droop". In conclusion, it can be said that neither type of unipolar signals is suitable for transmission over ac coupled lines.*

Table 2.1.2 shows the comparison of advantages and disadvantages of **polar line codes**, i.e., polar NRZ, polar RZ, and Manchester polar RZ.

Table 2.1.2 Comparison of Polar Line Codes

Parameter	Polar NRZ	Polar RZ	Manchester Polar RZ
Advantages	• Simplicity in implementation. • No dc component.	• Simplicity in implementation. • No DC component.	• No dc component. • Suitable for transmission over ac coupled lines. • Easy to synchronize. • Is transparent.

Disadvantages	• Continuous part is non-zero at 0 Hz. • Does not have any error correction capability. • Does not possess any clocking component for ease of synchronization. • Is not transparent.	• Continuous part is non-zero at 0 Hz. • Does not have any error-correction capability. • Does not possess any clocking component for easy synchronization. However, clock can be extracted by rectifying the received signal. • Occupies twice as much bandwidth as polar NRZ.	• Because of the greater number of transitions it occupies a significantly large bandwidth. • Does not have error-detection capability.

Figures 2.1.27, 2.1.28, and 2.1.29 show PSD waveforms of polar NRZ, polar RZ, and Manchester polar RZ line codes, respectively, for comparison.

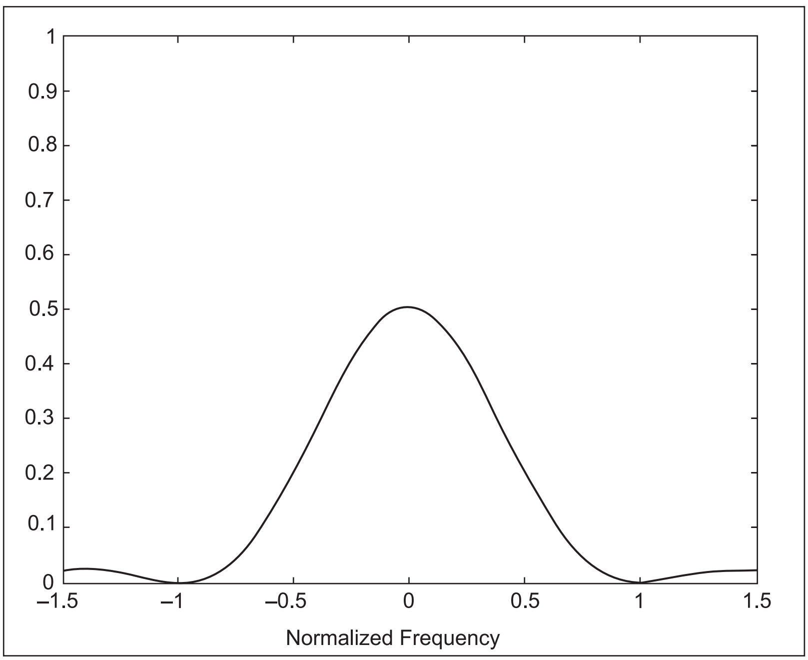


Figure 2.1.27 PSD of Polar NRZ Line Code

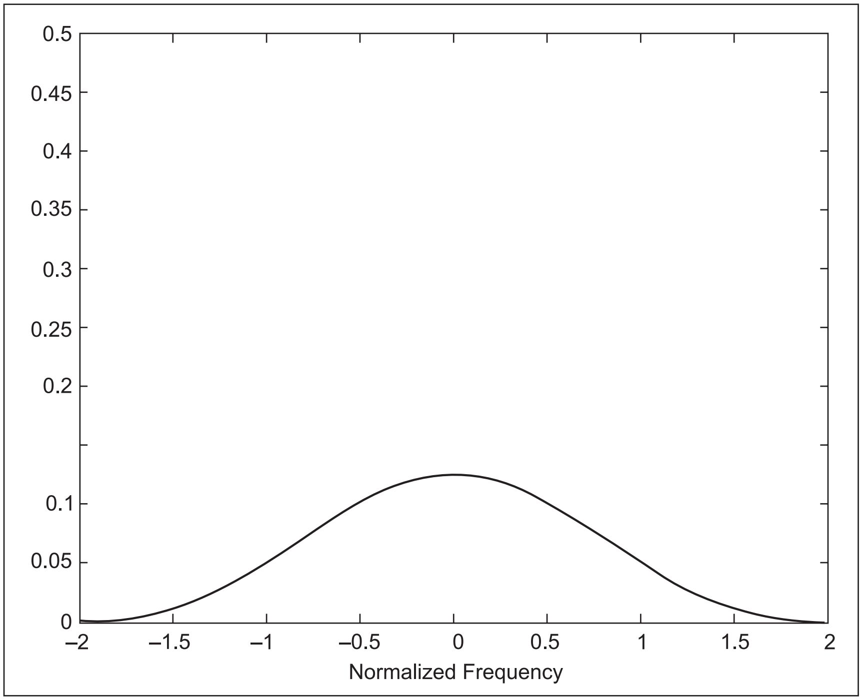


Figure 2.1.28 PSD of Polar RZ Line Code

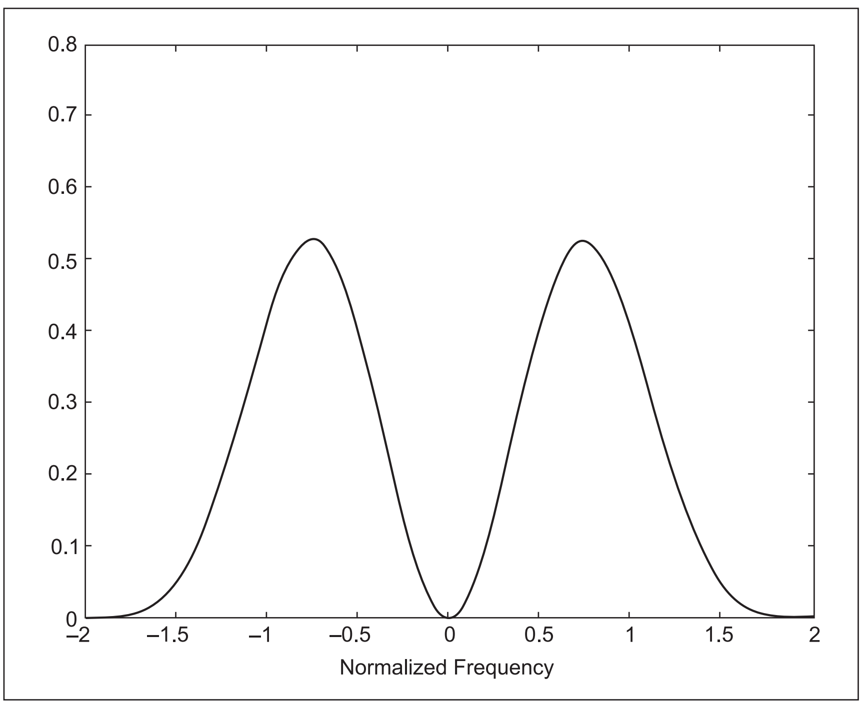


Figure 2.1.29 PSD of Manchester Polar RZ Line Code

IMPORTANT! Polar NRZ and RZ line codes have almost identical spectra to that of unipolar NRZ and RZ line codes. However, due to the opposite polarity of the 1 and 0 symbols, neither contain any spectral lines.

Manchester polar RZ line encoding is self-synchronizing. The transition at the center of every bit interval is used for synchronization at the receiver. Synchronization at the receiving end can be achieved by locking on to the transitions, which indicate the middle of the bits.

It is worth highlighting that the traditional synchronization technique used for unipolar, polar and bipolar schemes, which employs a narrow band-pass filter to extract the clock signal cannot be used for synchronization in Manchester encoding. This is because the PSD of Manchester encoding does not include a spectral line at symbol rate ($1/T_b$). Even rectification does not help.

The characteristics of Manchester RZ line code make it unsuitable for use in wide-area networks. However, it is widely used in local area networks such as Ethernet and Token Ring.

Table 2.1.3 shows the comparison of advantages and disadvantages of bipolar line codes, i.e., bipolar AMI NRZ, bipolar AMI RZ, and high-density bipolar (HDBn) RZ.

Table 2.1.3 Comparison of Bipolar Line Codes

Parameter	Bipolar AMI NRZ	Bipolar AMI RZ	HDBn RZ
Advantages	• No dc component. • Occupies less bandwidth than unipolar and polar NRZ schemes. • Suitable for transmission over ac coupled lines. • Possesses single error detection capability.	• No dc component. • Occupies less bandwidth than unipolar and polar RZ schemes. • Suitable for transmission over ac coupled lines. • Possesses single error detection capability. • Clock can be extracted by rectifying (a copy of) the received signal.	• No dc component. • Occupies less bandwidth than unipolar and polar RZ schemes. • Suitable for transmission over ac coupled lines. • Possesses single error detection capability. • Clock can be extracted by rectifying (a copy of) the received signal. • Is Transparent.
Disadvantages	• Does not possess any clocking component for ease of synchronization. • Is not transparent.	• Is not transparent.	

Figures 2.1.30, 2.1.31, and 2.1.32 show PSD waveforms of bipolar AMI NRZ, bipolar AMI RZ, and High-Density Bipolar (HDBn) RZ line codes, respectively, for comparison.

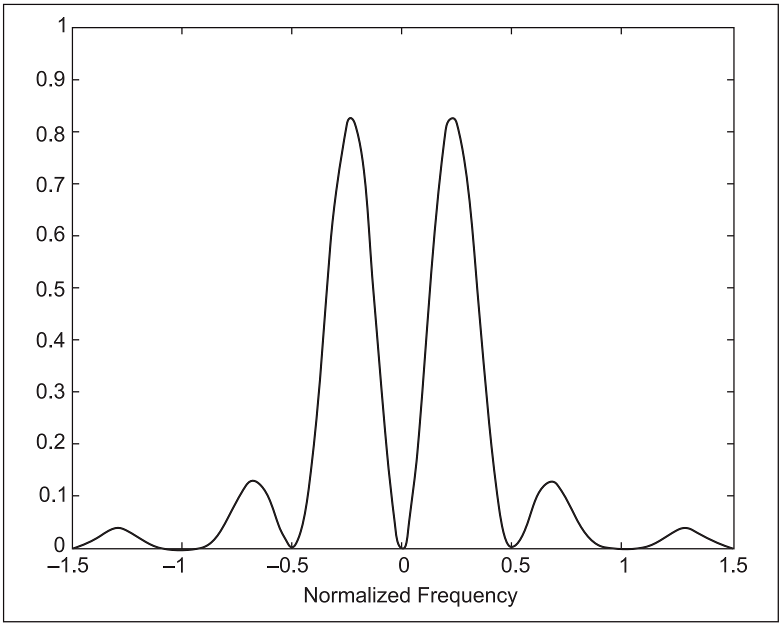


Figure 2.1.30 PSD of Bipolar AMI NRZ Line Code

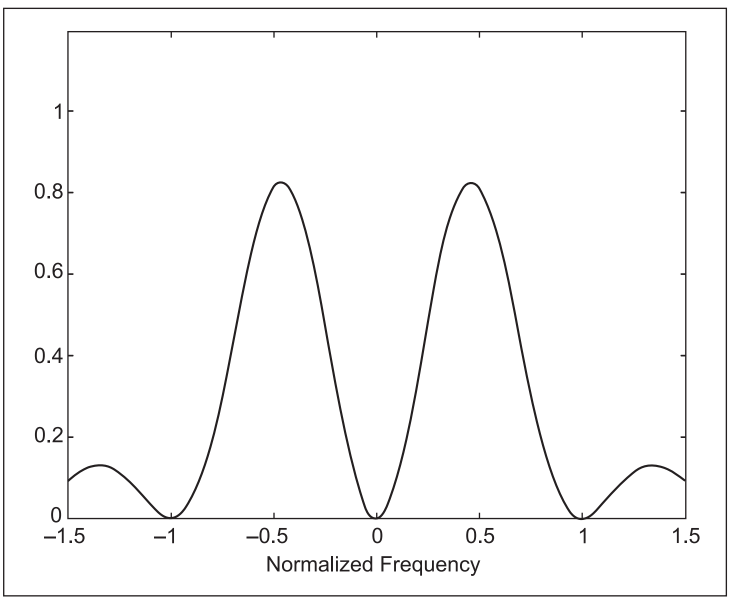


Figure 2.1.31 PSD of Bipolar AMI RZ Line Code

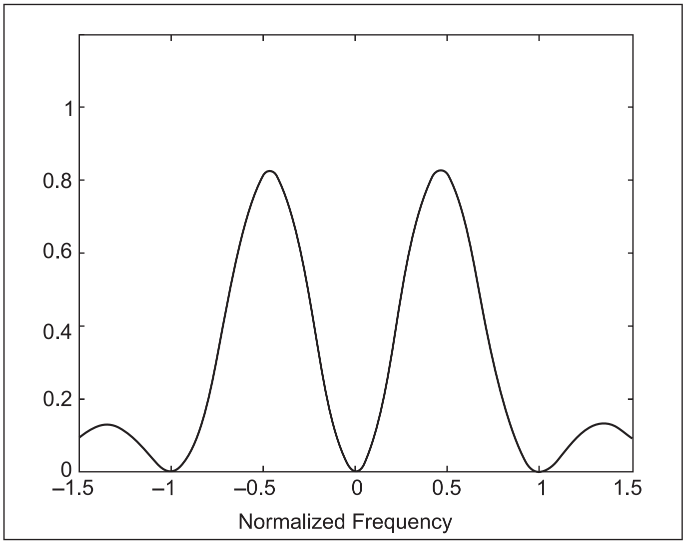


Figure 2.1.32 PSD of High Density Bipolar (HDB3) RZ Line Code

Note *The PSD of HDB_3 RZ is similar to the PSD of bipolar RZ. The characteristics of HDBn RZ line code make it ideal for use in Wide Area Networks.*

Table 2.1.4 gives the comparison of desirable properties in various line-coding techniques.

Table 2.1.4 Comparison of Desirable Properties in Line Coding Techniques

S. No.	Property	Polar NRZ	Polar RZ	Manchester	BP-AMI-RZ
1.	Transmission of dc component	Yes	Yes	No	No
2.	Signaling rate	$1/T_b$	$1/ T_b$	$1/T_b$	$1/T_b$
3.	Noise immunity	Low	Low	High	High
4.	Synchronizing capability	Poor	Poor	Very good	Very good
5.	Bandwidth required	$1/(2T_b)$	$1/T_b$	$1/(2T_b)$	$1/T_b$
6.	Crosstalk	High	High	Low	Low
7.	Power spectral density (PSD)	Mostly in main lobe extending up to $1/T_b$	----	The whole power lies within $1/T_b$	Mostly power lies up to $2/T_b$

Table 2.1.5 shows the comparison of various attributes of most common unipolar and bipolar line-coding techniques.

Table 2.1.5 Comparison of Attributes of Line-Coding Techniques

S. No.	Line-Coding Technique	Average dc Level	Minimum Bandwidth Required	Clock Recovery Capability	Error-Detection Capability
1.	UP-NRZ	+V/2	≈ ½ × bit rate	Not Good	No
2.	UP-RZ	+V/4	≈ bit rate	Good	No
3.	BP-NRZ	0 V	≈ ½ × bit rate	Not Good	No
4.	BP-RZ	0 V	≈ bit rate	Very Good	No
5.	BP-RZ-AMI	0 V	≈ ½ × bit rate	Good	Yes
6.	Manchester	0 V	≈ ½ × bit rate	Best	No

2.1.5 Application Areas of Line-Coding Techniques

Choice of Line Code

Binary data can be transmitted using a number of different types of pulses. The choice of a particular pair of pulses to represent the symbols 1 and 0 is called line coding and the choice is generally made on the grounds of one or more of the following considerations:

- Presence or absence of a dc level.
- Power spectral density—particularly its value at 0 Hz.
- Bandwidth.
- Transparency (i.e., the property that any arbitrary symbol, or bit, pattern can be transmitted and received).
- Ease of clock signal recovery for symbol synchronization.
- Presence or absence of inherent error detection properties.
- BER performance.

Table 2.1.6 Application Areas of Line-Coding Techniques

S. No.	Line-Coding Technique	Application Areas
1.	Unipolar NRZ	Internal computer waveforms
2.	Unipolar RZ	Baseband data transmission, Magnetic tape recording
3.	Polar NRZ-L	Digital logic circuits, Magnetic tape recording
4.	Bipolar RZ	Satellite telemetry communication links, Optical communications, Magnetic tape-recording systems
5.	Manchester	IEEE 802.3 standard for Ethernet local area network (LAN)
6.	Bipolar RZ-AMI	Telephone systems as signaling scheme, T-digital carrier systems
7.	B8ZS	T1 digital carrier system
8.	B6ZS	T2 digital carrier system
9.	B3ZS	T3 digital carrier system

SOLVED EXAMPLE 2.1.28 Various Line Coding Waveforms

Draw the line-coding waveforms for the binary data sequence 1 1 0 0 0 0 1 0 using bipolar RZ, polar NRZ-L, and bipolar NRZ-AMI types of line-coding techniques.

Solution Figure 2.1.33 illustrates the line-coding waveforms for the given binary data 1 1 0 0 0 0 1 0.

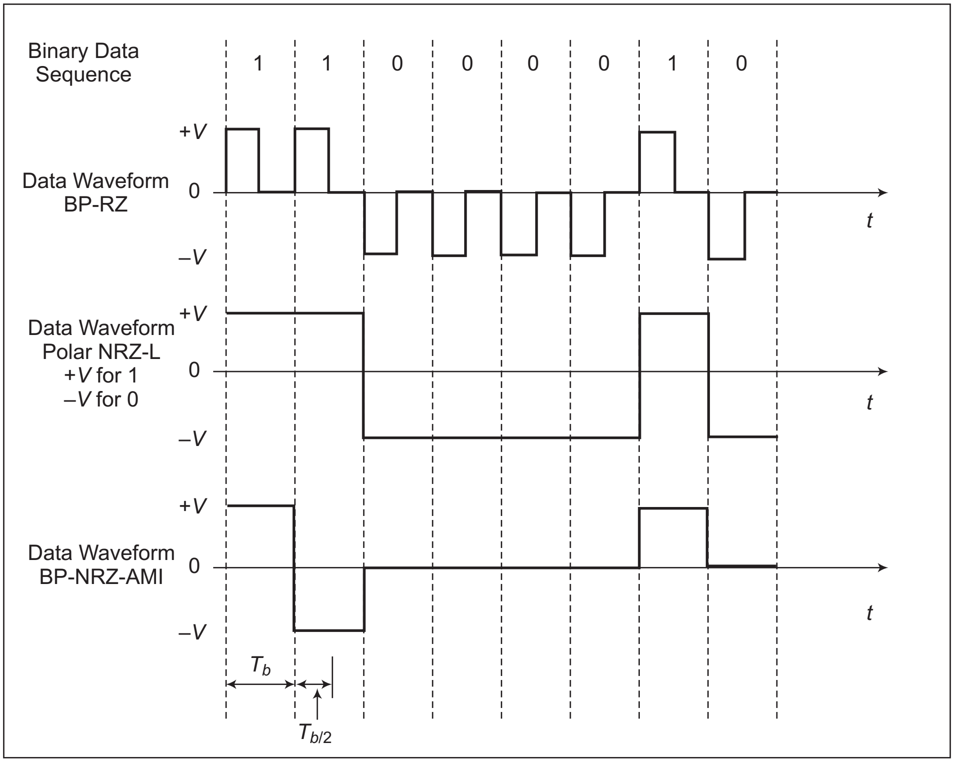


Figure 2.1.33 Line-Coding Waveforms for Example 2.1.28

SOLVED EXAMPLE 2.1.29 Various Line-Coding Waveforms

The binary data 1 0 1 1 0 0 0 1 is transmitted over a baseband channel. Draw the line-coding waveforms using unipolar NRZ, bipolar RZ, and BP-RZ-AMI.

Solution Figure 2.1.34 illustrates line-coding waveforms for the given binary data 1 0 1 1 0 0 0 1.

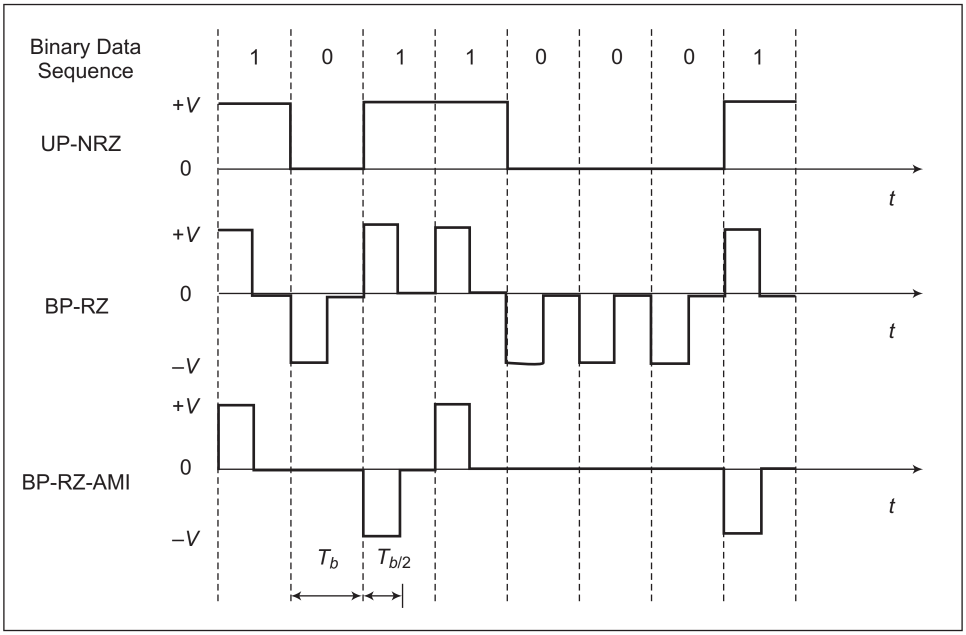


Figure 2.1.34 Line-Coding Waveforms for Example 2.1.29

SOLVED EXAMPLE 2.1.30 Various Line-Coding Waveforms

Consider the binary data sequence 1 1 0 0 0 0 1. Draw the waveforms for the following line-coding techniques:

(a) Bipolar NRZ-AMI (alternate mark inversion)
(b) Split phase or Manchester
(c) Unipolar RZ

Solution Figure 2.1.35 illustrates the line-coding waveforms for the given binary data 1 1 0 0 0 0 1.

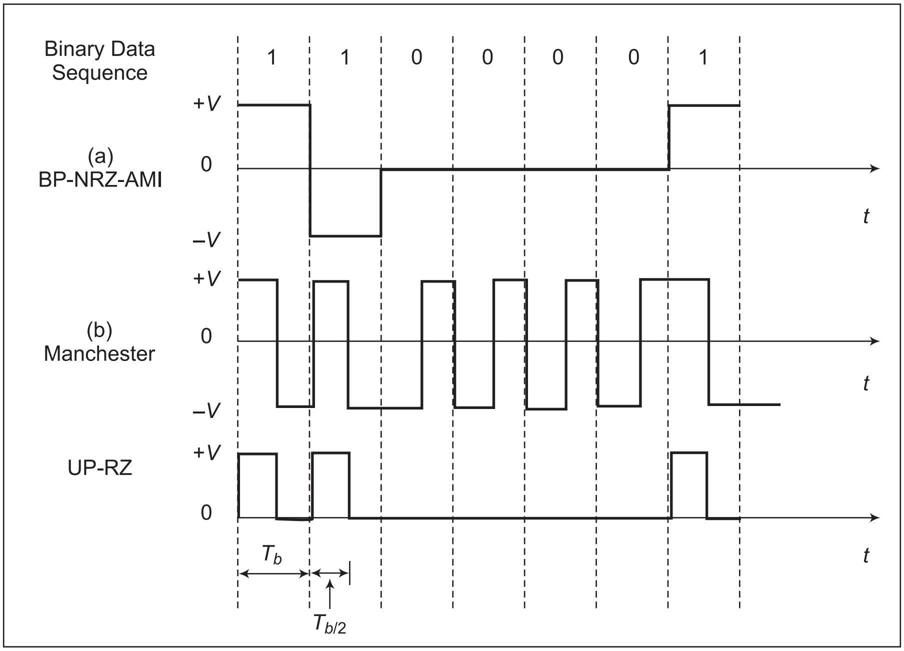


Figure 2.1.35 Line-Coding Waveforms for Example 2.1.30

MATLAB simulation exercises on generation of various types of line codes,

Scan the QR code given here OR visit: http://qrcode.flipick.com/index.php/100

Self-Assessment Exercise linked to LO 2.1

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/101

Q2.1.1 Distinguish between (a) a data element and a signal element, and (b) data rate and signal rate.

Q2.1.2 A signal has two data levels with a pulse duration of 1 ms. Calculate the pulse rate and bit rate. If a signal has four data levels with the same pulse duration, then show that the pulse rate remains same whereas the bit rate doubles.

Q2.1.3 Show that dc component or baseline wandering affects digital transmission.

Q2.1.4 Define the characteristics of a self-synchronizing signal. Evaluate how lack of synchronization between transmitter and receiver clocks affects the performance of digital transmission.

Q2.1.5 Given the binary data sequence 1 1 1 0 0 1 0. Represent it using the line codes: (a) Unipolar RZ, and (b) Bipolar RZ-AMI.

Q2.1.6 Given the binary data sequence 1 1 1 0 0 1 0, outline the transmitted sequence of rectangular pulses using the line coding techniques or digital data signaling formats as (a) unipolar RZ, (b) polar NRZ, (c) polar RZ, (d) Manchester, and (e) bipolar RZ-AMI. ○●●

Q2.1.7 In a digital transmission, the sender clock is 0.3 percent faster than the receiver clock. Calculate the data rate at which the receiver receives the data if the transmitted data rate is 1 Mbps. ○●●

Q2.2.8 A digital transmission system is using NRZ line-coding technique to transfer 10 Mbps data. Check that the average signal rate is 500 kbaud and the minimum bandwidth for the calculated baud rate is 500 kHz. ●●●

Note ○○● Level 1 and Level 2 Category
○●● Level 3 and Level 4 Category
●●● Level 5 and Level 6 Category

If you have been able to solve the above exercises then you have successfully mastered

LO 2.1: *Know how to convert digital data to digital signal using various line codes for reliable baseband transmission.*

LO 2.2

2.2 INTERSYMBOL INTERFERENCE (ISI)

What We Discuss Here ....

When line-coded rectangular waveform is transmitted over a bandlimited channel, spectral distortion occurs due to suppression of a small part of the spectrum. This results into spreading of the pulse, known as *pulse dispersion*. Obviously, spreading of a pulse beyond its allotted time period T_b (pulse width) will tend to interfere with adjacent pulses. This is known as *Intersymbol Interference (ISI)*.

What is InterSymbol Interference (ISI)?

In digital baseband signal transmission, ISI arises due to the dispersive nature of a communications channel. ISI is caused by non-ideal channels that are not distortionless over the entire signal bandwidth (since the transmitted signal bandwidth is more than the available channel bandwidth). ISI is not equivalent to channel noise but it is due to channel distortion. The ISI occurs due to the imperfections in the overall frequency response of the digital communication system. When a pulse of short width is transmitted through a band-limited communication system then the frequency components contained in the input pulse are differentially attenuated as well as delayed by the system. The pulse appearing at the output of the system will be dispersed over an interval which is longer than the transmitted pulse of short width.

2.2.1 Effects of ISI

Figure 2.2.1 shows a typical dispersed pulse due to ISI.

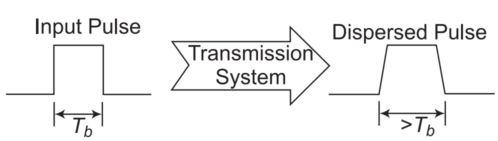


Figure 2.2.1 Dispersed Pulse due to ISI

It is seen that due to pulse dispersion, each symbol having short duration will interfere with each other when transmitted over the communication channel. This will result in ISI, as shown in Figure 2.2.2.

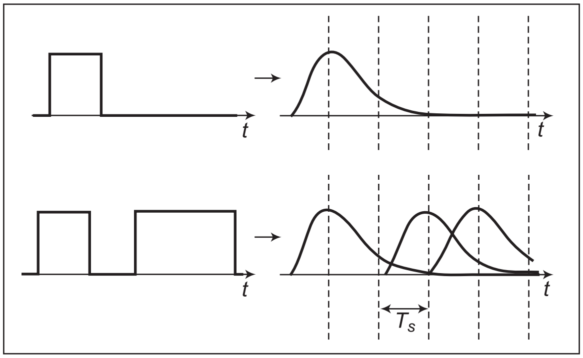


Figure 2.2.2 Effect of ISI on transmission of pulses

We know that in the absence of ISI and channel noise, the transmitted symbol can be decoded correctly at the receiver. However, due to occurrence of ISI, an error will be introduced in the decision-making device at the receiver output.[10]

In a bandlimited PCM channel, the received pulse waveforms may be distorted due to ISI and may extend to the next time slot (termed *interchannel interference*). If ISI is large enough then it can cause errors in detection of the received pulse at the regenerative repeater or the receiver of baseband digital communication system such as PCM.

Mathematical Expression of ISI

Let us consider a mathematical model of a **baseband binary data transmission system,** as shown in Figure 2.2.3.

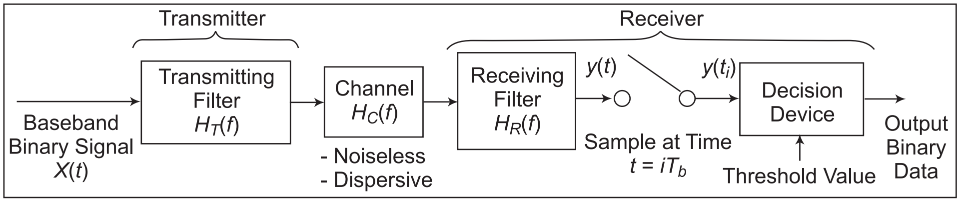


Figure 2.2.3 Baseband Binary Data Transmission System Model

Step I Let the waveform of the k^{th} pulse in the baseband-modulated binary signal be given as

$$X_k(t) = A_k v\,(t - kT_b)$$

where A_k is the coefficient, which depends on the input data and the type of line code used; $v(t)$ denotes the basic pulse waveform normalized such that $v(0) = 1$, and T_b is the pulse duration in seconds.

[10] When digital pulses from more than one source are multiplexed together, the amplitude, frequency, and phase responses become even more critical. For example, in Time-Division Multiplexed (TDM) systems, adjacent pulses from different sources may interfere with each other due to ISI, thereby resulting into ***crosstalk***. In TDM system, adjacent pulses could be from other conversations, so the conversations can interact with each other (thus the name ***crosstalk***) due to ISI.

Step II This signal passes through a transmitting filter of transfer function, $H_T(f)$ and presented to the channel.

Assume The channel is noiseless, but dispersive in nature.

Step III The output is distorted as a result of transmission through the bandlimited channel (represented as a bandpass filter) of transfer function, $H_C(f)$.

Step IV The channel output is passed through a receiving filter of transfer function, $H_R(f)$.

Step V This filter output is sampled synchronously with that of transmitter.

The output of the receiving filter, $y(t)$ may be expressed as

$$y(t) = \mu \sum_{k=-\infty}^{\infty} A_k v(t - kT_b)$$

where μ is a scaling factor.

The receiving filter output, sampled at time $t_i = iT_b$ (with i can assume integer values), yields

$$y(t_i) = \mu \sum_{k=-\infty}^{\infty} A_k v(iT_b - kT_b)$$

$$\Rightarrow \qquad y(t_i) = \mu \sum_{k=i} A_k v(iT_b - kT_b) + \mu \sum_{\substack{k=-\infty \\ k \neq i}}^{\infty} A_k v(iT_b - kT_b)$$

$$\Rightarrow \qquad y(t_i) = \mu A_i + \mu \sum_{\substack{k=-\infty \\ k \neq i}}^{\infty} A_k v(iT_b - kT_b)$$

In this expression, we note that

- the first term is produced by the transmitted pulse itself; and
- the second term represents the residual effect of all other transmitted pulses on the decoding of the i^{th} bit.

This residual effect is known as ***Intersymbol Interference (ISI)***.

Step VI The sequence of samples is used to reconstruct the original data sequence by means of a decision device depending upon the preset threshold value.

If there is no ISI at the decision-making instants in the receiver despite pulse spreading or overlapping, pulse amplitudes can still be detected correctly. So it is utmost necessary either to eliminate or at least minimize ISI before it happens! **IMPORTANT!**

SOLVED EXAMPLE 2.2.1 **Effect of Intersymbol Interference**

A speech waveform is sampled at the rate of 10,000 samples per second to produce a discrete time analog signal. The width of each sampling pulse is 0.01 ms. The resultant signal is transmitted through a communication channel which can be approximated by a low-pass filter with cutoff frequency at 100 kHz. How does the channel affect the transmitted pulse?

Solution Corresponding to the given sampling rate of 10,000 samples per second, it can be assumed that the maximum frequency of the input speech waveform has a maximum frequency less than 5 kHz in order to avoid aliasing errors.

Width of the transmitted pulse signal = 0.01 ms (Given)

This means that the cut-off frequency of low-pass filter representing communication channel will be 1/0.01ms = 100 kHz.

- When the transmitted pulse forms the input to this channel, it may be confined to its assigned time interval.
- But the filtering effects of the channel may widen the pulse to overlap adjacent intervals.
- The overlap from one time slot to adjacent time slots is intersymbol interference.

This may result into ***crosstalk*** (in TDM systems).

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.2.2 A speech waveform is sampled at the rate of 1000 samples per second to produce a discrete-time analog signal. The width of each sampling pulse is 0.1 ms. The resultant signal is transmitted through a communication channel which can be approximated by a low-pass filter with cut-off frequency at 10 kHz. How does the channel affect the transmitted pulse?

2.2.2 Eye Diagram

Recall

In the previous subsection, we learnt that when a pulse of short duration is transmitted through a bandlimited digital communication system then the frequency components contained in the transmitted pulse are differentially attenuated as well as delayed by the system. The pulse appearing at the output of the system will be dispersed over an interval which is longer than that of the transmitted pulse.

What is an Eye Diagram?

The eye diagram is a very effective tool for digital signal analysis during real-time experiments. An eye diagram, also known as an eye pattern, is a practical technique for determining the severity of the degradations introduced by intersymbol interference and channel noise into the line-encoded digital pulses in baseband transmission.

Thus, an eye diagram is a simple and convenient engineering tool applied on received signals for studying the effects of

- Intersymbol interference
- Accuracy of timing extraction
- Noise immunity
- Determining the bit-error-rate (BER)

Its Significance

An eye diagram provides information about the state of the channel and the quality of the received pulse. This information is useful for faithful detection of received digital signal and determination of overall performance of digital communication systems.

How to Generate an Eye Diagram?

An analog oscilloscope is generally used to plot an eye diagram on its display.

- The received pulse input which may be dispersed in time due to channel noise and ISI, is given to the vertical input of the oscilloscope.
- A sawtooth type time base generator is provided to its horizontal input. It has the same time period as the incoming data, that is, sweep rate is nearly same as the symbol rate.
- The symbol clock is applied to the external trigger input.
- At the end of the fixed time interval, the signal is wrapped around to the beginning of the time axis.

Thus, an eye diagram consists of many overlapping curves.

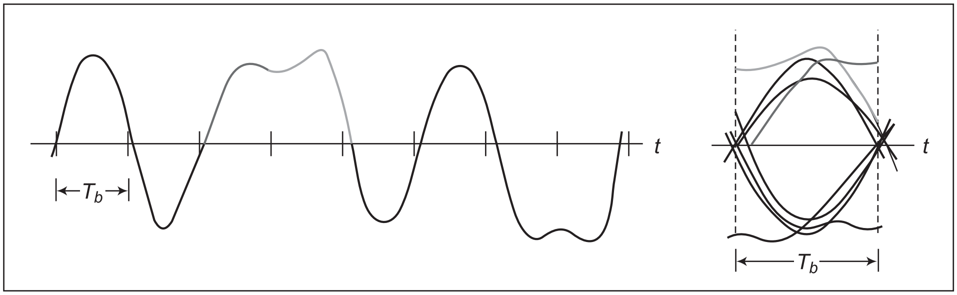


Figure 2.2.4 The Concept of Eye Diagram

Figure 2.2.5 shows a typical picture of eye diagram with many overlapping curves.

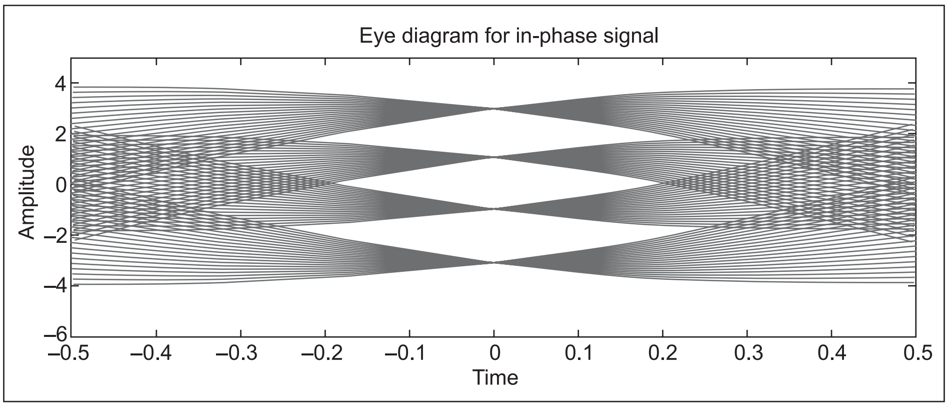


Figure 2.2.5 A Typical Eye Diagram

Inference from Eye Diagram

When the eye opening is quite wide in the eye diagram, it pertains to perfect eye pattern for noise-free, bandwidth-limited transmission of an alphabet of two digital waveforms encoding a binary signal (0s and 1s). The open part of the signal represents the time that we can safely sample the signal with fidelity. From the typical eye diagram, we can extract several key measures regarding the signal quality. These may include the following:

Noise Margin Noise margin is the amount by which the received signal level exceeds the specified threshold level for a proper logic '0' or '1'. In case of an eye diagram, noise margin or the vertical eye opening is related to the signal-to-noise ratio (S/N), and thus the BER. For example, a large eye opening corresponds to a low BER.

Sensitivity to Timing Jitter Timing jitter refers to short-term variations of a digital signal's significant instants from their ideal positions in time. In case of an eye diagram, the horizontal eye opening relates the jitter and the sensitivity of the sampling instant to jitter. Slope indicates sensitivity to timing error, smaller is the slope, better it is.

Level-crossing Timing Jitter It refers to the deviation from the ideal timing of detection of received pulse and the threshold level. The best sampling time occurs at the center of the eye. Jitter can cause bit errors by shifting the bit sampling instant away from the optimum position and into regions of the bit time that are close to (or beyond) the bit transition points (at the rising and falling edges of the pulse).

An eye diagram is used for finding the best decision point where the eye is most widely opened. That is, the following:

- When the received bit pattern is ideal (free of any errors), the oscilloscope display will show two vertical lines across its display.
- If the bit sequence is slightly distorted, the oscilloscope display shows the pattern which is very similar to the human eye, the central portion of the pattern representing the opening of the eye.
- If the received digital signal is further distorted, the eye appears to have closed.

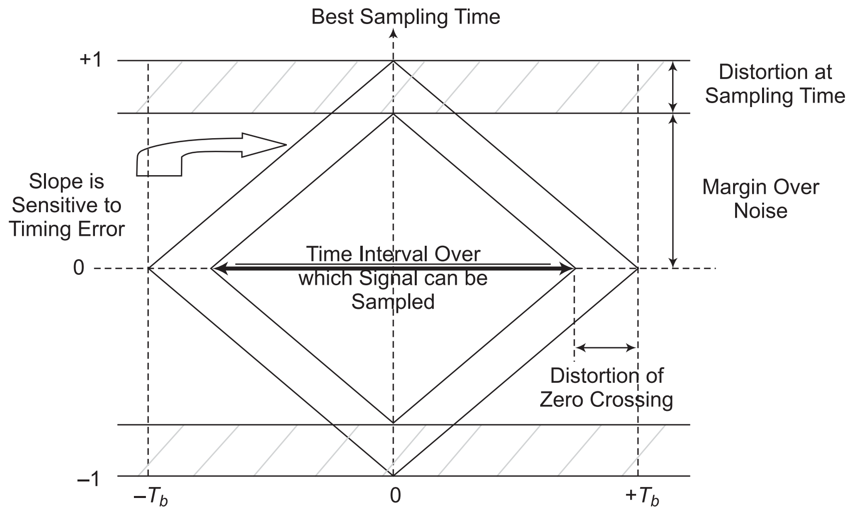


Figure 2.2.6 Monitoring Transmission Quality by An Eye Diagram

From this figure, we observe the following:

- The vertical lines labeled +1, 0, and –1 correspond to the ideal received amplitude levels.
- The horizontal lines, separated by the signaling interval, T_b, correspond to the ideal decision times.
- The eye opening is the area in the middle of the eye diagram.

As intersymbol interference increases, the eye opening reduces. If the eye is closed completely, it is next to impossible to avoid errors. The effect of pulse degradation is a reduction in the size of the ideal eye.

IMPORTANT!

In fact, ISI can be represented as the ratio of ideal vertical opening per unit length to degraded vertical opening in the same unit length in the eye diagram.

Benefits of Eye Diagram

- Eye diagram is a means of evaluating the quality of a received digital waveform. By quality we mean "the ability to correctly recover symbols as well as timing". The received signal could be examined at the input of a digital receiver or at some stage within the receiver before the decision stage.
- Eye diagram reveals two major issues: noise margin (sample value variations), and jitter and sensitivity of sampling instant.

- Eye diagrams reveal the impact of ISI and noise.
- Eye diagram can also give an estimate of achievable BER.

MATLAB simulation exercise on Eye Diagram,
Scan the QR code given here OR visit: http://qrcode.flipick.com/index.php/102

Self-Assessment Exercise linked to LO 2.2

Q2.2.1 Discuss different ways to reduce the effects of intersymbol interference.
Q2.2.2 Analyze the quality of digital transmission using the concept of an eye diagram.
Q2.2.3 Monitor the effects of intersymbol interference on an analog oscilloscope for transmission of binary sequence at a faster data rate.
Q2.2.4 Create an example of occurrence of ISI in an 8-bit PCM signal.
Q2.2.5 Differentiate between occurrences of crosstalk in PAM systems and intersymbol interference in PCM systems.

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/105

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 2.2: *Discuss intersymbol interference (ISI) problem in digital transmission.*

2.3 NYQUIST PULSE SHAPING AND EQUALIZATION

LO 2.3

Recall

The PSD of a baseband digital modulated signal (line-coded signal) varies for different line codes. It can be controlled by selecting a proper line code. So far we have considered a rectangular-shaped pulse waveform having either full-width or half-width bit duration to represent the digital data in any line code. In the case of half-width rectangular pulse, the signal bandwidth is, strictly speaking, infinity. Accordingly, the bandwidth of PSD of a rectangular pulse is also infinite although the essential bandwidth was finite. We have observed that most of the power of a bipolar line-coded signal is contained within the specified essential bandwidth up to R_b Hz from 0, but small amount of PSD is still available beyond $f > R_b$ Hz.

Why Pulse Shaping?

In digital baseband signaling, pulse shape is chosen to satisfy the following requirements:

- To yield maximum S/N at the time instance of decision
- To accommodate signal to channel bandwidth that means rapid decrease of pulse energy outside the main lobe in frequency domain improving filter design and lowering cross-talk in multiplexed systems

What We Discuss Here ....

The PSD of a particular line code can be controlled by proper shaping of the rectangular pulse waveform itself. Nyquist proposed different criteria for pulse shaping by which ISI is either eliminated (zero ISI) or controlled even after the pulses are allowed to overlap, known as *Nyquist pulse shaping*.

Equalization is a special technique that helps the demodulator at the baseband receiver to recover a rectangular pulse with the best possible signal-to-noise ratio, free of any intersymbol interference.

Before exploring the solution to the problem caused by ISI, let us first try to analyze this phenomenon. The system requirement is that

"we need to transmit a sequence of time-limited pulses (after baseband modulation or line coding), each of width T_b over a channel having finite bandwidth, and we must be able to detect the pulse amplitude correctly without ISI."

2.3.1 Nyquist Criterion for Zero ISI

Recap

If the ISI is not eliminated or at least the amount of ISI is not adequately controlled then its presence may introduce errors in the decision device at the receiver of baseband digital communication system.

What Can be Done?

The impact of ISI can be minimized by using a properly shaped bandlimited pulse prior to transmission.

Recalling from the previous subsection, ISI is represented by the term $\mu \sum_{\substack{k=-\infty \\ k \neq i}}^{\infty} A_k v(iT_b - kT_b)$.

The receiver extracts the received data sequence by sampling it at the instant $t_i = iT_b$. It implies that the contribution from the weighted pulse $A_k v(t - kT_b)$ for $k = i$ should be free from ISI due to all other weighted pulse contributions represented by $k \neq i$. That is,

$$A_k v(t - kT_b) = \begin{cases} 1; & i = k \\ 0; & i \neq k \end{cases}$$

IMPORTANT!

In fact, $y(t_i) = \mu A_i$ implies zero ISI. Then the received sequence is without any error in the absence of channel noise. This is the Nyquist criterion for distortionless baseband binary signal transmission.

Nyquist proposed different criteria for pulse shaping by which ISI is either eliminated (zero ISI) or controlled even after the pulses are allowed to overlap.

Nyquist's First Criterion

The simplest method (known as Nyquist's first criterion for zero ISI) is to choose a pulse shape that has a nonzero amplitude at its center (usually unity at $t = 0$) and zero amplitude at $t = \pm\, nT_b$ for $n = 1, 2, 3, ...$ where T_b is the pulse duration or the separation between successive pulses ($T_b = 1/R_b$).

A pulse satisfying this criterion causes zero ISI because the samples at $t = 0, T_b, 2T_b, 3T_b$ .... consists of the amplitude of only one pulse (centered at the sampling instant) and zero amplitude at the centers of all the remaining pulses, thereby no interference from all other pulses.

We know that transmission of R_b bits per second requires at least $R_b/2$ Hz bandwidth. Incidentally, there exists one and only one pulse which meets Nyquist's first criterion for zero ISI which also has the minimum bandwidth $R_b/2$ Hz.

IMPORTANT!

This pulse waveform is of the type given by sinc $(x) = [(\sin x)/x]$ pulse instead of a rectangular-shaped pulse waveform, known as *Nyquist pulse shaping*. The pulse signal given as $v(t)$ = sinc $(\pi R_b t)$ has the required property.

The Fourier transform of this pulse can be expressed as

$$V(f) = \frac{1}{2B_0}\text{rect}\left(\frac{f}{2B_0}\right)$$

where $B_0 = \frac{R_b}{2}$ denotes the bandwidth which is equal to one-half of the pulse rate R_b.

This shows that we can transmit this pulse at a rate of R_b pulses per second without ISI, over a bandwidth $B_0 = \frac{R_b}{2}$ only. The pulse signal $v(t)$ can be regarded as the impulse response of an ideal low-pass filter with passband amplitude response $\frac{1}{2B_0}$ and bandwidth B_0, and is given by

$$v(t) = \frac{1}{2B_0}\text{rect}\left(\frac{f}{2B_0}\right)$$

Figure 2.3.1 shows an ideal amplitude response of the basic pulse shape and its plot.

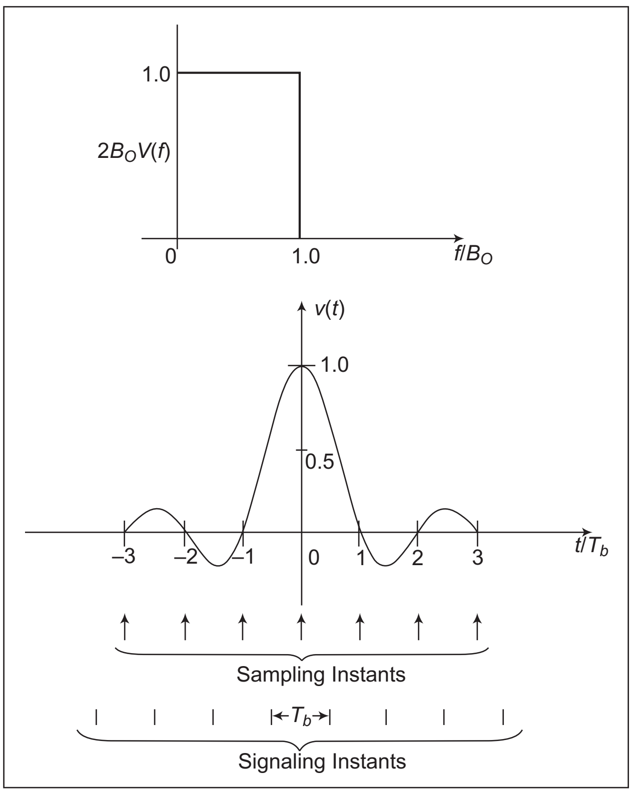


Figure 2.3.1 An Ideal Rectangular Pulse

The following can be seen:

- The waveform $v(t)$ has its peak value at the origin and goes through zero at $\pm nT_b$ where $n = 0, \pm 1, \pm 2, ...$
- The waveform $v(t)$ is optimized in bandwidth and offers the best solution for zero ISI with minimum bandwidth.
- However, there are certain practical difficulties such as it requires that the amplitude characteristics $V(f)$ be flat from $-B_0$ to B_0, and zero elsewhere.
- This is physically not realizable and impractical because it starts at $-\infty$ and requires abrupt transitions at $\pm B_0$ (flat response of a filter) which is not possible.

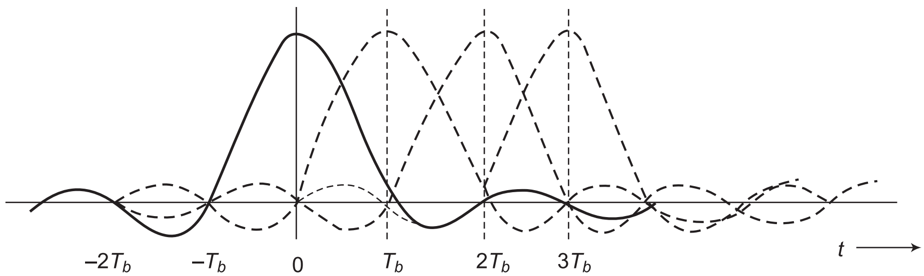


Figure 2.3.2 Minimum Bandwidth Pulse that Satisfies Nyquist's First Criterion

The function $v(t)$ decreases with $\dfrac{1}{|t|}$ for large $|t|$, resulting in a slow rate of decay. This is caused by discontinuity of $v(f)$ at $\pm B_0$. This too leads to some other serious practical difficulties.

For instance, the following:

- If the required nominal pulse rate of R_b bits/s is slightly deviated, the pulse amplitudes will not be zero at the other pulse centers.
- Moreover, the cumulative interference at any pulse center from all the remaining pulses will be very large.
- A similar result will be obtained if the sampling instants at the receiver deviate a little because of pulse time jitter.

So all these practical problems exist due to the fact that the pulse $v(t) = \sin c(\pi R_b t)$ decays too slowly as $1/t$. The solution is to apply a pulse that decays faster than $1/t$. Nyquist has shown that such a pulse requires a bandwidth slightly greater than $R_b/2$, that is $kR_b/2$ where $1 \le k \le 2$.

It is emphasized that the pulses arriving at the detector input of the receiver need to meet the desired Nyquist criterion. Hence, the transmitted pulses should be so shaped that after passing through the channel, they are received in the desired Nyquist form. However, in practice, pulses need not be shaped rigidly at the transmitter. The final pulse shaping can be carried out by an equalizer at the receiver.

Nyquist's Second Criterion

We have seen that the practical Nyquist pulse requires baseband transmission bandwidth slightly larger than the theoretical minimum bandwidth. In order to reduce the transmitted pulse bandwidth, the pulse width itself needs to be increased. We know that pulse broadening in the time domain leads to reduction in its bandwidth. This is *Nyquist's second criterion*.

However, widening the pulse means interference with the neighbouring pulses (ISI). Since there are only two symbols (1 and 0) in the binary transmission, a known and controlled amount of ISI may be allowed. The ISI is controlled in the same sense that it comes only from the succeeding symbol. For this, we use polar line codes, i.e., symbol 1 is transmitted by $+V$ and symbol 0 is transmitted by $-V$. The received signal is sampled at $t = kT_b$, and the pulse has zero value at all k except for $k = 0$ and 1, where its value is $+V$ or $-V$. It is obvious that such a pulse causes zero ISI with all the remaining pulses except the succeeding pulse.

The technique of introducing controlled ISI is known as *duo-binary signaling* or *correlative coding* or *partial-response signaling*, and the pulse used is known as *duo-binary pulse*.

IMPORTANT!

Thus, by adding ISI to the transmitted pulse signal in a controlled manner, it is possible to achieve the Nyquist signaling rate or Nyquist bandwidth. The effect of known ISI in the transmitted pulse signal can be easily interpreted at the receiver in a deterministic manner. *How?* Consider two such successive pulses located at $t = 0$ and $t = T_b$, respectively. There are three possible values only.

- First, if both pulses were positive, the sample value of the resulting signal at $t = T_b$ would be +2 V. As per the defined decision rule, the present bit is 1 and the previous bit is also 1 because the sample value is positive.
- Second, if both pulses were negative, the sample value of the resulting signal at $t = T_b$ would be –2 V.

So as per the decision rule, the present bit as well as and the previous bit is 0 because the sample value is negative.

- Third, if one pulse is positive and the second pulse is negative then the sample value of the resulting signal at $t = T_b$ would be zero.

In such a case, the present bit is the opposite of the previous bit. Then knowledge of the previous bit allows us to determine the present bit.[11]

The correlative-level coding may involve the use of tapped-delay-line filters with different tap-weights. The detection procedure followed at the receiver uses a stored estimate of the previous symbol, called decision feedback technique. It is essentially an inverse of the operation of the simple delay-line filter employed at the transmitter.

2.3.2 Raised-Cosine Pulse Shaping

An Alternate Solution to Minimize ISI

Use the *raised-cosine filter* whose transfer function can be expressed as

$$H(f) = \begin{cases} 1 & ;\ \text{for } |f| < (2B_0 - B) \\ \cos^2\left(\dfrac{\pi}{4}\dfrac{|f| - (2B_0 - B)}{B - B_0}\right) & ;\ \text{for } (2B_0 - B) < |f| < B \\ 0 & ;\ \text{for } |f| > B \end{cases}$$

where B is the absolute bandwidth; $B_0 = R_b/2$ represents the theoretical minimum Nyquist bandwidth for the rectangular pulse ($R_b = 1/T_b$; T_b being the pulse width) and the –6 dB

[11] ***Duo-binary baseband signaling*** or ***partial-response signaling*** utilizes controlled amounts of ISI for transmitting pulses at a rate of $\boldsymbol{R_b}$ bits per second over a communication channel having a bandwidth of $\boldsymbol{R_b}/2$ Hz. This means that the dc null occurs at $\boldsymbol{f} = 1/2\boldsymbol{T_b}$ and the essential bandwidth is at the minimum transmission bandwidth needed for a data rate of $\boldsymbol{R_b}$. Moreover, it provides an additional mean to reshape the PSD.

bandwidth for the raised cosine filter; $(B - B_0)$ is the excess bandwidth beyond the minimum Nyquist bandwidth.

Let us define the ***roll-off factor*** of the raised-cosine filter, α as the ratio of excess bandwidth $(B - B_0)$ to the -6 dB bandwidth B_0. That is,

$$\alpha = \frac{(B - B_0)}{B_0} = \frac{(B - B_0)}{R_b/2} = 2T_b(B - B_0)$$

Since $(B - B_0)$ cannot be larger than $R_b/2$, therefore, $0 \le \alpha \le 1$.
In terms of frequency f,

- the theoretical minimum bandwidth is $R_b/2$, and
- the excess bandwidth is $(B - B_0) = \alpha R_b/2$ Hz.

Therefore, the bandwidth of the pulse signal $V(f)$ is given as

$$B_T = \frac{R_b}{2} + \frac{\alpha R_b}{2} = \frac{(1+\alpha)R_b}{2}$$

Significance of Roll-off Factor

For a given B_0, the roll-off factor α specifies the required excess bandwidth as a fraction of B_0 and characterizes the steepness of the filter roll-off. It can be seen that

- The $\alpha = 0$ is the case of Nyquist minimum bandwidth
- The $\alpha = 1$ is the case of 100% excess bandwidth required

If $V(f)$ is a Nyquist first criterion spectrum with a bandwidth that is 50% more than the Nyquist bandwidth, its roll-off factor $\alpha = 0.5$ or 50%. Because $0 \le \alpha \le 1$, the pulse transmission varies from $2B_T$ to B_T, depending on the choice of roll-off factor α.

- A smaller value of α gives a higher signaling rate. But the resulting pulse decays slowly as in the case of rectangular pulse.
- For the raised-cosine filter having $\alpha = 1$ and $B_T = R_b$, we achieve half the theoretical maximum rate. But the pulse spectrum decays faster (as $1/t^3$) and is less vulnerable to ISI.[12]

Figure 2.3.3 shows the normalized frequency response, $2B_0V(f)$ of raised-cosine filter for various values of roll-off factor, $\alpha = 0, 0.5, 1$.

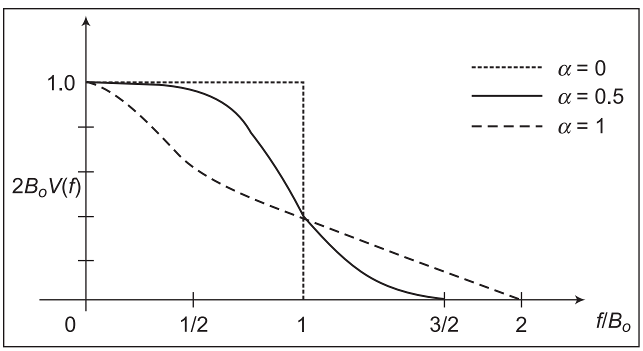


Figure 2.3.3 Nyquist Pulse and Raised-cosine Spectrum

It is observed that the bandwidth of this pulse is R_b Hz and has a value R_b at $t = 0$ and is zero at all the remaining signaling instants as well as at points midway between all the signaling

[12] The amount of ISI resulting from the pulse overlapping decreases as α increases from 0 to 1. Thus, a pulse-shaping filter should provide the desired roll-off. Its impulse response must be truncated to a finite duration so that it can be physically realizable.

instants. For $\alpha = 0.5$ or 1, the roll-off characteristics of $2B_0V(f)$ cut-off gradually as compared with an ideal low-pass filter ($\alpha = 0$), and is, therefore, easier to realize in practice.

SOLVED EXAMPLE 2.3.1 **16-level PAM System without ISI**

Determine symbol rate and the theoretical minimum system bandwidth needed for a 10 Mbps data using 16-level PAM without ISI.

Solution We know that the symbol rate, $R_s = \dfrac{R_b}{\log_2 L}$; where R_b is the bit rate of data and L is the number of levels in PAM signal.

For specified 16-level PAM and given R_b = 10 Mbps, we have

Symbol rate, $R_s = \dfrac{10 \text{ Mbps}}{\log_2(16)} = \dfrac{10 \text{ Mbps}}{\log_2(2^4)} = \dfrac{10 \text{ Mbps}}{4} = 2.5 \text{ Msymbols/s}$ **Ans.**

Theoretical minimum system bandwidth is given as

$$\text{BW} = \frac{R_s}{2} = \frac{2.5 \text{ Msymbols/s}}{2} = 1.25 \text{ MHz}$$ **Ans.**

SOLVED EXAMPLE 2.3.2 **Pulse Roll-off Factor**

A digital communication system using 16-level PAM has symbol rate of 2.5 Msymbols per second. If the applicable system bandwidth is 1.375 MHz then what should be the value of roll-off factor (α) of the transmitted pulse shape for zero ISI.

Solution We know that the system bandwidth, $W = \dfrac{1}{2}(1+\alpha)R_s$; where R_s is the symbol rate.

For given system bandwidth W = 1.375 MHz and R_s = 2.5 Msymbols/s, we have

$$1.375 \text{ MHz} = \frac{1}{2}(1+\alpha) \times 2.5 \text{ Msymbols/s}$$

$$\Rightarrow \qquad \alpha = \left(2 \times 1.375 \times \frac{1}{2.5}\right) - 1 = 0.1$$ **Ans.**

SOLVED EXAMPLE 2.3.3 **Raised Cosine Roll-off Filter**

A baseband digital system uses 4-level PAM along with a raised cosine roll-off filter characteristics. The system has a frequency response of 3.2 kHz. If binary data is transmitted at 9600 bps data rate, then what would be the symbol rate and roll-off factor (α) of the transmitted pulse shape for zero ISI?

Solution We know that the symbol rate, $R_s = \dfrac{R_b}{\log_2 L}$; where R_b is the bit rate of data and L is the number of levels in PAM signal (or, $n = \log_2 L$ is the number of bits per symbol).

For specified 4-level PAM and given R_b = 9600 bps, we have

$$R_s = \frac{9600 \text{ bps}}{\log_2(4) \text{ bits/symbol}} = \frac{9600 \text{ bps}}{\log_2(2^2) \text{ bits/symbol}}$$

$$= \frac{9600 \text{ bps}}{2 \text{ bits/symbol}} = 4800 \text{ symbols/s}$$ **Ans.**

We know that the system bandwidth, $W = \frac{1}{2}(1+\alpha)R_s$; where α is the roll-off factor of raised-cosine filter used for pulse shaping.
It is given that the system has frequency response of 3.2 kHz. This means that the system bandwidth $W = 3.2$ kHz.

Therefore, $$3.2 \text{ kHz} = \frac{1}{2}(1+\alpha) \times 4800 \text{ symbols/s}$$

$\Rightarrow$ $$\alpha = \left(2 \times 3200 \times \frac{1}{4800}\right) - 1 = 0.33$$ **Ans.**

SOLVED EXAMPLE 2.3.4 **System Bandwidth Requirement**

A voice signal having frequency range of 300 Hz to 3400 Hz is sampled at 8000 samples/s. We may transmit these samples directly as PAM pulses or we may first convert each sample to a PCM format and use binary (PCM) waveform for transmission. Assume the filter roll-off factor of 1. Compute the minimum system bandwidth required for the detection of

(a) PAM signal with no ISI.
(b) Binary PCM waveform if the samples are quantized to 8-levels.

Solution Given voice signal frequency range = 300 Hz – 3400 Hz
Specified sampling frequency f_s = 8000 samples/s
Given roll-off factor of the filter used for pulse shaping $\alpha = 1$

(a) In a binary PAM transmission without ISI, symbol rate R_s = 8000 pulses/s
We know that the required system bandwidth for ISI free transmission is given by

$$W = \frac{1}{2}(1+\alpha)R_s$$

$\Rightarrow$ $$W = \frac{1}{2}(1+1) \times 8000 \text{ pulses/s} = 8 \text{ kHz}$$ **Ans.**

(b) In a binary PCM transmission using 8-level quantization, the bit rate

$R_b = f_s \times \log_2 L$; where L is the levels of quantization (L = 8 given)

$\Rightarrow$ R_b = 8000 sample/s × $\log_2(8)$ = 24 kbps

We know that the required system bandwidth for ISI free transmission is given by

$$W = \frac{1}{2}(1+\alpha)R_b$$

$\Rightarrow$ $$W = \frac{1}{2}(1+1) \times 24 \text{ kbps} = 24 \text{ kHz}$$ **Ans.**

Thus, system bandwidth required increases by *three* times for 8-level quantized PCM as compared to binary PAM transmissions.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 2.3.5 Determine symbol rate and the theoretical minimum system bandwidth needed for a 100 Mbps data using 16-level PAM without ISI.

Ex 2.3.6 A digital communication system using 16-level PAM has symbol rate of 25 M symbols per second. If the applicable system bandwidth is 13.75 MHz, then what should be the value of roll-off factor (α) of the transmitted pulse shape for zero ISI.

Ex 2.3.7 A baseband digital system uses 8-level PAM alongwith a raised cosine roll-off filter characteristics. The system has a frequency response of 2.4 kHz. If binary data is transmitted at 9600 bps data rate, then what would be the symbol rate and roll-off factor (α) of the transmitted pulse shape for zero ISI.

Ex 2.3.8 A band-limited analog signal has bandwidth of 4 kHz. It is sampled at minimum Nyquist rate. Assume the filter roll-off factor of 1. Show that the minimum system bandwidth required for the detection of binary PCM signal quantized to 128 levels and with no ISI is 56 kHz.

2.3.3 Equalization to Reduce ISI

We know that a sequence of pulses is attenuated (reduced in amplitude level) as well as distorted (dispersed mainly due to an attenuation of certain critical frequency components of the pulses transmitted) during transmission through a channel. The signal attenuation can be compensated by the preamplifier at the front-end of the digital regenerative repeater or receiver. However, the function of reshaping the received dispersed pulses in order to minimize ISI is performed by a device known as equalizer.

What is meant by Equalization?

Equalization helps the demodulator to recover a baseband pulse with the best possible signal-to-noise ratio, free of any intersymbol interference (ISI). In order to eliminate the effects of amplitude, frequency, and phase distortions causing ISI, amplitude, delay, and phase equalizers are used, respectively.

What is an Equalizer?

An *equalizer* is a specially designed filter which is located between the demodulator and decision making device at the receiver.

Principle of Working of an Equalizer

The transfer function of the equalizer should be the inverse of the transfer function of the channel so that the effect of the channel characteristics at the input of the decision-making device at the receiver is completely compensated. It implies that the equalizer should be able to adapt to the changes in channel characteristics with time.

We can say that an equalizer is needed to eliminate or minimize ISI with neighboring pulses at their respective sampling instants only because the receiver decision is based on sample values.

IMPORTANT!

A pulse waveform satisfying a Nyquist criterion can be generated directly using a *transversal filter equalizer*, also known as *tapped delay line*.

Adaptive Equalizer

The *adaptive equalizer* need to be trained by a known fixed length of training sequence bits for a minimum bit error rate. The adaptive equalizer at the receiver utilizes a recursive algorithm to evaluate the channel and estimate the filter coefficients to compensate for ISI. In practice, we do the following:

- We transmit a pseudorandom training sequence prior to actual data.
- This training sequence is known to the receiver which enables to compute the initial optimum tap coefficients of the adaptive equalizer.
- After the training sequence, the adaptive equalizer then uses the previously detected data symbols.
- The filter tap coefficients are updated using appropriate algorithm to estimate the equalization error.

This is the simplest form of equalizer described here.

The pulse to be generated is sampled with a sufficiently small sampling interval T_s, and the filter tap gains are set in proportion to these sample values in sequence, as shown in Figure 2.3.4.

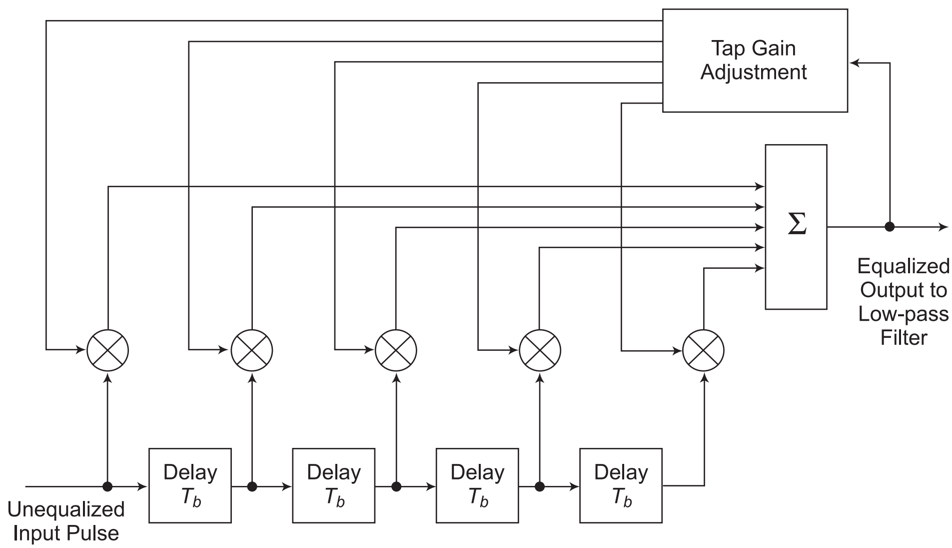


Figure 2.3.4 Linear Adaptive Tapped-Delay Line Equalizer

- When a narrow rectangular pulse having width T_s (same as sampling interval) is applied at its input, the output will be a staircase approximation of the input.
- This output is smoothed out by passing through a low-pass filter.
- Such an equalizer is easily adjustable to compensate against different channel characteristics, or even slowly time-varying channels.
- The time delay between successive taps is chosen to be same as the interval between pulses, i.e. T_b.
- This enables to force the equalizer output pulse to have zero ISI at the decision-making instants.
- That is why such type of equalizer is sometimes called *zero-forcing equalizer*.

Thus, the equalizer output pulses satisfy the Nyquist or the controlled ISI criterion. It may be noted that when number of delay lines is sufficiently large, then typically the residual nonzero sample values will be small, indicating that most of the ISI has been eliminated.

A **non-linear decision feedback adaptive equalizer** contains two filters: a forward filter and a feedback filter. The forward filter is similar to the linear adaptive equalizer whereas the feedback filter contains a tapped delay line whose inputs are the decisions made on the equalized signal. Figure 2.3.5 depicts a typical structure of non-linear adaptive equalizer.

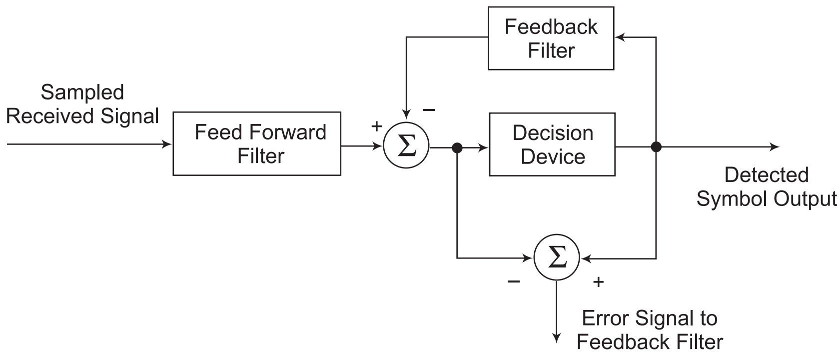


Figure 2.3.5 Nonlinear Adaptive Equalizer

As it can be seen that the received signal is the input to the feed-forward filter and the input to the feedback equalizer is the stream of the detected symbols. The filter coefficients are the estimates of the channel sampled impulse response. Due to past samples, intersymbol interference is cancelled while minimizing noise enhancement.

The Maximum-likelihood Sequence Estimation (**MLSE**) adaptive equalizer consists of two main parts: the adaptive channel estimator and the MLSE algorithm. The adaptive channel estimator measures the sampled impulse response of the channel, taken at symbol intervals. It is compared with the sequence of the sampled received signal with all possible received sequences. It then determines the most likely transmitted sequence of symbols. Figure 2.3.6 depicts a functional block schematic of the adaptive MLSE receiver.[13]

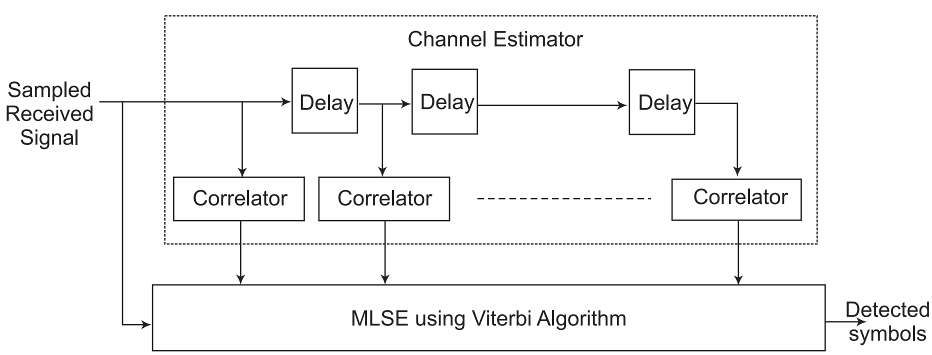


Figure 2.3.6 Adaptive MLSE Equalizer

[13]The MLSE is the optimal method of canceling the ISI. However, the complexity of MLSE receiver grows exponentially with the length of the channel impulse response. The decision feedback equalizer is particularly useful for channels with severe amplitude distortions and has been widely used in wireless communication applications. MLSE is mostly used in cellular communication applications such as GSM receivers.

2.3.4 Self-Synchronization

We know that the most important feature of baseband digital communication system such as PCM lies in its ability to control the effects of channel noise and distortion. Basically, there are three major functions that are required to be performed at the regenerative repeater or at the baseband receiver:

- *Reshaping the incoming distorted pulses* (due to ISI) by means of equalizer
- *Extracting the timing information* for sampling the equalized pulses at the instant where signal-to-noise ratio is maximum
- *Decision-making* based on the amplitude of the equalized pulse plus noise exceeding a predetermined threshold value.

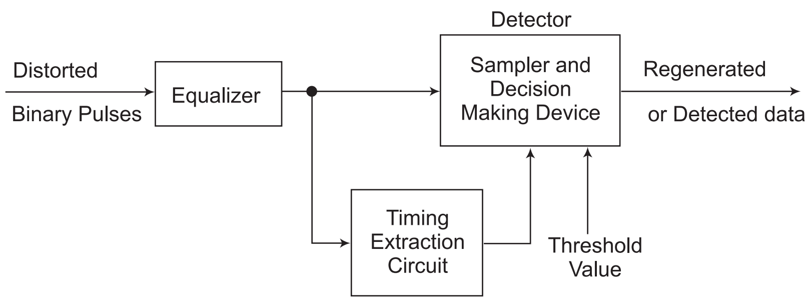


Figure 2.3.7 A Basic Functional Arrangement of Pulse Detection

Recall

We have discussed in the previous section as how an equalizer reshapes the received distorted pulses in order to compensate for amplitude, frequency, or phase distortions introduced due to imperfections in the transmission characteristics of the channel.

Bit Synchronization or Symbol Synchronization

The equalized pulses need to be sampled at precise instants, for which a clock signal at the receiver in synchronization with the clock signal used at the transmitter, delayed by the channel response.

Self-synchronization

The *self-synchronization* method is the most efficient one in which the *timing extraction* or *clock-recovery* circuit derives a periodic timing information from the received signal itself. It does not require additional channel capacity or transmission power as in case of pilot clock transmission along with data.

We have earlier seen the following:

- Polar RZ line coding techniques contains a discrete component of the clock frequency itself.
- In case of bipolar line codes, it has to be converted to a polar one by rectification process, which can then be used to extract timing information.

The extracted timing instants are used for sampling the equalized pulses at that instant of time where signal-to-noise ratio is maximum.

Small random deviations of the incoming pulses from their actual positions are always present due to presence of noise, interference, and mistuning of the clock circuits. Variations of the pulse positions or sampling instants cause *timing jitter*. This may have severe effect on the proper functioning of timing extraction and clock recovery circuit. If it is excessive, it may significantly degrade the performance of cascaded regenerative repeaters.

The signal received at the detector consists of the equalized pulses alongwith a random channel noise. The detector samples each pulse based on the clock information provided by the timing extraction circuit. The *decision-making device* is enabled when the amplitude of the equalized pulse plus noise exceeds a predetermined threshold value. It can be seen that the noise can cause error in pulse detection.

Detection Error Probability

In case of polar transmission, the threshold value is set as zero.

- If the pulse sample value is positive, the symbol is detected as binary 1.
- If the pulse sample value is negative, the symbol is detected as binary 0.

It may so happen that the random noise has a large negative value at the sampling instant for the transmitted symbol 1, the net value may be slightly negative. This will result into wrong detection of symbol 1 as 0.

Similarly, at some other sampling instant, the random noise has a large positive value for the transmitted symbol 0, the net value may be slightly positive. This will also result into wrong detection of symbol 0 as 1.

Generally, the performance of digital communication systems is typically specified by the average ratio of the number of detection errors to the total number of transmitted data, known as *detection error probability*.

Self-Assessment Exercise linked to LO 2.3

Q2.3.1 List different ways to minimize the effects of intersymbol interference.

Q2.3.2 Design a procedure to identify the phenomenon responsible for intersymbol interference in baseband binary data transmission system.

Q2.3.3 Analyze Nyquist's first and second criteria for pulse shaping and highlight the salient aspects to eliminate or control ISI.

Q2.3.4 Define roll-off factor in case of raised-cosine filter. Interpret its different values with the help of suitable diagram.

Q2.3.5 Classify equalizers based on their structures as well as type of algorithms used for decoding data.

Q2.3.6 Identify three major functions that are required to be performed at the regenerative repeater or at the baseband receiver. Illustrate it with the help of suitable illustration.

Q2.3.7 Show that the theoretical minimum system bandwidth needed for a 20 Mbps signal using 4-level PAM without ISI will be 5 MHz.

Q2.3.8 Binary data at 9600 bps are transmitted using 8-level PAM alongwith a raised cosine roll-off filter characteristics. The system has a frequency response of 2.4 kHz. Show that the symbol rate is 3200 symbols/s and roll-off factor is 0.5.

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/111

If you have been able to solve the above exercises then you have successfully mastered

LO 2.3: *Hypothesize Nyquist pulse-shaping criteria and outline equalizer schemes to minimize the effects of ISI.*

Sure of what you have learnt so far? For answers, scan the QR code

OR

visit http://qrcode.flipick.com/index.php/108

Be ready for the next sections!

Mid-Chapter Check

So far you have learnt the following:

- Line Coding and its Types
- Intersymbol Interference (ISI)
- Nyquist Pulse Shaping and Equalization

Therefore, you are now skilled to complete the following tasks:

LO 2.1

MQ2.1 Classify various line codes based on different voltage level representations of binary data 0 and 1. Compare and contrast the clock recovery and error-detection capabilities of non-return-to-zero (NRZ) and return-to-zero (RZ) line coding techniques.

MQ2.2 Discriminate between bipolar NRZ and Manchester line codes for synchronous transmission of binary data.

MQ2.3 With the help of appropriate example data and illustrations, evaluate and confirm that Manchester line code is preferred over BP-RZ-AMI line code.

MQ2.4 Calculate the transmission bandwidth requirement for unipolar NRZ line code if the pulse width is 10 µs.

MQ2.5 Generate the waveform for transmission of binary data sequence 1 0 1 1 0 0 1 1 0 1 0 1 over a baseband channel using BP-RZ-AMI and Manchester line-coding techniques.

LO 2.2

MQ2.6 Describe the problem of Intersymbol interference (ISI) in context to baseband transmission of digital signals. How can it be avoided?

MQ2.7 Select the four parameters observed from an eye pattern. How can you judge from the eye diagram that the binary symbol is received faithfully?

LO 2.3

MQ2.8 The output of a baseband binary data transmission system model is given by $y(t_i) = \mu A_i + \mu \sum_{\substack{k=-\infty \\ k\neq i}}^{\infty} A_k v(iT_b - kT_b)$. Identify the parameter(s) on which intersymbol interference depend.

MQ2.9 Outline Nyquist's criteria for distortionless baseband transmission.

MQ2.10 Moderate to severe intersymbol interference is experienced in a wireless communication channel. For a given number of taps, construct a typical linear adaptive tapped-delay line-feedback equalizer.

2.4 DIGITAL MULTIPLEXING AND TDM

LO 2.4

Recall

The analog signals are sampled to generate Pulse Amplitude Modulation (PAM) signals. A number of PAM signals can be multiplexed using Time-Division Multiplexing (TDM) technique, known as a *PAM/TDM system*. Each signal occupies a short time slot and multiple signals are then separated from each other in the time domain. The main advantage of such scheme is that entire bandwidth of the communication channel is available to each signal being transmitted, thereby resulting into increased data rate. The concept of TDM can be extended for data sources such as PC as well as pulse-code modulation (PCM) signals.

Need of Multiplexing

The available bandwidth of a transmitting medium linking more than one source is usually greater than the bandwidth requirement of individual sources. Several low-bit-rate digital signals from different sources such as a computer output, or a digitized voice signal (PCM) can be multiplexed to form one high-bit-rate digital signal. This combined signal can be then transmitted over a high-bandwidth medium (broadband channel) such as optical fiber, terrestrial microwave, satellite channel. Thus, multiplexing enables to share the communication link among many sources for simultaneous operation.

What is Digital Multiplexing

Digital multiplexing **is achieved by interleaving data from various sources in the time domain either on a bit-by-bit basis (known as *bit interleaving*), or on a word-by-word basis (known as *word interleaving*). This process of digital multiplexing is known as *time-division multiplexing* (*TDM*) that allows several signal sources to share the available high bandwidth of a communication link, occupying a portion of time by each source.**

Figure 2.4.1 shows the fundamental arrangement of a 4-channel TDM system.

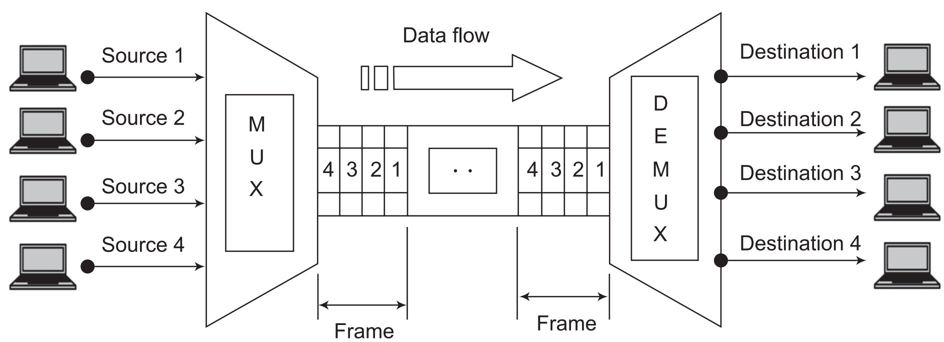


Figure 2.4.1 A 4-channel TDM System

As shown in the figure,

- Different data are arranged into smaller parts, called *packets (1, 2, 3, 4, …),* of equal length;
- Packets are then interleaved into their assigned time slots to form a *frame*; and
- A *header*, containing address and packet sequence number information, precedes each packet.

IMPORTANT!

The minimum length of the multiplexed frame must be a multiple of the lowest common multiple of the incoming channel bit rates. The minimum bandwidth of the multiplexed signal will be the multiplication of individual bandwidth and the number of signals.

The interleaved packets are transmitted in form of frames and received by the destination receiving stations through demultiplexer. The appropriate packets are extracted by each receiver and reassembled into their original form, as shown. Let us try to understand the design aspects of a TDM system with the help of following example.

SOLVED EXAMPLE 2.4.1 Design of a TDM System

Each of four channels carrying digital data sends binary data at 6400 bps rate. These channels are multiplexed using TDM. Let the number of bits per channel is 8.

(a) Calculate the size of one TDM frame.
(b) Determine the frame rate and duration of one TDM frame?
(c) Find the bit rate for the link.

Comment on the relationship between link bit rate and channel bit rate.

Solution

(a) Number of bits from each channel carried in one TDM frame = 8 bits (given)
Number of channels = 4 channels (given)
Therefore, size of one TDM frame = 4 channels × 8 bits = **32 bits** **Ans.**

(b) Transmission bit rate by each channel = 6400 bps (given)
Number of bits from each channel carried in one TDM frame = 8 bits (given)
Therefore, the frame rate = 6400 bps/8 bits = **800 frames/second** **Ans.**

(c) Duration of one TDM frame = 1/800 = **1.25 ms** **Ans.**
Number of TDM frames in one second = 800 frames ….. as calculated in Part (b)
Number of bits in one TDM frame = 32 bits ….. as calculated in Part (a)
Hence, bit rate for the link = 800 frames/second × 32 bits/frame = **25600 bps** **Ans.**

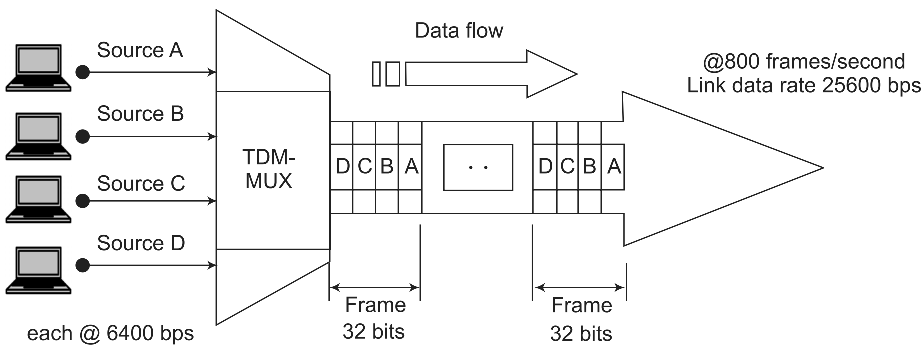


Figure 2.4.2 TDM System Design

Comments on relationship between link bit rate and channel bit rate:

Transmission bit rate by each channel = 6400 bps (given)

Link bit rate = 25600 bps … as calculated in part (c)

Hence, **link bit rate = 4 × channel bit rate**

Thus, it is concluded that the bit rate of the communication link is *n* times the channel bit rate where *n* is the number of channels being multiplexed.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.4.2 Design a TDM system for multiplexing 16 similar digital sources transmitting digital data at 6400 bps rate. Let the number of bits per channel is 8. Calculate the size of one TDM frame, the frame rate, duration of one TDM frame, and the combined bit rate for the link.

2.4.1 Synchronous TDM

Recall

For allocation of time slots to different sources to a digital multiplexer, there are two possibilities: fixed-length, and dynamically, i.e., on requirement basis.

Synchronous Digital Multiplexer

A digital multiplexer which allocates fixed-length time slots to all input sources connected to it is known as *synchronous digital multiplexer*. Synchronous TDM systems assign time slots of equal length to all packets regardless of whether or not any data is to be sent by the sender with an assigned time slot.

IMPORTANT!

A master clock governs all the data sources, with the provisions to distribute the master clock signal to all the sources. Hence, this method is called synchronous time-division multiplexed.

Good to Know!

Due to elimination of bit rate variations, the synchronous digital multiplexer systems attain very high throughput efficiency. It is easy to implement but it occupies higher bandwidth as well as the efficiency of channel bandwidth utilization is quite low.

Two Ways of Synchronous TDM

TDM is widely used in digital communication systems to maximize the use of channel capacity. The synchronous TDM can be achieved in two ways:

- Bit Interleaving
- Word Interleaving

Bit Interleaving In bit (binary digit) interleaving, one by one bit is taken from each of n input data channels. After the first bit from samples of all channels are taken, the multiplexer takes the second bit from all channel samples, and so on.

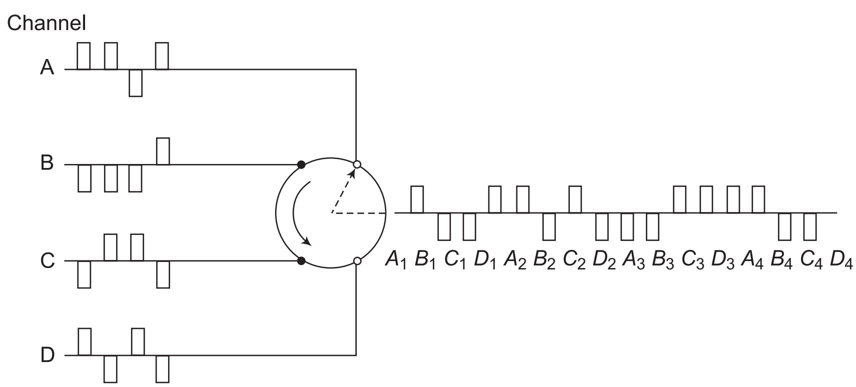


Figure 2.4.3 Synchronous TDM–Bit Interleaving

Word Interleaving All code bits of the sample of the first channel are taken together followed by all code bits of the sample of second channel and so on. In this method, the desired multiplexing speed is less than that required in the first method of synchronous TDM.

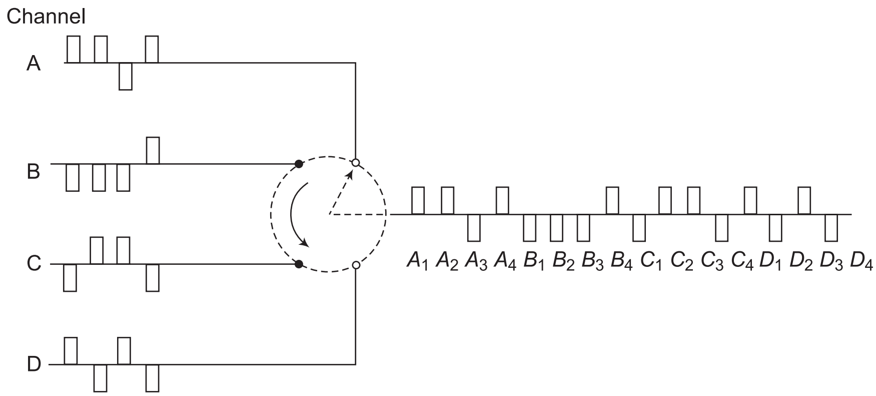


Figure 2.4.4 Synchronous TDM–Word Interleaving

Other Aspects of Synchronous TDM

- A synchronous TDM frame is constituted by combining all bits corresponding to a specific sample code from all channels.
- Thus, if there are n number of channels and m bits per sample, then the size of a frame is $(n \times m)$ bits.
- A synchronizing bit is added at the end of each frame for the purpose of providing synchronization between the multiplexer on the transmitting side and demultiplexer on the receiving side.
- The digital signal is bandlimited to the same frequency, resulting in the same sampling frequency for all channels.[14]

2.4.2 Asynchronous TDM

Asynchronous Digital Multiplexer

A digital multiplexer which allocates variable-length time slots to all input sources connected to it dynamically on requirement basis is known as *asynchronous TDM,* or *statistical TDM***.** All devices attached to the input of asynchronous multiplexer may not transmit data all the time.

IMPORTANT!

Asynchronous digital multiplexer has a finite number of low-speed data input lines with one high-speed multiplexed data output line. Each input line has its own digital encoder and buffer amplifier. The multiplexer scans the input buffers collecting data until a TDM frame is filled. As such control bits must be included within the frame.[15]

[14] *Quasi-synchronous digital multiplexers* are arranged in a hierarchy of increasing bit rates to constitute the basic building blocks of an interconnected digital telecommunication system. Quasi-synchronous multiplexers are needed when the input bit rates vary within specific limits. Quasi-synchronous multiplexers should have a sufficiently high output bit rate so that it can accommodate the maximum input bit rate.

[15] The asynchronous digital multiplexers are used for the digital data sources which produce data in the form of bursts of characters with a variable spacing between data bursts. Character interleaving and buffering techniques make it possible to merge these data sources into a synchronous multiplexed bit stream.

Good to Know!

The data rate on the multiplexed line is lower than the combined data rates of the attached devices. The frame format used by a statistical TDM multiplexer is significant in determining the system performance. It is quite desirable to have minimum overhead in order to improve data throughput.

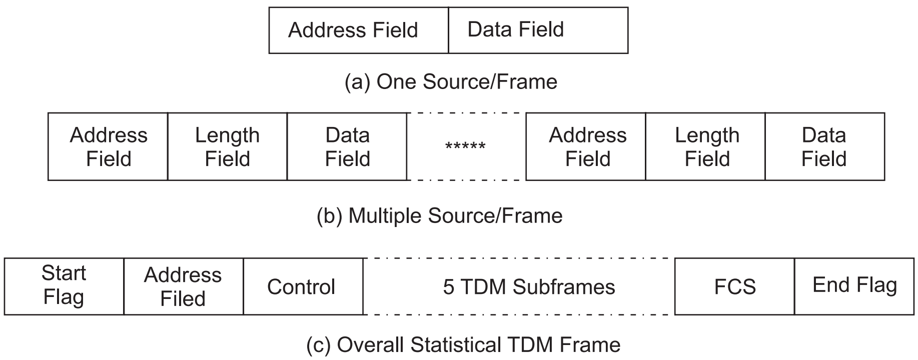


Figure 2.4.5 Basic Asynchronous TDM Frame Structure

In general, the overall statistical TDM frame includes the following fields:

- Start and end flags to indicate the beginning and end of the frame
- An address field to identify the transmitting source
- A control field
- A statistical TDM sub frame containing data
- A frame check sequence (FCS) field that provides error detection

With multiple sources, the TDM frame consists of sequence of data fields labeled with an address and a bit count. The address field can be shortened by using relative addressing technique. Each address specifies the position of the current transmit source relative to the previously transmitted source and the total number of sources. The data field length is variable and limited only by the minimum length of the frame. The length field indicates the size of the data field.

Procedure for Asynchronous TDM Operation

Step I Different information data signals are bandlimited to different frequencies using low-pass filters having different cut-off frequencies.

Step II Accordingly, they are sampled at different sampling frequencies.

Step III The samples are then stored on different storage devices. It may be noted that the rate of storing data of each storage device is different due to the different sampling frequencies.

Step IV The stored signals are retrieved at different rates in such a way that the output sample rate of each device is the same.

Step V Thus, these signals can now be synchronously time-division multiplexed, and then transmitted.

Step VI At the receiver, this process is just reversed to recover each signal.

SOLVED EXAMPLE 2.4.3 Synchronous versus Asynchronous TDM

Consider four sources *A*, *B*, *C*, and *D*. The data on each of these sources are sampled independently in pre-defined time slots t_1, t_2, t_3, and t_4. The availability of data on each of four sources in different time slots is as follows:

Source A: Data available during the time slots t_1, and t_4
Source B: Data available during the time slots t_1, t_2, and t_3
Source C: Data available during the time slots t_2, and t_4
Source D: Data available during the time slot t_4

Illustrate the operation of synchronous and asynchronous TDM.

Solution

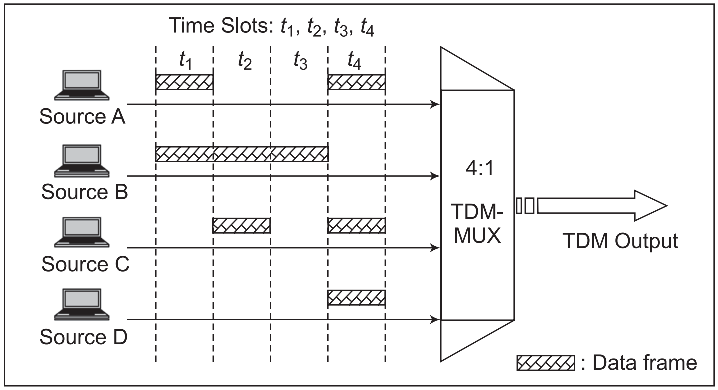


Figure 2.4.6 Illustration of TDM-MUX operation for Example 2.4.3

In synchronous TDM, the data frames from all sources in a particular time slot (irrespective of whether there is data present or no data present) form a TDM frame and are sent together, followed by data frames of next time slot.

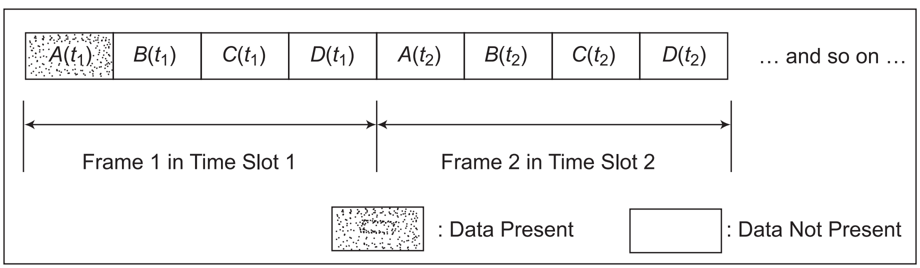


Figure 2.4.7 Synchronous TDM Operation

In asynchronous TDM, the data frames from only those sources which contain data in a particular time slot form a TDM frame and are sent together, followed by data frames of next time slot.

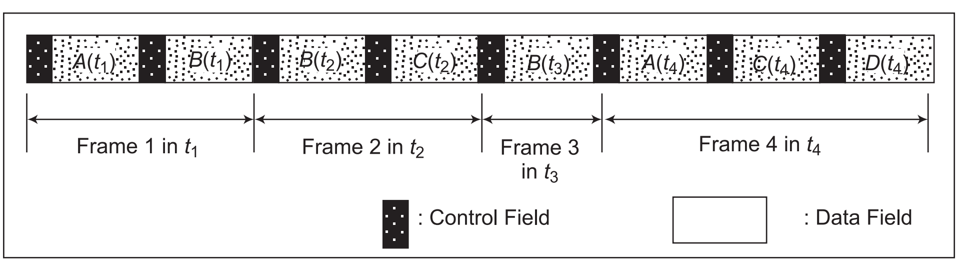


Figure 2.4.8 Asynchronous TDM Operation

This illustrates not only the operational difference between synchronous and asynchronous TDM but also higher efficiency obtained with asynchronous TDM.[16]

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.4.4 Consider four data sources *A*, *B*, *C*, and *D*. The data on each of these sources are sampled independently in pre-defined time slots t_1, t_2, t_3, and t_4. The availability of data on each of four sources in different time slots is as below:

Source *A*: Data available during the time slots t_2, and t_3
Source *B*: Data available during the time slots t_2, t_3, and t_4
Source *C*: Data available during the time slots t_1, and t_3
Source *D*: Data available during the time slot t_1

Illustrate the operation of synchronous and asynchronous TDM.

Self-Assessment Exercise linked to LO 2.4

Q2.4.1 Summarize the salient features of time-division multiplexing.

Q2.4.2 The frame format used by asynchronous TDM multiplexer is significant in determining the system performance. Generate hypotheses to account for enhanced efficiency of asynchronous TDM technique.

Q2.4.3 Implement the operation of asynchronous TDM for multiplexing of data from four different sources *A*, *B*, *C*, and *D*. The data on each of these sources are sampled independently in pre-defined time slots t_1, t_2, t_3, and t_4. The availability of data on each of four sources in different time slots is as below:

Source *A*: Data available during the time slots t_2, and t_4
Source *B*: Data available during the time slots t_1, and t_3
Source *C*: Data available during the time slots t_2, t_3, and t_4
Source *D*: Data available during the time slot t_3

Q2.4.4 Predict advantages and disadvantages of asynchronous time-division multiplexing.

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/103

If you have been able to solve the above exercises then you have successfully mastered

LO 2.4: *Illustrate the concept of digital multiplexing and design TDM systems.*

[16] A lower data rate is required in asynchronous TDM as compared to that needed by synchronous TDM to support the same number of inputs. Alternately, asynchronous TDM can support more number of users if operating at same transmission rate. But there is more overhead per time slot because each time slot must carry an address as well as data field.

LO 2.5

2.5 DIGITAL CARRIER SYSTEMS

Recap

The process of digital multiplexing, known as *time-division multiplexing (TDM),* allows several data signal sources (such as PCM) to share the available high bandwidth of a communications channel, occupying a portion of time by each source. This combined signal can be then transmitted over a high-bandwidth transmitting medium (broadband channel) such as digital carrier lines.

What We Discuss Here .....

We begin with a single-channel PCM transmission system that serves as fundamental building block for most of digital carrier systems based on TDM, known as digital signal at zero level (*DS*0), operating at 64 kbps data rate.

North American digital carrier system, also known as *T lines*, are digital transmission lines which are designed to represent digitized audio or video, or digital data signal. Then we elaborate two popular types of digital carrier systems—North American T (Transmission) or DS (Digital System) lines, and European E lines.

A single-channel PCM transmission system is the fundamental building block for most of the TDM systems in the North American or European digital signal hierarchy with a DS0 channel (digital signal level 0 channel), as shown in Figure 2.5.1.

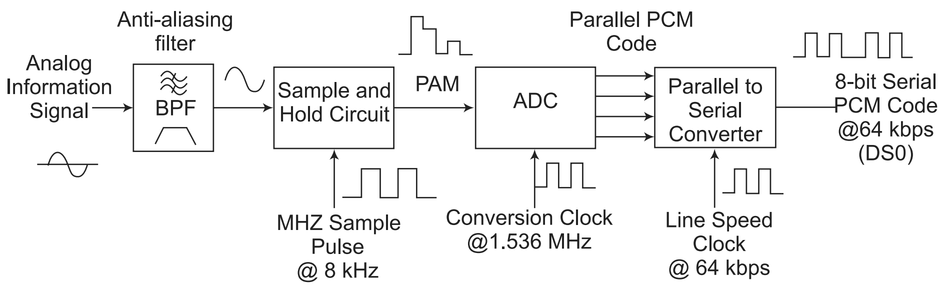


Figure 2.5.1 Single-Channel PCM System

As shown, the sample pulse rate is 8 kHz, i.e., 8000 samples/second. An 8-bit PCM code per sample is used for encoding the quantized signal in analog-to-digital converter (ADC). The transmission line speed of a single channel PCM system is given as 8000 samples/second multiplied by 8 bits/sample, that is, **64 kbps**.

2.5.1 North American Digital Carrier System

In general, a *digital carrier system* is a baseband digital communication system that uses digital multiplexers such as TDM for multiplexing pre-specified number of data channels of given data rates. North American digital carrier system, also known as *T-lines*, are digital transmission lines which are designed to represent digitized audio or video, or digital data signal. There are four levels of digital multiplexing in North American digital carrier system—T1, T2, T3, T4, and T5. Figure 2.5.2 depicts a functional block diagram of North American Digital Carrier System, or Digital Signal Hierarchy.

Now we will describe various levels of T-lines with calculation of respective data rates.

T1 Digital Carrier System

T1 (Transmission One) digital carrier system, also known as DS1 (digital level 1) digital carrier system is a North American digital hierarchy ANSI standards for telecommunications and is recognized by ITU-T.

IMPORTANT!

T1 digital carrier system is designed to multiplex 24 PCM voice grade channels (each carrying digitally encoded voice-band telephone signals or data), each operating at a maximum data rate of 64 kbps.

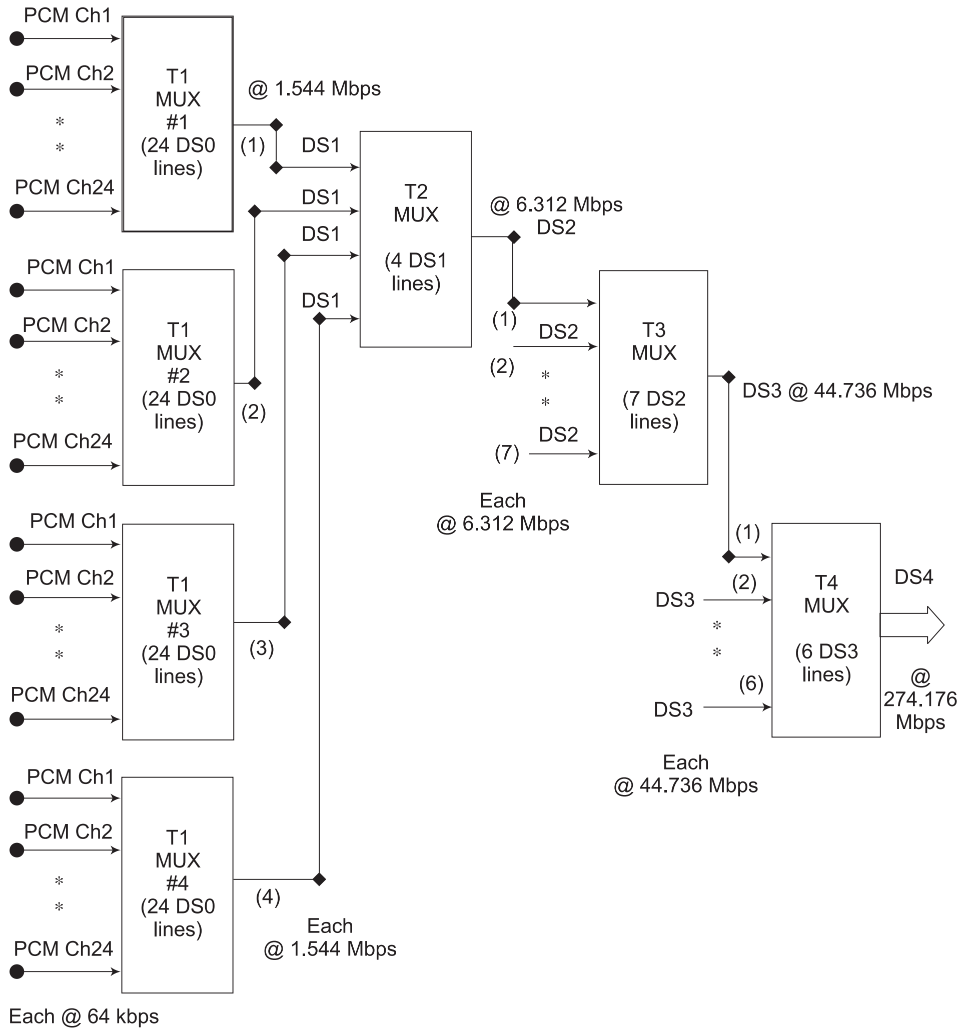


Figure 2.5.2 North-American Digital Carrier System

Figure 2.5.3 depicts a functional block schematic of T1 TDM-PCM encoded digital carrier system.

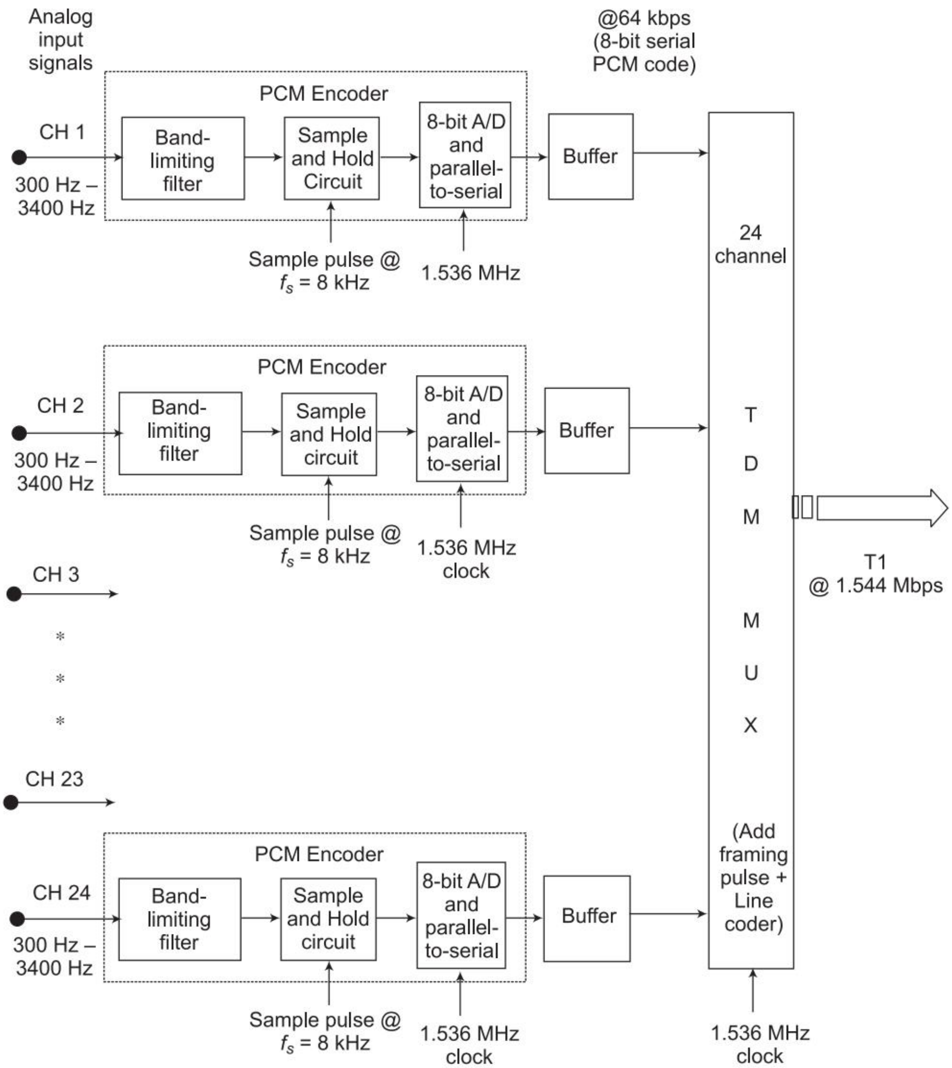


Figure 2.5.3 T1 TDM-PCM Encoded Digital Carrier System

As stated earlier, twenty-four DS0 lines are multiplexed into a digital signal at level 1 (*DS1 line*), commonly known as T1 digital carrier system or simply T1 line.[17]

Synchronization in T1 Digital Carrier System

It is essential to make available the data-encoded bits as well as synchronizing or timing information at the receiver. *Synchronization* in T1 digital carrier system is provided by adding one bit at the end of the 192-bits data-encoded in each frame. Thus, one TDM frame essentially contains 193 bits.

[17] The transmission data rate (line speed) of a T1 digital carrier system is 1.544 Mbps. This includes an 8 kbps sync bit. It uses BP-RZ-AMI and B8ZS line encoding techniques. T1 lines are designed to interconnect stations that are placed up to 80 km apart. The transmission medium used is generally twisted pair metallic cable (19″–22″ wide). T1 line is multiplexed further to constitute T2/T3/T4 lines to provide higher data rates.

Figure 2.5.4 shows PCM-TDM T1 frame using channel associated signaling.

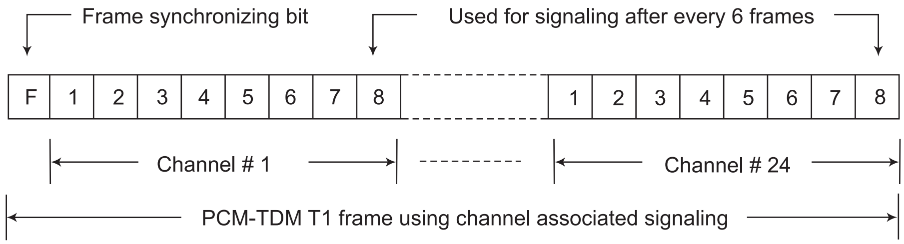


Figure 2.5.4 PCM-TDM T1 Frame

In order to establish synchronization at the receiver,

A special 12-bit code comprising of 1 1 0 1 1 1 0 0 1 0 0 0 is transmitted repetitively once every 12 frames.

The synchronizing bit in a TDM frame occur once per frame or every 125 μs

Thus, time taken for 12 frames = 12 × 125 μs = 1500 μs or 1.5 ms

The frame sync code rate = 1/(1.5 ms) = 667 frame sync code per second.

Channel Associated Signaling in T1 Digital Carrier System

A process of *bit-slot sharing* is utilized which allows a single PCM channel to transmit both voice and signaling information data.

In bit-slot sharing scheme, the following hold:

- The 8^{th} bit (LSB) of each encoded sample of every 6^{th} frame is used for signaling purpose only.
- During five successive TDM frames, each sample is PCM-encoded using 8 bits and in 6^{th} TDM frame, the samples are encoded into 7 bits only; the 8^{th} bit (LSB) is used for signaling.
- In one frame out of every six frames, each of the least significant bits in the 24 samples is used for signaling information rather than as part of the PCM signal.

This pattern is repeated every 6 frames. Thus, in six TDM frames, the number of bits used for quantization encoding is (5 frames × 8 bits/sample of a channel + 1 frame × 7 bits/sample of that channel) 47 bits.

Therefore, average number of bits per sample is (47 bits/6 frames) 7.82 bits approximately.

This implies that the frequency of the bits used for signaling is one-sixth of the frame bit rate.

Since frame rate is 8000 frames/second, the signaling bit rate of T1 digital carrier system is ($1/6^{th}$ of 8000) 1.333 kbps.

This type of signaling is known as ***channel-associated signaling*.**

ATTENTION

There is a variant of T1 digital carrier system, known as T1C digital carrier system which uses 48 DS0 lines or 2 T1 lines. All other parameters are similar to that of T1 digital carrier system.

SOLVED EXAMPLE 2.5.1 **T1 Digital Carrier System Line Speed**

Show that transmission data rate or the line speed for a standard T1 digital carrier system is 1.544 Mbps.

Solution

Given: In a standard T1 digital carrier system,

Number of channels in each TDM frame = 24 channels

Number of bits in each channel = 8 bits

Number of TDM frames in one second = 8000 frames

Number of sync bits in one TDM frame = 1 bit

Step I: Number of data bits in each TDM frame = 24 channels/frame × 8 bits/channel

= 192 bits/frame

Step II: Duration of one frame = 1/8000 frames = 125 μs

Step III: Number of data bits per second = 192 bits/frame × 8000 frames/second

or, Data transmission speed = 1.536 Mbps (without sync bit)

Step IV: Total number of bits in each TDM frame = 192 + 1 = 193 bits/frame

We know that the samples must be transmitted at the same data rate as they were obtained from their respective signal sources. This requires the multiplexed signal to be sent at a rate of 8000 frames per second. Therefore,

Step V: Data rate or line speed of T1 line = 193 bits/frame × 8000 frames/second

= **1.544 Mbps**

Alternately,

Step V: Duration of one bit in a frame = 125 μs / 193 bits = 0.6476 μs

Step VI: Data rate of T1 line= 1/(0.6476 μs) = **1.544 Mbps** **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.5.2 Show that the minimum bandwidth required to transmit multiplexed data on T1 line is 772 kHz.

T2 Digital Carrier System

Four T1 lines (equivalent to 96 DS0 lines) are multiplexed into a digital signal at level 2 (*T2 line*), also known as DS2 digital carrier system. Control bits or additional bits, sometimes called *stuffed bits* are required to be added to yield a steady output data rate. Additional synchronizing bits totaling 17 sync bits are required in T2 digital carrier systems due to increased number of channels to be multiplexed.[18]

SOLVED EXAMPLE 2.5.3 **Transmission Data Rate for T2 Line**

Show that transmission data rate or the line speed for a standard T2 digital carrier system is 6.312 Mbps.

[18] The transmission data rate (line speed) of a T2 digital carrier system is 6.312 Mbps. This includes (17 × 8) kbps sync bit. It uses B6ZS RZ line coding technique. T2 lines are designed to interconnect stations that are placed up to 800 km apart with repeater spacing of 3.7 km typical.

Solution

Given: In a standard T2 digital carrier system,

Number of T1 lines multiplexed in a DS2 line = 4

Data rate of one T1 line = 1.544 Mbps

Number of sync bits at T2 level = 17 bits

Step I: Raw data rate of T2 line = 4 × 1.544 Mbps = 6.176 Mbps

Step II: After adding 17 sync bits at T2 level,

Net data rate of T2 line = 6.176 Mbps + (17 × 8) kbps = **6.312 Mbps** **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.5.4 Show that the minimum bandwidth required to transmit multiplexed data on T2 line is 3156 kHz.

T3 Digital Carrier System

Seven T2 lines (equivalent to 28 T1 lines, or 672 DS0 lines) are multiplexed into a digital signal at level 3 (*T3 line*), also known as DS3 digital carrier system. Additional synchronizing bits totaling 69 sync bits are required in T3 digital carrier systems due to much increased number of channels to be multiplexed.[19]

SOLVED EXAMPLE 2.5.5 Transmission Data Rate for T3 Line

Show that transmission data rate or the line speed for a standard T3 digital carrier system is 44.736 Mbps.

Solution

Given: In a standard T3 digital carrier system,

Number of T2 lines multiplexed in a T3 line = 7

Data rate of one T2 line = 6.312 Mbps

Number of sync bits at T3 level = 69 bits

Step I: Raw data rate of T3 line = 7 × 6.312 Mbps = 44.184 Mbps

Step II: After adding 69 sync bits at T3 level,

Net data rate of T3 line = 44.184 Mbps + (69 × 8) kbps = **44.736 Mbps** **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 2.5.6 Show that the minimum bandwidth required to transmit multiplexed data on T3 line is 22.368 MHz.

T4 Digital Carrier System

Six T3 lines (equivalent to 42 T2 lines, or 168 T1 lines, or 4032 DS0 lines) are multiplexed into a digital signal at level 4 (*T4 line*), also known as *DS4 digital carrier system*. Total 720 sync

[19] The transmission data rate (line speed) of a T3 digital carrier system is 44.736 Mbps. This includes (69 × 8) kbps sync bit. It uses bipolar 3ZS RZ line-coding technique.

bits are required in T4 digital carrier systems due to very much increased number of channels to be multiplexed.[20]

SOLVED EXAMPLE 2.5.7 **Transmission Data Rate for T4 Line**

Show that transmission data rate or the line speed for a standard T4 digital carrier system is 274.176 Mbps.

Solution

Given: In a standard T4 digital carrier system,

Number of T3 lines multiplexed in a T4 line = 7

Data rate of one T3 line = 44.736 Mbps

Number of sync bits at T4 level = 720 bits

Step I: Raw data rate of T4 line = 6 × 44.736 Mbps = 268.416 Mbps

Step II: After adding 720 sync bits at T4 level,

Net data rate of T4 line = 268.416 Mbps **+** (720 × 8) kbps = **274.176 Mbps**[21] **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 2.5.8 Show that the minimum bandwidth required to transmit multiplexed data on the T4 line is 137.088 MHz.

Ex 2.5.9 Show that the bandwidth efficiency of a T4 line is 11.77%.

Table 2.5.1 summarizes the key parameters for various levels of T-lines.

Table 2.5.1 Key Parameters for T Lines

Digital Service Line#	Transmission Carrier Line#	No. of PCM voice channels	Raw Data Rate (Mbps)	Net Data Rate (Mbps)	Minimum bandwidth	Typical medium used
DS1	T1	24 DS0 lines	1.536	1.544	**772** kHz	Twisted-pair cable
DS1C	T1C	48 DS0 lines (2 T1 lines)	3.088	3.096	1.548 MHz	Twisted-pair cable
DS2	T2	96 DS0 lines (4 T1 lines)	6.176	6.312	3.156 MHz	Twisted-pair cable, micro-wave

(*Contd.*)

[20] The transmission data rate (line speed) of a T4 digital carrier system is 274.176 Mbps. This includes (720 × 8) kbps sync bit. It uses polar NRZ line coding technique. The maximum system length is up to 800 km with typical repeater spacing of 1.6 km.

[21] It may be noted that 17 synchronizing bits are required in T2 digital carrier systems due to increased number of channels to be multiplexed whereas 69 synchronizing bits are required in T3 digital carrier systems due to much increased number of channels to be multiplexed. Similarly, total 720 synchronizing bits are required in T4 digital carrier systems due to very much increased number of channels to be multiplexed.

Table 2.5.1 (*Contd.*)

DS3	T3	672 DS0 lines (28 T1 lines, or 7 T2 lines)	44.184	44.736	22.368 MHz	Coaxial cable, microwave
DS4	T4	4032 DS0 lines (168 T1 lines, or 42 T2 lines, or 6 T3 lines)	268.416	274.176	137.088 MHz	Coaxial cable, optical fiber cable
DS-5	T5	8064 DS0 lines	516.096	560.16	280.08 MHz	Optical fiber cable

ATTENTION

The input signals to a T1 digital multiplexer need not to be restricted only to digitized voice channels alone. Any digital signal of 64 kbps of appropriate format can be transmitted. The case of higher levels such as T2, T3, and T4 is similar. The T1 and T2 digital carrier systems are widely used in voice-band telephone, video phone, and digital data applications. Whereas T3 and T4 digital carrier system finds application in TV broadcast with higher capacity also.

2.5.2 European Digital Carrier System

The *European digital carrier system* (recommended by the ITU as an standard) has a TDM digital multiplexing hierarchy similar to the North American digital hierarchy except that it is based on the basic 32-line-slots (30-voice channels of 64 kbps rate and 2 control channels) *E1 lines* (European one) in contrast to 24 T1 lines in North American hierarchy.

In each E1 line, 30 voice-band channels are time-division multiplexed in 32 equal time slots in a 125 μs frame. Each time slot has 8 bits. Time slot 0 is used for a frame alignment pattern whereas time-slot 17 is used for a common signaling channel on which the signaling for all 30 voice-band channels is accomplished. There are four levels of digital multiplexing in European digital signal hierarchy, as shown in Figure 2.5.5.

The European TDM scheme uses added-channel framing method in which one of the 32 time slots in each frame is dedicated to a unique synchronizing bit sequence. The average number of bits needed to acquire frame synchronization in E1 system is 128.5 bits. Corresponding to E1 transmission line speed of 2.048 Mbps, the synchronizing time is approximately 62.7 μs. Table 2.5.2 summarizes number of PCM voice channels and transmission bit rate for various E-lines.

Table 2.5.2 European Digital Carrier Hierarchy (E lines)

Transmission Carrier Line#	No. of PCM voice channels	No. of PCM control channels	Transmission bit Rate (Mbps)
E1	30	2	2.048
E2 (= 4 E1 lines)	120	8	8.448*
E3 (= 4 E2 lines)	480	32	34.368*
E4 (= 4 E3 lines)	1920	128	139.264*
E5 (= 4 E4 lines)	7680	512	565.148*

*Includes synchronizing bits

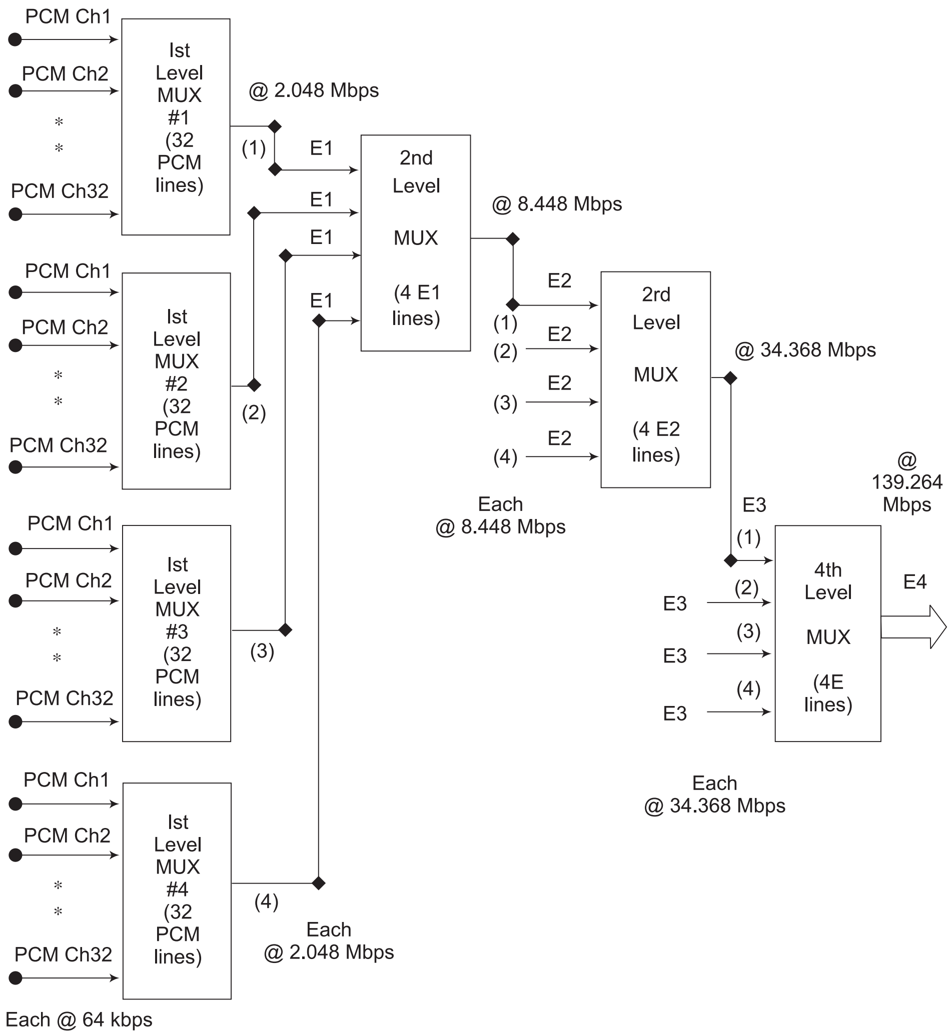


Figure 2.5.5 Digital Signal Hierarchy (European)

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/106

Self-Assessment Exercise linked to LO 2.5

Q2.5.1 A PCM-TDM system multiplexes 24 voice-band channels. The sampling rate is 9000 samples per second. Each sample is encoded into seven bits, and a framing bit is added in each frame. Determine line speed and the minimum transmission bandwidth, if BP-AMI-RZ line coding technique is used.

Q2.5.2 A PCM-TDM system multiplexes 20 voice-band channels. Each sample is encoded into 8 bits, and a framing bit is added in each frame. The sampling rate is 10,000 samples per second. The line encoding technique used is BP-AMI-RZ. Calculate the line speed and the minimum Nyquist bandwidth.

Q2.5.3 Evaluate the raw transmission data rate of an E1 line.

> ✓ **If you have been able to solve the above exercises then you have successfully mastered**
>
> **LO 2.5:** *Integrate TDM-PCM encoded digital carrier systems and calculate their data rates.*

Key Concepts

- alternate mark inversion (AMI)
- asynchronous TDM
- bipolar code
- framing bits
- E-lines
- eye diagram
- interleaving
- line code
- Manchester line code
- multiplexing
- non-return-to-zero (NRZ)
- polar line code
- polar NRZ
- return-to-zero (RZ) code
- synchronous transmission
- time-division multiplexing (TDM)
- T-lines
- unipolar code

Learning Outcomes

LO 2.1

- Line coding is the process of converting binary data (1 and 0) to a digital signal (voltage levels).
- Line-coding techniques must eliminate the dc component and provide a means of synchronization between the source and destination.
- Line-coding techniques can be categorized as unipolar, polar, and bipolar. NRZ, RZ, and Manchester line coding techniques are the most popular polar-coding techniques.
- Bipolar AMI-RZ is widely used bipolar line-coding technique in several applications related to baseband digital transmission.

LO 2.2

- Intersymbol Interference (ISI) is spreading of a pulse beyond its allotted time period (i.e., pulse width) and tends to interfere with adjacent pulses, due to the dispersive nature of a communications channel.
- Due to occurrence of ISI, an error will be introduced in the decision-making device at the receiver output.
- An eye diagram, also known as an eye pattern, is a practical technique for determining the severity of the degradations introduced by intersymbol interference and channel noise into the digital pulses in baseband digital transmission systems.

LO 2.3

- The impact of ISI can be controlled by a properly shaped bandlimited Nyquist pulse prior to transmission.
- This pulse waveform of the type sinc $(x) = [(\sin x)/x]$ is popularly known as Nyquist pulse shaping.
- Equalization helps the demodulator to recover a baseband pulse with the best possible signal-to-noise ratio, free of any intersymbol interference (ISI).

LO 2.4

- Multiplexing is simultaneous transmission of multiple baseband signals across a single data link.
- Time-Division Multiplexing (TDM) technique is used for digital signals, either digital data or digitized analog signals.
- In TDM, digital signals from *n* sources are interleaved with one another, forming a frame of data bits.
- Framing bits allow TDM multiplexer to synchronize properly.

LO 2.5

- Digital signal (DS) is a hierarchy of TDM signals.
- T carrier lines (T1 to T4) are the implementation of DS services. A T1 line consists of 24 voice channels.
- E carrier lines (E1 to E5) is based on the basic 32-line-slots (30-voice channels of 64 kbps rate and 2 control channels) E-lines (European lines).

Hands-on Projects

2.1 Imagine you wish to transmit 10-digit mobile number over a wired digital communication medium. First, you will need to convert your mobile number from its decimal (base 10) representation into an equivalent binary (base 2) representation. Using clearly labeled diagrams, show an encoding of your mobile number using (a) NRZ-L, (b) bipolar-RZ-AMI, (c) Manchester, and (d) differential Manchester line code signaling formats.

2.2 Devise a TDM system to time-division multiplex two channels. Verify the operation of the system by sending two distinct short messages (data packets) at a very slow data rate so as to check that they are received correctly.

2.3 Study MAXIM's DS34T108DK Evaluation Kit for 1- to 8-Port TDM-over-Packet ICs and configure it to be used as T1/E1 Interface.

2.4 Search through the Web and study the data sheet of T1/E1 line interface IC CS61535A manufactured by CIRRUS LOGIC. Develop an application for line interface or SONET.

Objective-Type Questions

For Interactive Quiz with answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/81

2.1 Identify the technique for creating digital database of real signals.
(a) Pulse amplitude modulation
(b) Pulse code modulation
(c) Binary conversion
(d) Manchester coding

2.2 Choose the fundamental frequency of alternating 1s and 0s of a UP-NRZ signal if a UP-NRZ serial data sequence is transmitted at data rate of 1,000 bps.
(a) 100 Hz
(b) 500 Hz
(c) 2000 Hz
(d) 4000 Hz

2.3 Select the most commonly used line-coding technique with the best overall desirable line properties.
(a) BP-NRZ
(b) BP-RZ
(c) BP-AMI-RZ
(d) UP-RZ

2.4 Check that the fundamental frequency of UP-RZ line-coding signaling format is ________ that of the UP-NRZ line-coding signaling format.
(a) same as
(b) half
(c) twice
(d) four times

2.5 *Statement I:* The non-return-to-zero (NRZ) line coding causes a binary logic 1 data bit to be returned to a 0 V at the midpoint of the bit interval.
Statement II: The return-to-zero (RZ) line coding causes a binary logic 1 data bit to be returned to a 0 V at the midpoint of the bit interval.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

2.6 Exemplify that the information which can be obtained from an eye diagram include(s) the effects of
(a) noise
(b) Intersymbol interference (ISI)
(c) jitter
(d) noise, ISI, and jitter

2.7 Find the range of roll-off factor of the raised-cosine filter used for pulse shaping in baseband data transmission system.
(a) $0 \le \alpha \le 0.5$
(b) $0 \le \alpha \le 1$
(c) $0 \le \alpha \le \infty$
(d) $1 \le \alpha \le \infty$

2.8 *Statement I:* An equalizer is a specially designed filter which is located between the demodulator and decision making device at the receiver.
Statement II: Equalization helps the demodulator to recover a baseband pulse with the best possible signal-to-noise ratio, free of any intersymbol interference (ISI).
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

2.9 The important functions that must be performed by a digital multiplexer include one of the following.
(a) To establish a bit.
(b) To assign a number of unique bit slots within the frame to any one input.
(c) To insert control bits for frame identification and synchronization.
(d) Not to make any allowance for any variations of the input bit rate by bit stuffing or word stuffing in multiplexed data stream.

2.10 A T1 carrier line handle ________ voice-grade PCM channels, whereas an E1 line handles _______ voice-grade PCM channels.
(a) 32; 24
(b) 30; 24
(c) 24; 30
(d) 24; 32

Short-Answer-Type Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/45

2.1 What is the importance of a baseband digital transmission system? Give an account of performance determining metrics (figure of merit) of a digital communications channel.

2.2 Bring out clearly similarities and dissimilarities between Manchester and differential Manchester line-coding techniques, considering a binary data sequence of 0 1 0 0 1 1 1 0 1 0.

2.3 Distinguish between waveform coding and line coding. Why are line codes necessary for baseband transmission of signals?

2.4 Map advantages and disadvantages of non-return-to-zero (NRZ) and return-to-zero (RZ) line coding techniques.

2.5 Identify at least four primary causes of occurrence of Intersymbol Interference (ISI) in baseband digital data transmission system.

2.6 Discuss critically the effects of intersymbol interference on the performance of digital transmission.

2.7 List the benefits and drawbacks of Nyquist pulse shaping.

2.8 Define roll-off factor (α) of a raised-cosine filter. Draw its frequency response for $\alpha = 0, 0.5, 1$.

2.9 Determine the bandwidth expansion factor for the PCM signal with rectangular pulses, whose basic bandwidth is 4 kHz and the number of bits per sample is 8. If sync pulses are used in place of rectangular pulses, what is the bandwidth expansion factor?

2.10 Equalization is used to minimize intersymbol interference arising due to transmission of digital data in time-dispersive channels. Hypothesize three essential requirements of a good equalizer.

2.11 TDM has several inherent advantages and some disadvantages. Outline them.

2.12 Pulse stuffing and word stuffing are two techniques that enable to multiplex asynchronously sampled signals. Articulate the effect of them in TDM.

2.13 A digital carrier system is a communications system that uses digital pulses rather than analog signals to encode information. Highlight key features of the T1 digital carrier system.

2.14 Compare the number of sync bits used in standard T1, T2, T3, and T4 lines.

2.15 Specify the bandwidth of a T1 and an E1 carrier lines.

Discussion Questions

2.1 The synchronous data sequence with RZ line-coding technique has transitions of state for consecutive logic 1 data bits that are not present in NRZ line coding. Paraphrase the specific benefit achieved with RZ line coding. **[LO 2.1]**

2.2 There are several ways to reduce the effects of intersymbol interference or pulse spreading in a baseband digital communication system. Carry out concise study of various practical methods that can be applied effectively. **[LO 2.2]**

2.3 If a random pattern of ones and zeros is transmitted and the signal is applied to an analog oscilloscope sweeping at the bit rate, the pattern observed on CRO display resembles an human eye, known as eye diagram. Under what situation, would the eye diagram begin to close? **[LO 2.2]**

2.4 The intersymbol interference causes distortion that can be minimized by incorporating a suitable equalizer. Classify and explain equalizers based on structure and type of algorithms used for decoding the data at the digital receiver. **[LO 2.3]**

2.5 In the synchronous digital multiplexer, a master clock governs all the data sources, with the provisions to distribute the master clock signal to all the sources. Explain the concept of synchronous TDM with a suitable example. **[LO 2.4]**

2.6 It is essential to make available the data-encoded bits as well as synchronizing or timing information at the receiver in digital carrier systems. Show a PCM-TDM frame using channel associated signaling in a T1 line. **[LO 2.5]**

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/104

Problems

2.1 Determine the transmission bandwidth for unipolar NRZ signaling format if the pulse width is 10 μs.

2.2 Calculate the bit duration if the transmission bandwidth of bipolar RZ line format is 1 kHz.

2.3 Represent the given data sequence 1 0 1 1 0 0 1 0 with the resulting waveform using unipolar RZ, BP-AMI-RZ, and Manchester line coding techniques.

2.4 Consider a binary data sequence with a long sequence of binary 1s followed by a single binary 0, and then a long sequence of binary 1s. Draw the waveforms for the given binary data sequence, using bipolar NRZ and bipolar AMI RZ in order to understand the difference between these two line codes.

2.5 Binary data at 9600 bps are transmitted using 8-ary PAM modulation with a system using a raised cosine roll-off filter characteristics having $\alpha = 0.5$. Determine the frequency response of the system.

2.6 A voice signal in the frequency range 300 Hz to 3400 Hz is sampled at 8000 samples/s.

(a) If we transmit these samples directly as PAM pulses, then evaluate the minimum system bandwidth required for the detection of PAM with no ISI and with a filter roll-off factor of 1.

(b) If we first convert each sample to a PCM format and use binary PCM waveform for transmission, then using the same roll-off, evaluate the minimum bandwidth required for the detection of binary PCM waveform if the samples are quantized to 16-levels.

(c) Repeat the part (b) using 128 quantization levels.

(d) Comment on the results obtained.

For answer keys, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/110

2.7 Four independent bandlimited information data signals have bandwidths of 100 Hz, 100 Hz, 200 Hz, and 400 Hz, respectively. Each signal is sampled at Nyquist rate, and the samples are time-division multiplexed and transmitted. Determine the transmitted sample rate in Hz.

2.8 Plot the output waveform of a baseband quaternary (4-level) PAM system for the input binary data sequence 0 0 1 0 1 1 0 1 1 1.

2.9 Twenty-four voice-band signals (each having f_m = 3.4 kHz), are sampled uniformly using flat-top samples with 1 μs pulse duration, and then time-division multiplexed. The Mux operation includes provision for synchronization by adding an extra pulse of 1 μs duration. Assuming a sampling rate of 8 kHz, calculate the spacing between successive pulses of multiplexed signal.

2.10 Consider time-division multiplexing of 5 PAM signals with sampling time of 1 ms. If the width of each sample pulse is 150 μs, find guard time. If it is required to maintain the same guard time to avoid interference between samples, design new pulse width to transmit 10 PAM signals in 1 ms duration.

2.11 Design a TDM system (i.e., the size of one TDM frame, the frame rate, duration of one TDM frame, and the combined bit rate for the digital communication link) for multiplexing 16 similar digital sources transmitting data at 6400 bps rate. Assume the number of bits per channel as 8.

2.12 We know that the number of time slots in E1 TDM frame is 32, having 8 bits per time slot. If number of frames transmitted in one second is 8000, then solve for total number of bits for frame and data rate.

Critical Thinking Questions

2.1 Consider a random binary sequence in which bits are statistically independent and equally likely. It is yet to be ascertained about type of line code to be employed so as to ensure reliable clock synchronization at the sender and receiver. Analyze at least three different types of line codes which can meet this objective. **[LO 2.1]**

2.2 What functions can be carried out directly using bipolar alternate mark inversion line code that cannot be achieved with unipolar type data formats? Implement BP-AMI-RZ line code and unipolar NRZ line code for an example binary sequence and compare their advantages and disadvantages. **[LO 2.1]**

2.3 In baseband digital transmission, Intersymbol Interference (ISI) arises due to the dispersive nature of a communications channel. Evolve a model depicting baseband binary data transmission system and derive an expression for ISI. **[LO 2.2]**

2.4 A matched filter is an optimum detector of a known pulse in the presence of additive white noise. Compare the important characteristics of matched filters with that of conventional filters, highlighting their utility in modern digital communication receivers. **[LO 2.3]**

2.5 T1 digital carrier lines are susceptible to numerous types of problems and impairments during the transmission of PCM-encoded digitized and time-division multiplexed data over the twisted-pair lines. Review various types of line impairments that are specific to T1 lines. **[LO 2.5]**

References for Further Reading

[Cou00] Couch, L; *Digital and Analog Communication Systems.* Upper Saddle River, NJ: Prentice Hall, 2000.

[SI12] Singal, T L; *Analog and Digital Communications.* Tata McGraw-Hill, 2012.

3 Digital Modulation and Demodulation

Learning Objectives

To master digital modulation and demodulation, you must cover these milestones:

LO 3.1 Analyze binary digital modulation techniques such as ASK, FSK and PSK

LO 3.2 Describe *M*-ary digital modulation techniques including QPSK, Offset QPSK, π/4-QPSK, MSK, GMSK, and QAM

LO 3.3 Understand coherent and non-coherent detection including carrier recovery methods

LO 3.4 Develop performance-determining criteria in terms of bandwidth efficiency and error probabilities of various digital modulation techniques

Essence of Digital Modulation

Baseband signals are suitable for transmission over a pair of wires and coaxial cables because they have significant power at low frequencies. However, baseband signals cannot be transmitted over a radio or satellite link using wireless communication. One of the main reasons for this is because of impractical large antennas required for efficient radiation of the low-frequency spectrum of baseband signals. Hence, for wireless communication systems, the spectrum of the baseband signal must be shifted to a high-frequency band. We know that the spectrum of a baseband signal can be shifted to a higher frequency by modulating a high-frequency sinusoidal carrier signal. Digital modulation is a simple case of transmitting digital data (information) using analog carrier signals. Digital modulation systems offer several outstanding advantages over traditional analog modulation systems such as easier and faster signal processing as well as multiplexing, and greater noise immunity and robustness to channel impairments. In digital modulation techniques, the modulated signal occupies a bandwidth centered on the carrier frequency.

INTRODUCTION

Bandpass Transmission of Digital Signals

When the baseband signal, also known as modulating signal in this case, consists of binary or multi-level digital data; the process of carrier modulation is known as *bandpass digital modulation*. Thus, **digital modulation** is the process in which amplitude, frequency, or phase characteristics of a sinusoidal carrier signal is varied in accordance with the instantaneous value of the modulating signal consisting of binary or multi-level digital data. In *Amplitude Shift Keying (ASK)*, the amplitude of the analog carrier signal is switched from one value to the other in proportion to the instantaneous value of the digital input signal. And when the frequency or the phase of the carrier is switched from one value to other (keeping its amplitude constant) in proportion to the instantaneous value of the digital input signal, we obtain *Frequency Shift Keying (FSK)*, or *Phase Shift Keying (PSK)*, respectively. If both the amplitude and phase-angle of the carrier signal are varied in proportion to the instantaneous value of the digital input signal, a hybrid digitally modulated signal known as *Quadrature Amplitude Modulation (QAM)* is obtained. Just like baseband signalling, we can have binary or *M-ary digital modulation techniques* in bandpass digital transmission. In *M-ary ASK*, M ($= 2^n$) possible signals will have M distinct carrier amplitude levels. In *M-ary FSK*, M possible signals will have M different carrier frequencies but having the same amplitude levels, and in M-ary PSK, they will all have the same amplitude and frequency but M different phase angles. M-ary bandpass signaling enables a given speed of data transmission to be achieved using a smaller bandwidth.

Geometric interpretation, also known as *signal constellation,* is a popular way of representing binary as well as M-ary shift keying signals in which the set of message points (in a signal space) corresponding to the set of all the transmitted signals. Based on whether or not a locally generated carrier signal that is in perfect frequency and phase synchronism with the carrier signal used in the transmitter, is used in the receiver for detecting the data from the received modulated signals, bandpass digital modulation techniques may be classified as non-coherent and coherent types. ASK and FSK signals may be detected either non-coherently, or coherently whereas PSK, M-ary FSK and QAM signals are detected coherently only. PSK-derived signals produce *orthogonal signaling* and the phase information of the received signal is used in the detection process to improve the noise performance.

The Power Spectral Density (PSD) provide an insight into the bandwidth of the digital modulated signals and also the possibility of interference with adjacent channels when multiplexing is used. However, we need not derive the spectra of the digital modulated signals if we know the PSD of the baseband signal. In order to analyze how fast we can transmit signals within a given channel bandwidth, the performance parameter `*bandwidth efficiency*' (the ratio of signalling rate to the bandwidth required) is more important. It depends on the type of digital modulation, and for a given modulation technique it depends on whether it is binary or M-ary signalling. For a specified transmission data rate, as M increases, the required bandwidth decreases and so the bandwidth efficiency improves. But we cannot increase the value of M indefinitely as the transmission power requirement also increases in order to maintain a specified probability of error.

A PRIMER

Why Passband Modulation?

Since the early days of electronics, as advances in technology were made, the boundaries of both local and global communications began eroding, resulting in a world that is smaller and hence more easily accessible for the sharing of knowledge and information. The pioneered work by Graham Bell and Marconi formed the cornerstone of the information age exists today and paved the way for the future of telecommunications. Traditionally, local communications was done over wires, as this presented a cost-effective way of ensuring a reliable exchange of information. For long-distance communications, transmission of information over radio waves was needed. Although this was convenient from a hardware point of view, radio wave transmission raised doubts over the corruption of the information and was often dependent on high-power transmitters to overcome environmental conditions, large buildings, and interference from other electromagnetic sources.

We have earlier noted that baseband signal in the form of digital data cannot be transmitted over analog or wireless channel because the digital encoded data is in the form of sequences of pulses. In the frequency domain, these pulses contain frequency range starting from 0 to infinity. This means that these pulses can be transmitted without any distortion over a communication channel having an infinite bandwidth. This is not practical! All types of communication channels such as telephone lines, optical fiber, wireless, or satellite links, are band-limited, and represent bandpass channels. Hence, finite bandwidth of analog channels is the major limiting factor of transmitting almost infinite bandwidth of digital data. So, for transmitting digital data over bandpass channels, the digital data has to be first converted into a suitable form which is compatible to band-limited communication channel. So it is required to modulate the digital data on a sinusoidal carrier signal for transmission over a bandpass channel.

Long-haul communication over a wireless link requires modulation to shift the baseband signal spectrum to higher frequencies at which efficient radiation of electromagnetic signal is possible using antennas of practical dimensions. Communication that uses modulation to shift the frequency spectrum of a baseband signal is known as *bandpass communication* or *carrier communication*. As the transmission bandwidth changes with the use of modulation, it results in better performance against interference. In general, the process of modulation provides ease of RF transmission and frequency division multiplexing. Modulation can be analog or digital. For example, traditional communication systems such as AM/FM radio and NTSC television broadcast signals are based on analog modulation, whereas advanced wireless systems such as 2G/3G cellular and HDTV are digital modulation systems.

In this chapter...

LO 3.1 >
- We analyze binary digital modulation techniques (ASK, FSK, PSK) for their modulated waveforms, power spectral density representations, transmission bandwidth, functional block schematics of modulators and demodulators.

LO 3.2 >
- Then, we describe multi-level digital modulation techniques such as quadrature phase shift keying (QPSK) and their variants, minimum shift keying (MSK) and their derivatives including Gaussian MSK, and M-ary digital modulation.

LO 3.3 >
- The discussion is carried forward to understand the fundamentals of coherent and non-coherent detection including carrier recovery methods.

LO 3.4 >
- Finally, we try to develop performance comparison criteria of various digital modulation techniques in terms of bandwidth efficiency.

LO 3.1

3.1 BINARY DIGITAL MODULATION TECHNIQUES

Different Types

As stated earlier, *digital modulation* is the process in which amplitude, frequency, or phase characteristics of a sinusoidal carrier signal is varied in accordance with the instantaneous value of the modulating signal consisting of binary data (0s and 1s) Accordingly, we obtain Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), and Phase Shift Keying (PSK). In ASK, a finite number of amplitude levels is used for modulation. Similarly, for FSK, a finite number of frequencies, and phase for PSK. In the case of QAM, as in-phase signal (the I-signal, for example a sine waveform) a quadrature-phase signal (the *Q*-signal, for example a cosine waveform) are amplitude modulated with a finite number of amplitudes. The resulting signal is the combination of ASK and PSK, with a finite number of at least two amplitude levels, and a finite number of at least two phases in QAM. Binary digital modulation techniques such as BASK, BFSK and BPSK are discussed in this section whereas QAM is discussed in Section 3.2.6.

IMPORTANT!

When the baseband signal, also known as modulating signal, consists of binary data; the process of carrier modulation is known as digital bandpass modulation, or simply digital modulation. For digital signal transmission the term 'keying', means switching is used instead of modulation because of the nature of digital data (having discrete levels rather than continuous)—hence the names as ASK, FSK, and PSK.

3.1.1 Amplitude Shift Keying (ASK)

Define

Amplitude Shift Keying (ASK) is a form of digital bandpass modulation technique in which the amplitude of the analog sinusoidal carrier signal is varied to represent the input binary data. The frequency and phase of the carrier signal remains unchanged.

Mathematical Expression

Consider the digital baseband signal (i.e., the modulating signal) is represented by unipolar-NRZ(L) line code, i.e., $v_m(t) = 1$ for binary symbol '1' and $v_m(t) = 0$ for binary symbol '0' for entire bit duration T_b, in the form of on-off signal.[1]

[1] Unipolar NRZ(L) rectangular pulse waveform has been used here for the sake of illustration only. Of course, we can modulate polar, bipolar, or any other line-coded baseband signal $v_m(t)$. In practice, the baseband signal may utilize any spectrally shaped pulse to eliminate ISI which will generate an ASK signal that does not have a constant amplitude during the transmission of binary logic '1'.

Let the carrier signal is represented by $\sin(2\pi f_c t)$, where f_c is its frequency in Hz.

As per the definition of ASK, when the amplitude of the carrier signal is varied in proportion to the modulating signal $v_m(t)$, we can write the ASK modulated signal as

$$v_{\text{ASK}}(t) = v_m(t)\sin(2\pi f_c t)$$

In other words, the modulated binary ASK signal can be expressed as

$$\boxed{v_{\text{ASK}}(t) = \begin{Bmatrix} \sin(2\pi f_c t) & \text{for binary '1'} \\ 0 & \text{for binary '0'} \end{Bmatrix}}$$

Using the definition and mathematical expression for ASK signal, Figure 3.1.1 shows the ASK signal waveform for the given binary data 1 0 1 1 0 1 0 1 1 1.

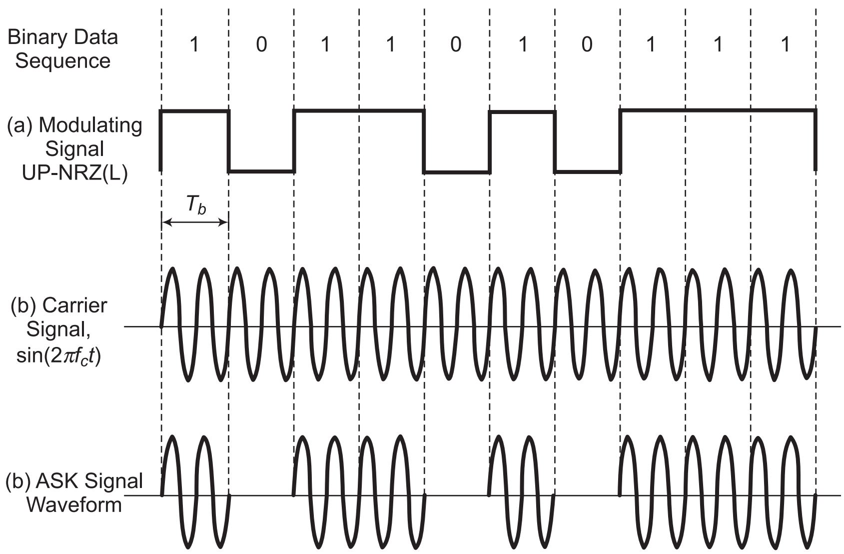


Figure 3.1.1 ASK Signal Waveform

Question

Why is binary amplitude shift keying (BASK) also known as on-off keying (OOK)?

Answer

We see that the modulated ASK signal is either the carrier signal (for binary 1) or no carrier signal (for binary 0). That simply means that the carrier signal is either ON or OFF, depending on whether the input binary data is either binary 1 or 0, respectively. This is the reason that sometimes *binary amplitude-shift keying* is also referred to as *on-off keying (OOK)*.

Interpretation of Binary ASK Signal Waveform

For every change in the input binary data stream, either from logic 1 to logic 0 or vice-versa, there is corresponding one change in the ASK signal waveform.

- For the entire duration of the input binary data logic 1 (high), the output ASK signal is a constant-amplitude, constant-frequency sinusoidal carrier signal.
- For the entire duration of the input binary data logic 0 (low), the output ASK signal is zero (no carrier signal).

Bit Rate and Baud

We know that the *bit rate* is the reciprocal of the bit duration (T_b). That is,

bit rate $$f_b = 1/T_b$$

The time of one signaling element (T_s) is the reciprocal of the baud (*Baud* refers to the rate of change of a signal element which may represent several information bits, expressed as symbols per second).

In case of binary data, the time of one bit (T_b) is equal to the time of one signaling element (T_s).

Therefore, the rate of change of the ASK signal (expressed in baud) is the same as the rate of change of the binary input data (expressed in bps).

This means **baud rate is same as bit rate in ASK.**

Bandwidth of ASK Signal

The bandwidth of ASK signal depends on the bit rate of input data.

In general, the transmission bandwidth for ASK signal is given by

$$B_{\text{ASK}} = (1 + r)f_b$$

where r (typically $0 < r < 1$) is related to the modulation process by which the signal is filtered to establish a bandwidth for transmission; and f_b is the bit rate.

Thus, the transmission bandwidth of ASK signal is directly related to the input bit rate.

The minimum bandwidth of ASK signal corresponds to $r = 0$, that is, $B_{\text{ASK}}(\text{minimum}) = f_b$, and its maximum bandwidth corresponds to $r = 1$, that is, $B_{\text{ASK}}(\text{maximum}) = 2f_b$.

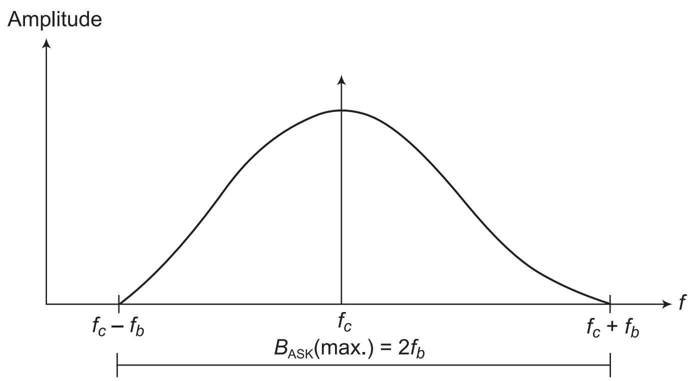


Figure 3.1.2 ASK Signal Bandwidth

PSD of ASK Signal

The ASK signal is basically the product of the binary digital data input and the sinusoidal carrier signal. In general, for an appropriately chosen carrier frequency, modulation causes a shift in the baseband signal power spectral density (PSD). Now, in the ASK signal considered here, the modulating signal is an on-off signal (using Unipolar NRZ pulse having pulse width as T_b). Hence, the PSD of the ASK signal is the same as that of the baseband binary data on-off signal but shifted in the frequency domain by $\pm f_c$, where f_c is the carrier signal frequency.

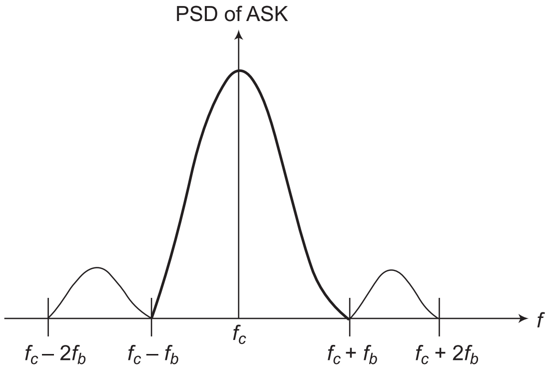


Figure 3.1.3 PSD of ASK Signal

Interpretation of PSD of ASK Signal Waveform It is seen that the PSD of binary ASK[2] or on-off keying signal has no discrete components except at dc. Therefore, the ASK spectrum has discrete component only at f_c. The spectrum of ASK signal shows that it has an infinite bandwidth. However, the bandwidth of ASK signal can be reduced by using smooth pulse waveform instead of rectangular pulse waveform at the binary digital data input signal.

ASK Modulator

We know that the ASK modulated signal is simply the multiplication of the modulating signal and the carrier signal, $v_{ASK}(t) = v_m(t)\sin(2\pi f_c t)$. Therefore, an ASK signal can be generated by applying the input binary data $v_m(t)$ and the sinusoidal carrier signal $\sin(2\pi f_c t)$ to the two inputs of a balanced modulator.

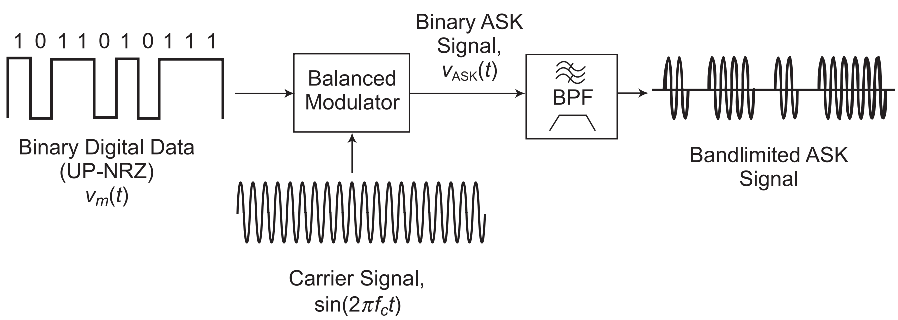


Figure 3.1.4 Functional Block Schematic of ASK Modulator

Functional Description of ASK Modulator

- The baseband signal is a binary digital data which is a unipolar NRZ(L) line-coded signal which acts as the modulating signal input to the balanced modulator.
- The other input to the balanced modulator is the locally generated analog carrier signal.
- The output of balanced modulator is the product of the UP-NRZ signal and the sinusoidal carrier signal, i.e., $v_{ASK}(t) = v_m(t)\sin(2\pi f_c t)$.
- The output is passed through a bandpass filter in order to contain the frequency spectrum of the ASK signal.

[2] ASK is the simplest form of digital modulation scheme. It is simple to design, easy to generate and detect. ASK requires low transmission bandwidth and very less power to transmit the binary data. However, ASK signal it is susceptible to sudden amplitude variations due to noise and interference.

ASK Demodulation or Detection

The process of recovering the signal from the modulated signal (i.e., translating the spectrum to its original position) is referred to as *demodulation* or *detection*.

As seen from the PSD of ASK signal, the ASK modulation translates or shifts the frequency spectrum to the left and right by f_c (i.e., at $+f_c$ and $-f_c$). To recover the original signal from the modulated signal, it is necessary to retranslate the spectrum to its original position.

Coherent Binary ASK Detection

If we observe it closely, the process of demodulation is almost identical to modulation, i.e., multiplication of the incoming modulated signal by identical carrier signal as used at the modulator, followed by an integrator (low-pass filter) and a decision-device to determine whether the transmitted data was binary 1 or 0. Therefore, for demodulation, the receiver must generate a carrier signal in frequency and phase synchronism with the incoming carrier. These type of demodulators are synonymously called synchronous or coherent detectors. Thus, *coherent binary ASK detector* comprises of a balanced modulator, followed by an integrator and a decision-making device.

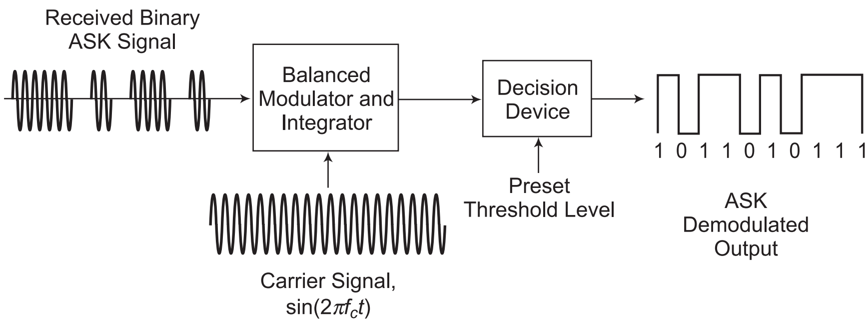


Figure 3.1.5 Functional Block Schematic of Coherent Binary ASK Detector

Functional Description of Coherent ASK Detector

- The received binary ASK signal and a sinusoidal carrier signal generated by a local oscillator are applied to the balanced modulator.
- The integrator operates on the output of the balanced modulator for successive bit intervals, T_b and essentially performs a low-pass filtering action.
- Its output is applied to a decision-making device which compares it with a preset threshold level.
- It makes a decision in favour of the symbol 1 when the threshold level is exceeded, otherwise 0.
- Thus, in coherent ASK detector, the local carrier signal is in perfect synchronization with the corresponding carrier signal as used in ASK modulator on transmitter side.
- This means that the frequency and phase of the locally generated carrier signal is same as those of carrier signal used in ASK modulator.

Synchronization in Coherent ASK Detector

In fact, there are two forms of synchronization required for the operation of coherent or synchronous ASK demodulator—phase synchronization and timing synchronization.

- *Phase synchronization* ensures that the carrier signal generated locally in a coherent ASK demodulator is locked in phase with respect to the one that is used in ASK modulator.
- *Timing synchronization* enables proper timing of the decision-making operation in the ASK demodulator with respect to the switching instants, that is switching between 1 and 0 as in the original binary data.

Thus, we see that the coherent detector requires more elaborate equipment. It has superior performance, especially when the signal-to-noise power ratio (S/N) is low.[3]

Non-Coherent Binary ASK Detection

ASK can be demodulated both coherently (for synchronous detection) as described in the previous section or non-coherently (for envelope detection). In non-coherent demodulation method of digitally modulated signals, knowledge of the carrier signal's phase is not required. Therefore, *non-coherent binary ASK detector* does not require a phase-coherent local oscillator.

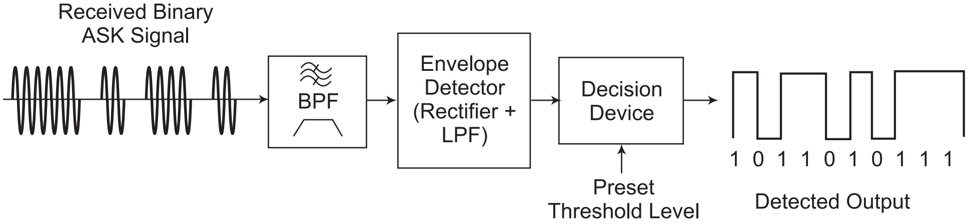


Figure 3.1.6 Functional Block Schematic of Non-coherent Binary ASK Detector

Functional Description of Non-Coherent ASK Detector

- It involves rectification and low-pass filtering as part of the envelope detector.
- The output is followed by switching operation at bit period, T_b.
- The signal is passed through a decision-making device with preset threshold which determines whether the received symbol is 1 or 0.

IMPORTANT!

The design of non-coherent binary ASK detector is quite simple but its error performance is poor as compared to that offered by coherent binary ASK detector. However, for higher S/N power ratio, the non-coherent detector (basically the envelope detector) performs almost as well as the coherent detector.[4]

SOLVED EXAMPLE 3.1.1 ASK Signal Bandwidth

A 10 kbps binary data signal is required to be transmitted using ASK digital modulation technique. Determine the minimum transmission bandwidth and baud rate necessary.

Solution We know that in ASK digital modulation technique.

The minimum bandwidth, $B_{\text{ASK}}(\text{minimum}) = f_b$

where f_b is the binary information bit rate.

For given f_b = 10 kbps, $B_{\text{ASK}}(\text{minimum})$ = 10 kHz **Ans.**

We also know that in binary ASK digital modulation technique, 1 symbol = 1 bit

[3] When we use a carrier of exactly the same frequency (and phase) as the carrier used for modulation, the method of recovering the baseband signal is called synchronous detection, or coherent detection. Thus, for demodulation, we need to generate a local carrier at the receiver in frequency and phase coherence (synchronism) with the carrier used at the modulator.

[4] ASK has been mostly used for very low-speed data rate (up to 1200 bps) requirements on voice-grade lines in telemetry applications. In practice, ASK as an on-off scheme is commonly used today in optical fiber communications in the form of laser-intensity modulation to transmit digital data. It is also used in wireless infrared transmissions using a directed beam or diffuse light in wireless LAN applications.

For given f_b = 10 kbps, baud rate, f_b = 10 k symbols/sec. **Ans.**[5]

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 3.1.2 A 1200 bps binary data signal is required to be transmitted using ASK digital modulation technique. Determine the minimum transmission bandwidth and necessary baud rate.

3.1.2 Frequency Shift Keying (FSK)

Define

Frequency shift keying (FSK) is a form of digital modulation technique in which the frequency of the analog sinusoidal carrier signal f_c is varied to represent the input digital data. However, the amplitude of the modulated carrier signal remains constant.

Mathematical Expression

The FSK signal may be viewed as a sum of two interleaved ASK signals (but with constant amplitude), one with a carrier frequency f_{c0} for transmitting binary '0', and the other with another carrier frequency f_{c1} for transmitting binary '1'.

In other words, the FSK signal is a superposition of two ASK signals with different carrier frequencies but same amplitudes. Thus, binary FSK (BFSK) signal can be expressed as

$$\boxed{v_{\text{FSK}}(t) = \begin{Bmatrix} \sin(2\pi f_{c0}t) & \text{for binary 0} \\ \sin(2\pi f_{c1}t) & \text{for binary 1} \end{Bmatrix}}$$

where f_{c0} and f_{c1} are typically offset frequencies from the carrier frequency f_c by equal but opposite values.

Using the definition and mathematical expression for binary FSK signal. Figure 3.1.7 shows the FSK signal waveform for the given binary data 1 0 1 1 0 1 0 1 1 1.

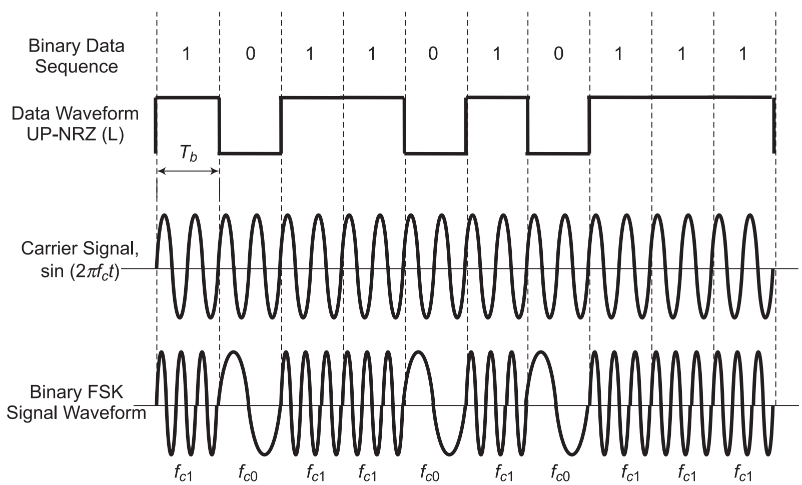


Figure 3.1.7 Binary FSK Signal Waveform

[5] There is a natural and clear connection between digital and analog bandpass (carrier) modulation techniques—ASK and AM respectively, because the message information (digital in ASK and analog in AM) is directly reflected in the varying amplitude of the modulated signals (the carrier signal being sinusoid in both cases). Because of its non-negative amplitude, ASK is essentially an AM signal with modulation index of unity.

Interpretation of Binary FSK Signal Waveform

- When the input binary data changes from a binary logic 1 to logic 0 and vice versa, the BFSK output signal frequency shifts from f_{c1} to f_{c0} and vice versa ($f_{c1} > f_{c0}$).
- It can be seen that when the input data changes from a logic 1 to 0 or vice versa, there is an abrupt phase discontinuity in the analog binary FSK signal.
- Continuous-phase frequency-shift keying (CP-FSK) is a binary FSK except that two frequencies f_{c1} and f_{c0} are separated from the center frequency f_c by an exact multiple of one-half the bit rate f_b.
- This ensures a smooth phase transition in the BFSK signal when it changes from f_{c1} to f_{c0} or vice versa.

Bit Rate and Baud

It is seen that the time duration of one bit (T_b) is the same as the symbol time duration (T_s) of the binary FSK signal output. Thus, the bit duration is identical to the time of an FSK signaling element. Hence, the *bit rate* (f_b) equals the *baud*.

Bandwidth of FSK Signal

The minimum bandwidth for binary FSK signal is given as

$$B_{\text{BFSK}} = \left|(f_{c1} + f_b) - (f_{c0} - f_b)\right|$$

$$\boxed{B_{\text{BFSK}} = \left|(f_{c1} - f_{c0})\right| + 2f_b}$$

PSD of FSK Signal

Although there are only two carrier frequencies (f_{c0} and f_{c1}) at the binary FSK signal output, the process of modulation produces a composite signal which is a combination of many signals, each with a different frequency. If the difference ($f_{c1} - f_{c0}$) is chosen to be equal to $2f_b$ then bandwidth of binary FSK signal is $4f_b$. In this case, the interference between the spectrum at f_{c0} and f_{c1} is not significant. We have also shown that the FSK signal may be viewed as a sum of two interleaved ASK signals using the pulse width as T_b. Hence, the PSD of FSK is the sum of PSD of two ASK signals at frequencies f_{c0} and f_{c1}.

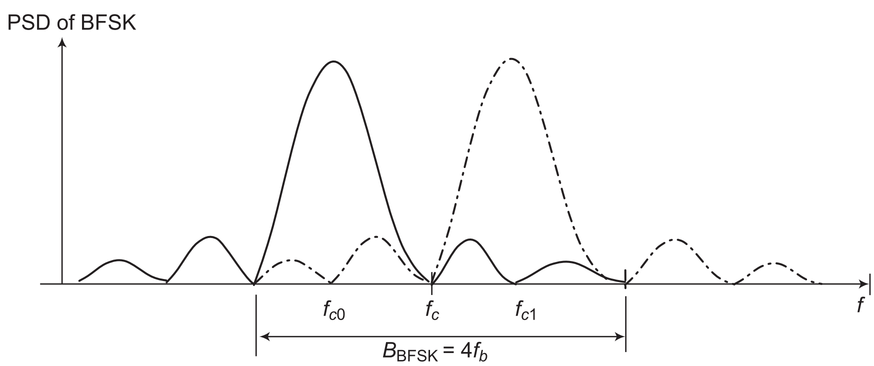


Figure 3.1.8 PSD of Binary FSK Signal

Interpretation of PSD of FSK Signal Waveform It can be shown that by properly choosing the values of f_{c0} and f_{c1} and by maintaining phase continuity during frequency switching, discrete components can be eliminated at f_{c0} and f_{c1}. Thus, no discrete components appear in the spectrum of FSK signal.[6]

[6] It is important to note that the bandwidth of FSK signal is higher than that of ASK signal. However, FSK is less susceptible to errors and thus has better noise immunity than ASK.

FSK Modulator

Recall

The binary FSK can be viewed as two interleaved ASK signals with carrier frequencies f_{c0} and f_{c1}, respectively. Therefore, binary FSK modulator basically comprises of two balanced modulators.

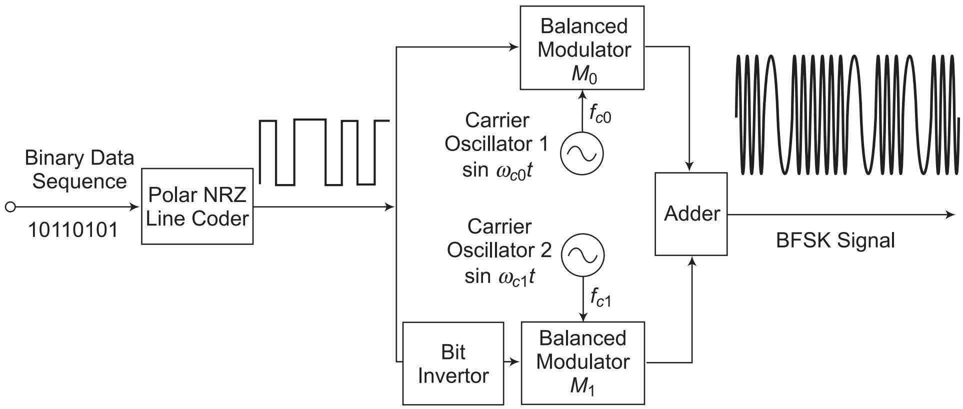


Figure 3.1.9 Functional Block Schematic of FSK Modulator

Functional Description of FSK Modulator

- The input binary data is passed through a polar NRZ line coder.
- Its output is applied to two independent balanced modulators M_0 and M_1.
- The other inputs to balanced modulators M_0 and M_1 are carrier oscillator signals at f_{c0} and f_{c1}, respectively.
- It may be noted here that the frequencies f_{c0} and f_{c1} are typically offset frequencies from the carrier frequency f_c by equal but opposite values.
- The outputs of balanced modulators are added together in a linear adder circuit.
- The resultant binary FSK signal will either have a frequency signal f_{c0} or f_{c1}, as shown.

Coherent Binary FSK Detection

An FSK signal can also be detected coherently by generating two carrier frequencies f_{c0} and f_{c1} in synchronization with the modulation carriers respectively. These reference frequencies are then applied to two balanced modulators, to demodulate the signal received.

Functional Description of Coherent FSK Detector (Refer Fig. 3.1.10)

- The received BFSK signal is applied to two correlators 1 and 2, each of which comprises of a balanced modulator, followed by an integrator (low-pass filter).
- As mentioned, the other inputs to balanced modulators are carrier oscillator signals at f_{c0} and f_{c1} respectively.
- The outputs of two correlators are then subtracted which is then applied to a decision device (comparator).
- The decision device compares the input signal with a preset threshold level, usually zero volt.
- If its input signal level is greater than 0 volt, the detected output is the binary symbol 1.
- If it is less than 0 volt, the detected output is the binary symbol 0.
- Since the two transmitted frequencies are not generally continuous, it is not practical to reproduce a local reference that is coherent with both of them.

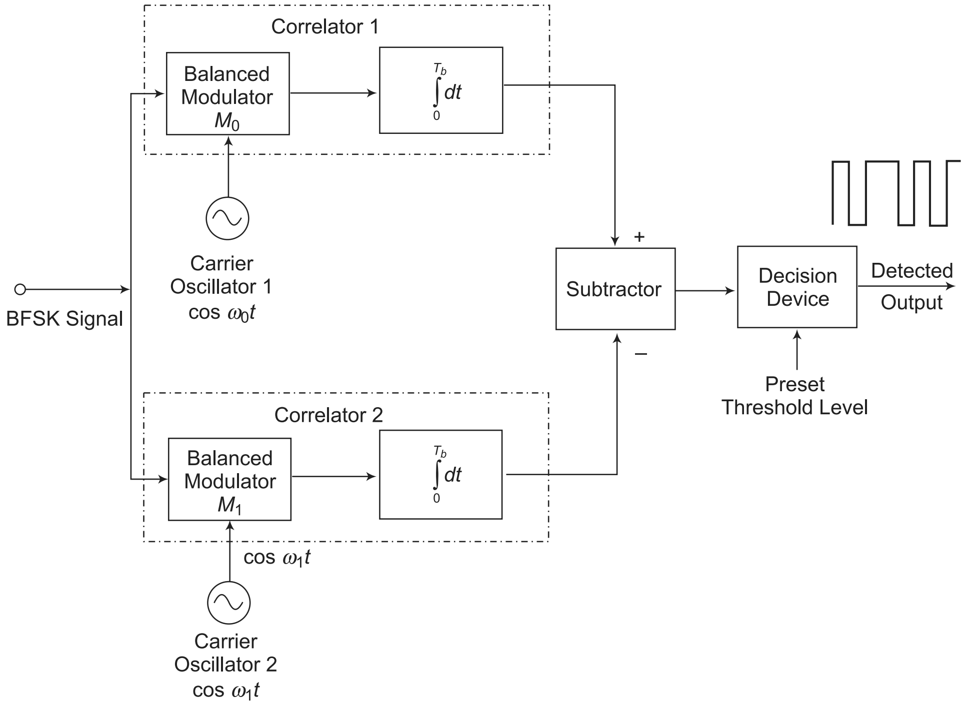


Figure 3.1.10 Functional Block Schematic of Coherent Binary FSK Detector

Non-Coherent Binary FSK Detection

In non-coherent FSK detection, there is no reference carrier signal involved in the demodulation process that is synchronized either in frequency, phase, or both with the received FSK signal.

Figure 3.1.11 shows a simplified functional block schematic of non-coherent binary FSK detector.

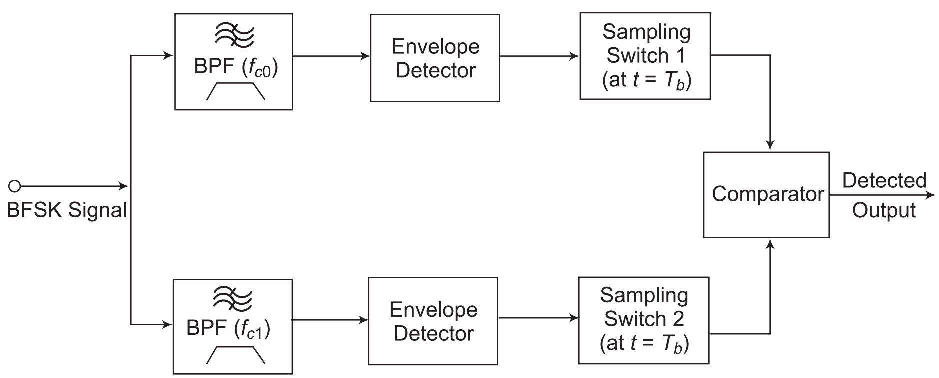


Figure 3.1.11 Functional Block Schematic of Non-coherent Binary FSK Detector

Functional Description of Non-Coherent FSK Detector

- In non-coherent detection, the received binary FSK signal is applied to two bandpass filters, tuned at f_{c0} and f_{c1}, respectively.
- The filtered signals are then applied to its corresponding envelope detectors.
- The outputs of the two envelope detectors are sampled at $t = T_b$ where T_b is the pulse width, and compared separately.
- The comparator responds to the larger of the two signal inputs and detected output is produced.
- For example, if a binary 0 is transmitted by a pulse of frequency f_{c0}, then this pulse will appear at the output of the filter tuned at f_{c0}.
- At that instant, practically no signal appears at the output of the filter tuned at f_{c1}.
- Hence, the sample of the envelope detector output following the filter tuned at f_{c0} will be greater than the sample of the envelope detector output following the filter tuned at f_{c1}.
- Therefore, the output of the comparator will be the binary symbol 0.
- Similarly, if a binary 1 is transmitted by a pulse of frequency f_c then following the same procedure, the output of the comparator will be binary symbol 1.

IMPORTANT! Due to its simplicity, a non-coherent FSK detector is generally used. Consequently, a coherent FSK detector is quite complex and is seldom used.

FSK Detection using PLL

FSK detection using *phase-locked loop* (PLL) is the most commonly used circuit for demodulating binary FSK signals.

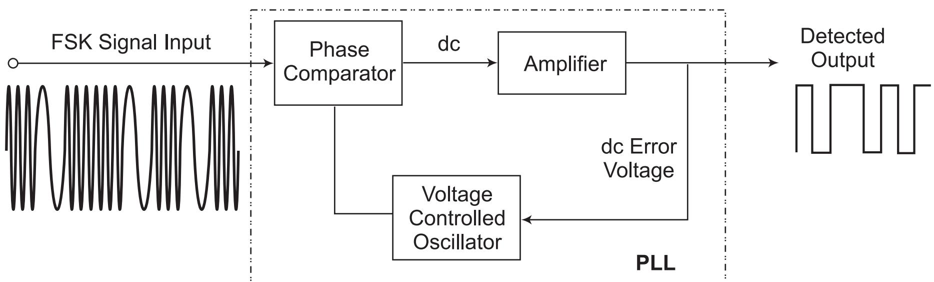


Figure 3.1.12 Functional Block Schematic of Binary FSK Detector using PLL

Description of FSK Detector using PLL

- The incoming FSK signal is applied to the PLL circuit.
- Generally, the natural frequency of the PLL is made equal to the center frequency of the FSK modulator.
- As the input signal shifts between two frequencies f_{c0} and f_{c1}, the phase comparator gets locked to the input frequency.
- A corresponding dc signal is produced which is used as the *dc error voltage* to correct the output of Voltage Controlled Oscillator (VCO).
- Thus, the VCO tracks the input frequency between two frequencies f_{c0} and f_{c1}.
- In fact, the changes in the dc error voltage follow the changes in the input FSK signal frequency and are symmetrical around 0 V.
- Because there are only two frequencies f_{c0} and f_{c1}, corresponding to binary symbol 0 and 1.
- Therefore, the output is also a two-level binary representation of the incoming FSK signal.

Binary FSK has a comparatively poorer error performance than PSK or QAM, and consequently seldom used for high-performance digital radio systems. It is generally used in low-performance, low-cost, asynchronous low-speed data modems (up to 1200 bps) over analog voice-band telephone lines. It also finds application in pager systems, HF radio tele-type transmission systems, and Local Area Networks (LAN) using coaxial cables.[7]

Application

SOLVED EXAMPLE 3.1.3 **Binary FSK Signal Bandwidth**

A 1000 bps binary information data signal is required to be transmitted in half-duplex mode using binary FSK digital modulation technique. If the separation between two carrier frequencies is 3000 Hz, determine the baud and the minimum bandwidth of the FSK signal.

Solution Given input bit rate, f_b = 1000 bps

We know that in binary FSK, baud = bit rate

Therefore, baud = 1000 **Ans.**

The minimum bandwidth of the binary FSK signal is given as

$$B_{\text{BFSK}} = \left|(f_{c1} - f_{c0})\right| + 2f_b$$

For given values of $(f_{c1} - f_{c0})$ = 3000 Hz, and f_b = 1000 bps, we get

$$\text{B}_{\text{BFSK}} = 3000 \text{ Hz} + 2 \times 1000 \text{ bps} = 5000 \text{ Hz}$$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 3.1.4 Determine the baud and the minimum bandwidth for the binary FSK signal using 39 kHz frequency for transmitting binary symbol ‘0’ and 41 kHz frequency for transmitting binary symbol ‘1’. The input bit rate is specified as 2 kbps.

Ex 3.1.5 Determine the bandwidth and baud rate for an FSK signal with two frequency offsets placed at 32 kHz and 24 kHz, and a bit rate of 4 kbps.

3.1.3 Phase Shift Keying (PSK)

Define

Phase Shift Keying (PSK) is a form of digital modulation technique in which the phase angle of the analog sinusoidal carrier signal f_c is varied to represent the input digital data. However, the amplitude and frequency of the modulated signal remains constant.

Mathematical Expression

In *binary phase shift keying (BPSK*), the phase of the sinusoidal carrier signal is changed by 0° or 180° (π radians) corresponding to two different voltage levels of binary modulating signals (1 and 0). This is the reason that binary PSK is sometimes called *biphase modulation* or *phase reversal keying*. However, the amplitude of the transmitted BPSK signal remains fixed. For a sinusoidal carrier signal sin $(2\pi f_c t)$, the binary PSK signal can be expressed as

$$\boxed{v_{\text{BPSK}}(t) = \begin{Bmatrix} \sin(2\pi f_c t) & \text{for binary 1} \\ \sin(2\pi f_c t + \pi) & \text{for binary 0} \end{Bmatrix}}$$

where f_c is the frequency of the carrier signal.

[7] There is a natural and clear connection between digital and analog bandpass (carrier) modulation techniques—FSK and FM respectively, because the message information (digital in FSK and analog in FM) is directly reflected in the varying frequency of the modulated signals (the carrier signal being sinusoid in both cases). Thus, FSK is essentially an FM signal with modulation index of unity.

It may be noted that we have arbitrarily taken amplitude of the signal to be unity because only the waveforms of the signals are relevant for our discussion and not the amplitude levels at various points in the process of modulation and demodulation.

In BPSK, the data $d(t)$ is a stream of binary symbols 1 and 0 with voltage levels +1 V and –1 V (say) for polar NRZ waveform, respectively. Hence, the binary PSK signal can be re-written as

$$\boxed{v_{\text{BPSK}}(t) = d(t)\sin(2\pi f_c t)}$$

where $d(t)$ = +1 V for binary 1, and –1 V for binary 0.

Using the definition and mathematical expression for binary PSK signal. Figure 3.1.13 shows the PSK signal waveform for the given binary data 1 0 1 1 0 1 0 1 1 1.

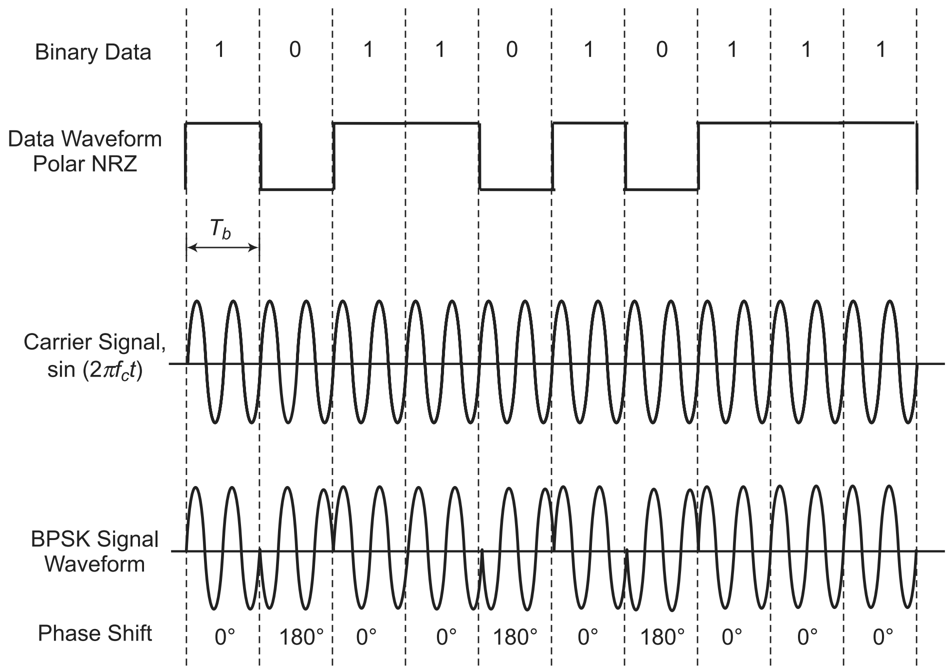


Figure 3.1.13 Binary PSK Signal Waveform

Interpretation of Binary PSK Signal Waveform

- When the input binary data changes from 1 to 0 or vice versa, the binary PSK output signal phase shifts from 0° to 180° or vice versa.
- Hence, the two pulses are π radians apart in phase.
- The information (input binary data) resides in the phase or the sign of the pulse in binary PSK signal.

Constellation Diagram

Geometric interpretation is another way of representing PSK signals, called *signal state-space diagram*. It is similar to a phasor diagram except that the entire phasor is not drawn, instead only the relative positions of the peaks of the phasors (called as dots) are indicated. Such set of possible combinations of amplitude levels on the *x*-*y* plot in the form of a pattern of dots is also known as a *constellation diagram*.

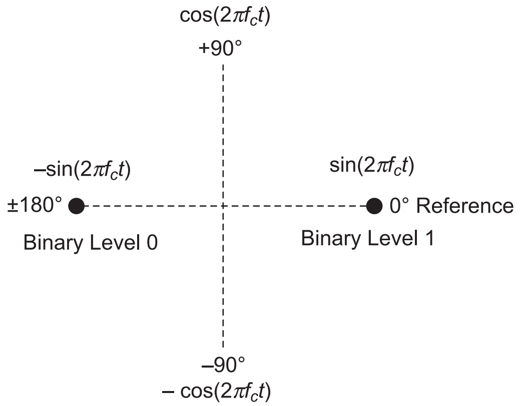


Figure 3.1.14 Constellation Diagram of BPSK Signal

Table 3.1 Relationship between Symbols, Bits, and Phase Shift in BPSK Signal

Symbol	Bits in Symbol	Phase Shift in Carrier Signal
s_0	Binary level '0'	180° or π radians
s_1	Binary level '1'	0° or 0 radians

Bit Rate and Baud Rate

It is seen that a 180° phase transition is required for the symbol transition from 0 to 1 or vice versa. Thus, the transmitted binary PSK signal has to go to zero amplitude momentarily as it makes this transition. Clearly, the time duration of one bit (T_b) is the same as the symbol time duration (T_s) of the binary PSK signal output. Hence, the bit rate (f_b) equals the baud rate.

PSD of PSK Signal

As we know that for an appropriately chosen carrier frequency, the process of modulation causes a shift in the baseband signal PSD. When a polar NRZ line code, having pulse width as T_b, is used to represent the input binary data, the PSD of a binary PSK signal is the same as that of the polar baseband signal shifted to $\pm f_c$.

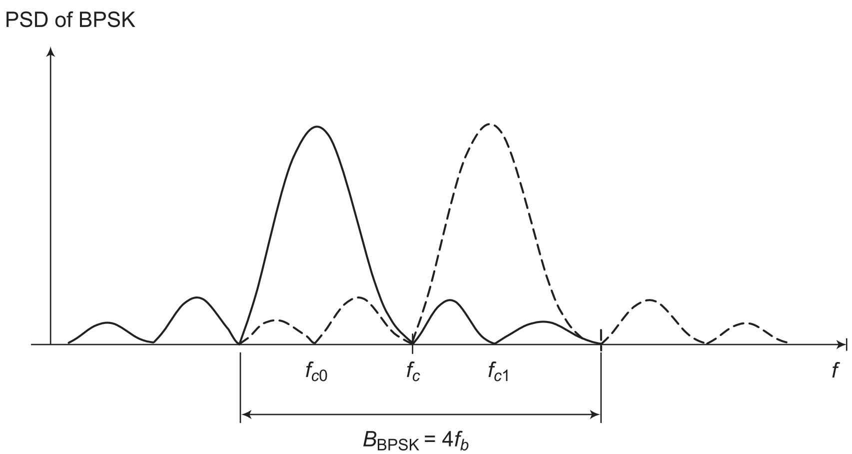


Figure 3.1.15 PSD of Binary PSK Signal

Interpretation of PSD of BPSK Signal Waveform

It may be noted here that the PSD of a binary PSK signal has the same shape (with a different scaling factor) as the PSD of the ASK signal except its discrete components. Hence, when a data stream having bit duration T_b is to be transmitted by BPSK, the *transmission bandwidth* must be nominally $2f_b$ where $f_b = 1/T_b$ is the input bit rate. The spectrum of power spectral density of a BPSK signal extends over all frequencies. But the main lobe contains 90% of power of the output waveform. Due to overlapping of spectra for multiplexed binary PSK signals, interchannel interference as well as intersymbol interference arises.[8]

PSK Modulator

In practice, a BPSK signal is generated by applying the sinusoidal carrier signal $\sin(2\pi f_c t)$ and the baseband signal $d(t)$ as the modulating signal to a balanced modulator.

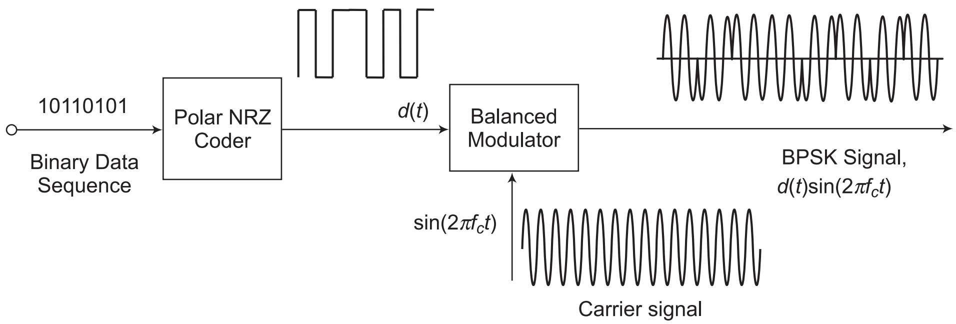


Figure 3.1.16 Functional Block Schematic of a PSK Modulator

Functional Description of PSK Modulator

- The binary data sequence of 0s and 1s is converted into a polar NRZ signal.
- Its output is then applied to a balanced modulator whose other input is from a local carrier oscillator.
- The output of the balanced modulator will be in phase with the carrier signal phase for binary data 1, and 180° out of phase with the carrier signal phase for binary data 0.
- Thus, the balanced modulator acts as a phase reversing switch.

Coherent Binary PSK Detection

As we know that in binary PSK, a binary symbol 1 is transmitted by a sinusoidal signal $\sin(2\pi f_c t)$ and a binary symbol 0 by a sinusoidal signal $\sin(2\pi f_c t + \pi)$ for a polar NRZ signal. The information, therefore, resides in the carrier phase. The binary PSK signal cannot be demodulated by non-coherent detection technique which uses an envelope detector because the envelope always remains constant for both binary symbols 1 and 0. Hence, binary PSK signals can be detected by coherent technique only.

Functional Description of Coherent PSK Detector (Refer Fig. 3.1.17)

- The incoming BPSK signal is applied to correlators which consists of a synchronous demodulator (multiplier) and an integrator (low-pass filter).
- The other input signal to the synchronous demodulator is the recovered carrier signal.
- It may be noted that the carrier signal can be recovered from the received signal by applying it to a square-law device, followed by to a bandpass filter tuned at $2f_c$, and then to a half-frequency divider comprising of a *D*-type flip-flop and a narrowband filter tuned at f_c.
- This part of the schematic is not shown here for the sake of simplicity.

[8] The binary PSK modulation is closely connected with the analog QAM signal, as well as it is effectively a digital manifestation of the DSB-SC amplitude modulation. Binary PSK has the same transmission bandwidth as that of ASK but is more power efficient as DSB-SC is more power efficient than AM.

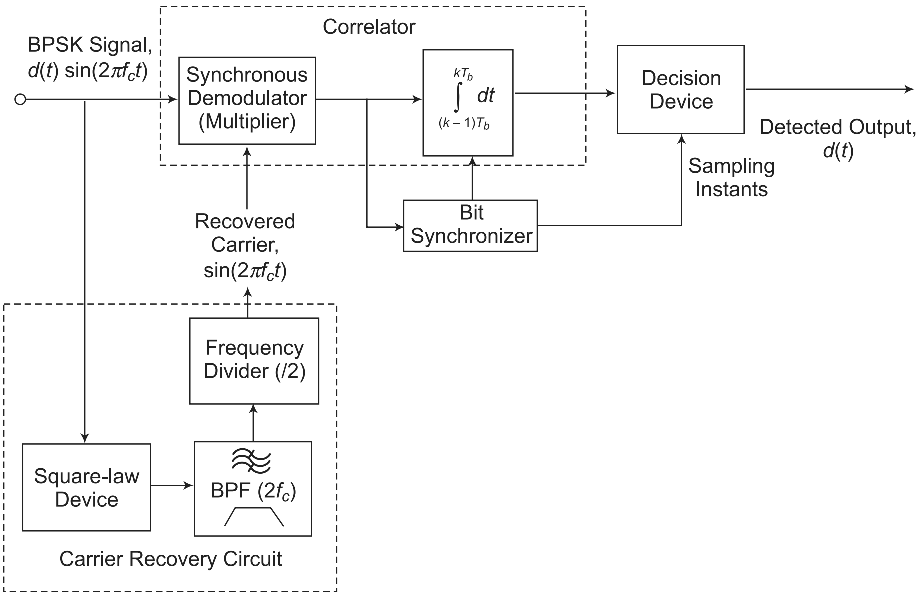


Figure 3.1.17 Functional Block Schematic of a Coherent Binary PSK Detector

- Thus, the recovered carrier signal has the identical frequency as that used in the binary PSK modulator.
- The output of a synchronous demodulator is applied to an integrator as well as a bit synchronizer.

Bit Synchronizer

The *bit synchronizer* is a device which is able to recognize precisely the instants which correspond to the end of the time interval allotted to one bit (i.e., T_b) and the beginning of the next bit.

Integrate and Dump Filter

The output of the bit synchronizer closes the switch of the integrator at that instant for a very brief period to discharge the integrator capacitor, keeping it open during the remaining period of the bit interval, closing the switch again very briefly at the end of the next bit time, and so on. This operation is known as *integrate-and-dump* filter operation because the output signal of interest here is the integrator output at the end of a bit interval but immediately before the closing of the switch.

Decision Device

The output of the bit synchronizer is also made available to the decision device (another fast-operating switch) which samples the output signal just prior to dumping the capacitor. Thus, the operation of the bit synchronizer allows us to detect each bit independently of every other bit. The brief closing of both switches, after each bit has been determined, also ensures that the detector deals exclusively with the present bit. Therefore, we see that our system reproduces at the demodulator output the transmitted bit stream $d(t)$.

IMPORTANT!

This consolidates our earlier viewpoint that binary PSK[9] signal must be detected coherently to achieve good S/N ratio. So far our discussion on coherent detection of binary PSK signal has

[9] The binary PSK, being polar signaling in nature, requires 3 dB less power than ASK or FSK for the same noise immunity, that is, for the same error probability in pulse detection. Thus, it is the most power-efficient scheme.

ignored the effects of thermal noise, frequency jitter in the recovered carrier signal and random fluctuations in propagation delay of the transmitted signal through the channel. For considering the effects of these factors, a phase-locked synchronization system should be applied for carrier-recovery. So, the design of a binary PSK coherent detector is quite complex.

Application Binary PSK has been used in earlier telephone modems @ 2400 bps and 4800 bps. It is commonly applied in digital satellite communications.

SOLVED EXAMPLE 3.1.6 **Binary PSK Symbol Rate and Bandwidth**

In a binary PSK digital communication system, the bit rate of a polar NRZ data sequence is 1 Mbps and carrier frequency of transmission is 100 MHz. Determine the symbol rate of transmission and the minimum bandwidth requirement of the communications channel.

Solution Given bit rate of a bipolar NRZ data sequence, f_b = 1 Mbps or 1×10^6 bps

Therefore, bit period, $T_b = \dfrac{1}{f_b} = \dfrac{1}{1\text{ Mbps}} = 1\ \mu\text{s}$

In binary PSK digital communication system, each binary bit is a symbol.

That is, symbol duration, $T_s = T_b = 1\ \mu$s

Therefore, symbol rate of transmission $= \dfrac{1}{T_s} = \dfrac{1}{1\ \mu\text{s}} = 10^6$ symbols/second **Ans.**

Minimum bandwidth, $B_{BPSK} = 2f_b = 2 \times 1$ Mbps, or 2 MHz **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 3.1.7 Determine the minimum bandwidth and baud rate for a binary PSK modulator with a carrier frequency of 40 MHz and an input bit rate of 500 kbps.

Ex 3.1.8 For a binary PSK modulator with an analog carrier signal of 70 MHz and an input bit rate of 10 Mbps, determine the minimum bandwidth of BPSK signal, and the baud.

MATLAB simulation exercises on binary digital modulations FSK and PSK,

Scan the QR code given here OR visit: http://qrcode.flipick.com/index.php/122

3.1.4 Differential Binary Phase Shift Keying (DBPSK)

Recall In ordinary binary phase shift keying (BPSK), a signal $v_{\text{BPSK}}(t) = \sin(2\pi f_c t)$ is transmitted in a given time slot, if the corresponding baseband binary digit in that time slot is a 1 and a signal $v_{\text{BPSK}}(t) = \sin(2\pi f_c t + \pi) = -\sin(2\pi f_c t)$ is transmitted in a given time slot, if the corresponding baseband binary digit in that time slot is a 0. The BPSK receiver requires coherent detection which involves the use of a carrier signal that is in frequency and phase synchronization with that of the carrier signal used at the transmitter. As one of these two signals, forming an antipodal pair, is transmitted during any time slot, BPSK has the least probability of error for a given bit energy.

Define *Differential binary PSK (DBPSK),* or simply *differential PSK (DPSK)*, is an alternate form of binary PSK where the binary data information is contained in the difference between the phases of two successive signaling elements rather than the absolute phase.

A slightly different technique from BPSK, known as differential BPSK (DBPSK), or simply DPSK, does not require coherent detection at the receiver. However, it is sub-optimum in its error performance in the sense that the transmission rate for this system is fixed but errors tend to occur in pairs.

Why Differential BPSK?

A binary 1 is represented by sending a signal bit of opposite phase (by 180°) to the preceding one. A binary 0 is represented by sending a signal bit of the same phase as the previous one. In DBPSK, a symbol consists of two input bits. Therefore, symbol duration, $T_s = 2\ T_b$, where T_b is the bit duration.

Interpretation

The bandwidth of DBPSK signal is

Bandwidth of DBPSK

$$2/T_s = 2/2\ T_b = 1/T_b = f_b$$

This implies that the minimum bandwidth in DBPSK is equal to the maximum baseband signal frequency, that is, f_b. Hence, the bandwidth of DBPSK signal is one-half that of BPSK signal.

The DBPSK receiver makes use of the signal received in the preceding time slot itself as the reference phase for detecting the present signal. This is done by correlating the signal received during the present time slot with that received in the previous time slot. This implies that the receiver must have one-bit storage facility.

Why is DBPSK a Sub-optimum Method?

- If the two signals are correlated, then the receiver decides that the data bit received in the present time slot is a binary 1.
- If the two signals are uncorrelated, then the receiver decides that the data bit received in the present time slot is a binary 0.

Instead of using the actual carrier signal for phase reference, DBPSK technique utilizes the noisy received signal of the preceding time slot as the reference signal for detection purpose, so it is a sub-optimum method.

SOLVED EXAMPLE 3.1.9 **Illustration of DBPSK Operation**

Consider a binary data sequence 1 1 0 0 1 1 0 0 which is required to be transmitted using DBPSK. Show the step-by-step procedure of generating and detecting DBPSK signal. Assume arbitrary start-up reference bit as 1.

Solution Let the given binary data sequence is denoted by $b(t)$, and the differentially encoded DBPSK data sequence by $d(t)$. The step-by-step procedure of generating and detecting DBPSK signal is given in Table 3.1.1.

Table 3.1.1 DBPSK Signal–Generation and Detection

Step#	Parameter	Ref.	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8
1	Data sequence, $b(t)$	—	1	1	0	0	1	1	0	0
2	Encoded sequence, $d(t)$	1	1	1	0	1	1	1	0	1
3	Transmitted phase of $d(t)$	0	0	0	π	0	0	0	π	0
4	Detected sequence, $b'(t)$	—	1	1	0	0	1	1	0	0

Thus, it is observed that the given binary data sequence, $b(t)$: 1 1 0 0 1 1 0 0, is same as the detected sequence, $b'(t)$: 1 1 0 0 1 1 0 0.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 3.1.10 For a given binary data sequence 1 0 0 1 1 0 0 1, show the step-by-step procedure of generating and detecting DBPSK signal. Assume arbitrary start-up reference bit as 1.

Ex 3.1.11 Consider a binary data sequence 1 0 0 1 1 0 0 1 which is required to be transmitted using DBPSK. Show the step-by-step procedure of generating and detecting DBPSK signal. Assume arbitrary start-up reference bit as 0.

DBPSK Modulator

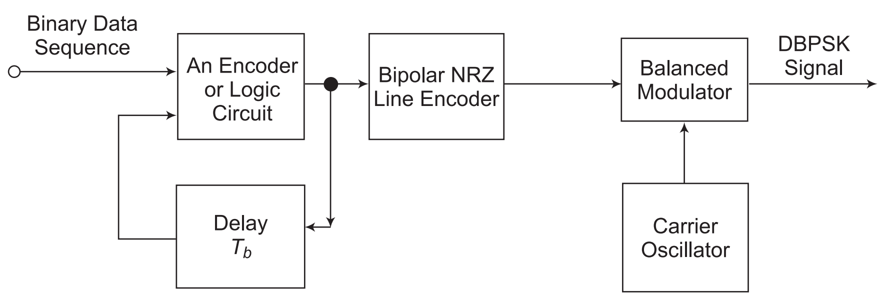


Figure 3.1.18 Differential Binary PSK Modulator

Functional Description of DBPSK Modulator

- The input binary data sequence is applied to the input of an encoder or logic circuit such an Ex-OR logic gate (complement of an Ex-OR gate).
- The other input to encoder is one-bit delayed version of previous bit.
- The output of encoder is then applied to bipolar NRZ line encoder, followed by balanced modulator.
- The other input to balanced modulator is from a sinusoidal carrier signal oscillator. The output is DBPSK signal in which the phase shift depends on the current bit and the previous bit.

Non-Coherent DBPSK Detection Differential binary PSK detector may be viewed as a non-coherent version of binary PSK detector.

Figure 3.1.19 shows a simplified functional block schematic of non-coherent differential binary PSK detector.

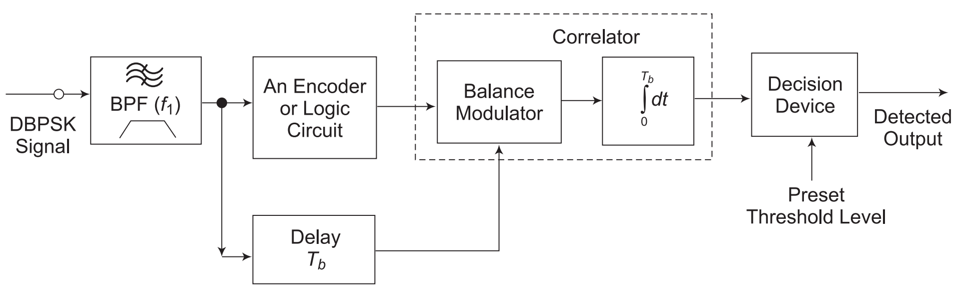


Figure 3.1.19 Functional Block Schematic of Non-Coherent DBPSK Detector

Functional Description of Non-Coherent DBPSK Detector

- A non-coherent DBPSK detector is simple to implement because the carrier recovery circuit is not needed.
- The received signal is applied to a bandpass filter before applying to an encoder or logic circuit.
- The configuration of encoder is inverted as compared to that of in DBPSK detector.
- The input data is delayed by one bit interval.
- Both these signals are then applied to a correlator which comprises of a balanced modulator and an integrator.
- The difference in original data and its delayed version is proportional to the difference between the carrier phase of the received DBPSK signal and its delayed version, measured in the same bit interval.
- This phase difference is used to determine the sign of the correlator output for each bit interval.
- When the phase difference is 0°, the integrator output is positive; and when the phase difference is 180°, the integrator output is negative.
- The output of the integrator is then compared with zero volt preset threshold level by the decision device—if input to the decision device is greater than 0 volt then the detected output is 1, and if input to the decision device is less than 0 volt then the detected output is 0.
- And, if the reference phase is incorrectly assumed then only the first demodulated bit is in error.

Table 3.1.2 gives comparison of some parameters of BPSK and DBPSK digital modulation techniques.[10]

Table 3.1.2 Comparison of BPSK and DBPSK

S. No.	Parameter	BPSK	DBPSK
1.	Variable characteristics of analog carrier signal	Phase	Phase
2.	Maximum transmission bandwidth in terms of bit rate, f_b	$2f_b$	f_b
3.	Probability of error	Low	Higher than BPSK
3.	Noise immunity	Good	Better than BPSK
5.	Bit detection at receiver	Based on single-bit interval	Based on two successive bit intervals
6.	Synchronous carrier at demodulator	Required	Not required
7.	System complexity	Moderate	High

3.1.5 Differentially Encoded Phase Shift Keying (DEPSK)

What is DEPSK?

In *Differentially Encoded PSK (DEPSK)*, the input sequence of binary data is modified such that the next bit depends upon the previous bit.

[10] Differential PSK requires less transmission bandwidth and has higher error probability and bit-error-rate because it uses two successive bits for its detection (error in the first bit creates error in the second bit too). It requires additional 1–3 dB SNR to achieve the same BER performance as that of conventional BPSK scheme.

It is a common form of phase modulation that conveys data by changing the phase of the carrier wave. As mentioned for BPSK, there is an ambiguity of phase if the constellation is rotated by some effect in the communication channel through which the signal passes. We can overcome this problem by using the data to change rather than setting the phase.

Example

In DEPSK, a binary '1' may be transmitted by adding 180° (π radians) to the current phase and a binary '0' by adding 0° to the current phase. So, the phase-shifts are 0°, 90°, 180°, 270° corresponding to data '00', '01', '11', '10'. This kind of encoding may be demodulated in the same way as for conventional PSK but the phase ambiguities can be ignored.

Interpretation

- A transition in the given binary sequence with respect to previous encoded bit is represented by a phase change of π radians.
- No transition in the given binary sequence with respect to previous encoded bit is represented by no phase change or by 0.

Figure 3.1.20 shows the DEPSK signal in the time-domain.

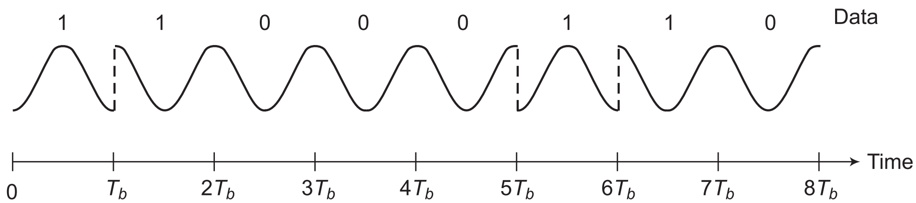


Figure 3.1.20 DEPSK Signal in Time-Domain

DEPSK Modulator Differential encoded PSK modulator is identical to differential PSK modulator except that it does not require delay T_b at carrier frequency.

DEPSK Detector

The DEPSK needs coherent detection to recover the differentially encoded bit stream. In a DEPSK detector, the previously received bits are used to detect the present bit. The signal $b(t)$ is recovered in the same way as done in BPSK, and then applied to one input of an Ex-OR gate. The other input to Ex-OR gate is $b(t - T_b)$, that is delayed version of $b(t)$ by T_b. The output of a DEPSK detector will be binary logic 0, if $b(t) = b(t - T_b)$. Similarly, the output of DEPSK detector will be binary logic 1, if $b(t) = b\widehat{(t - T_b)}$.[11]

SOLVED EXAMPLE 3.1.12 Illustration of DEPSK Operation

A binary data sequence 0 0 1 0 0 1 1 is to be transmitted using DEPSK. Show the step-by-step procedure of generating and detecting DBPSK signal. Assume arbitrary starting reference bit as 0.

Solution Let the given binary data sequence be denoted by $b(t)$, and the differentially encoded data sequence by $d(t)$. The step-by-step procedure of generating and detecting a DEPSK signal is given in Table 3.1.3.

[11] In DEPSK, errors must result from a comparison with the preceding and succeeding bit. Thus, in case of DEPSK it can be seen that the errors always occur in pairs, whereas in case of DBPSK the errors may occur either in pairs or in single-bit. Analysis shows that differential encoding approximately doubles the error rate in AWGN channel. However, the physical channel will introduce an unknown phase-shift to the PSK signal. In these cases the differential schemes can yield a better error-rate than the ordinary schemes which rely on precise phase information.

Table 3.1.3 DEPSK Signal—Generation and Detection

Step#	Parameter	Ref.	Value1	Value2	Value3	Value4	Value5	Value6	Value7
1	$b(t)$	—	0	0	1	0	0	1	1
2	$d(t)$	0	1	0	0	1	0	0	0
3	Phase of $d(t)$	π	0	π	π	0	π	π	π
4	$d(t–T_b)$	—	0	1	0	0	1	0	0
5	Phase of $d(t–T_b)$	—	π	0	π	π	0	π	π
6	Phase comparison	—	–	–	+	–	–	+	+
7	Detected $b(t)$	—	0	0	1	0	0	1	1

Thus, it is observed that the given binary data sequence, $b(t)$: 0 0 1 0 0 1 1, is same as detected $b(t)$: 0 0 1 0 0 1 1.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 3.1.13 For a given data stream of 1 0 1 1 0 1, determine the phase shift of the carrier in differential binary phase shift keying.

Ex 3.1.14 For a given data stream of 1 0 1 1 0 1, determine the phase shift of the carrier in a differentially encoded phase-shift keying system.

Self-Assessment Exercise linked to LO 3.1

Q3.1.1 Speech signals must be encoded into a format suitable for digital transmission. Identify types of speech-signal formatting techniques that can be used along with digital modulation. ○○●

Q3.1.2 Differentiate between bit rate and baud rate. ○●●

Q3.1.3 The use of binary FSK is restricted to low-performance, low-cost, asynchronous data modems that are used for data communications over analog voice-band telephone lines. Why is it rarely used for high-performance digital radio system? ●●●

Q3.1.4 Using the binary data sequence 1 1 1 0 0 1 0, represent it using the (a) ASK; (b) FSK; (c) PSK. ○○●

Q3.1.5 Specify the possible reasons for FSK being suitable for HF radio applications. ○●●

Q3.1.6 How is binary phase-shift keying digital modulation technique different from that of conventional analog phase modulation? ●●●

Q3.1.7 Consider a digital message having an input data rate of 8 kbps. Compute the transmission bandwidth required if the message is transmitted through binary PSK. ○●●

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/123

Note ○○● Level 1 and Level 2 Category
○●● Level 3 and Level 4 Category
●●● Level 5 and Level 6 Category

> ✓ **If you have been able to solve the above exercises then you have successfully mastered**
>
> **LO 3.1:** *Analyze binary digital modulation techniques such as ASK, FSK and PSK.*

LO 3.2

3.2 *M*-ARY DIGITAL MODULATION TECHNIQUES

Recall

In binary digital modulation techniques, we have seen that the required transmission bandwidth for a data stream whose bit duration is T_b must be nominally $2f_b$, where the input bit rate $f_b = 1/T_b$.

For example, in binary PSK, we transmit each data bit individually. Depending on whether the bit is logic 1 or logic 0, we transmit one or another of a sinusoidal carrier signal for the bit interval T_b, the sinusoids differing in phase by 180°.

What We Discuss Here .....

In multi-level or *M*-ary digital modulation techniques, one of the *M* possible signal elements are transmitted during each bit interval of T_b seconds. We begin our discussions on Quadrature Phase Shift Keying (*QPSK*) and its variants such as *offset QPSK* and π/4-*QPSK*. Then, we move on to understand constant envelope digital modulation techniques such as Minimum Shift Keying (*MSK*) and *Gaussian MSK*. Finally, we describe hybrid digital modulation technique known as Quadrature Amplitude Modulation (*QAM*).

3.2.1 Quadrature Phase Shift Keying (QPSK)

What is QPSK?

Quadrature Phase-Shift Keying, or Quadriphase-Shift Keying (QPSK) is one of bandwidth efficient bandpass digital modulation technique that makes use of quadrature multiplexing. The QPSK system can be considered equivalent to two BPSK systems operating in parallel and having carrier signals which are of the same frequency but in phase quadrature. Just like in BPSK, in QPSK too, the information is carried in the phase of the transmitted signal. However, unlike in BPSK, in QPSK, we combine two successive bits in a bit stream to form a symbol.

Mathematical Expression

With two bits, there are four possible conditions: 11, 10, 01, and 00. Depending on which of the four two-bit symbols (called *dibits*) develops in the input data stream, we transmit one or another of four sinusoids of duration $2T_b$, the sinusoids differing in phase by 90° or 180° with another. Thus, in a QPSK modulator, each dibit code generates one of the four possible output phases (+45°, +135°, –45°, –135°), hence, the name "quadrature" meaning "four" phases.

Mathematically, the QPSK signal for one symbol duration, consisting of two bits each, can be expressed as

$$\boxed{S_{\text{QPSK}}(t) = \begin{Bmatrix} \sin\,(2\pi f_c\, t - 3\pi/4) & \text{for } 0\ 0 \\ \sin\,(2\pi f_c\, t - \pi/4) & \text{for } 0\ 1 \\ \sin\,(2\pi f_c\, t + 3\pi/4) & \text{for } 1\ 0 \\ \sin\,(2\pi f_c\, t + \pi/4) & \text{for } 1\ 1 \end{Bmatrix}}$$

In general, we can say that QPSK is an M-ary constant-amplitude quadrature PSK digital modulation scheme in which number of bits is two ($n = 2$) and the number of signaling elements (symbols) are four, i.e., $M = 4$ (hence, sometimes known as "quaternary" meaning "4", 4-ary PSK).

Interpretation

Table 3.2.1 Symbols, Bits, and Phase Shift in QPSK Signal

Symbol	Binary Input	Phase Shift in QPSK Signal
s_1	0 0	$-135°$ or $-3\pi/4$ radians
s_2	0 1	$-45°$ or $-\pi/4$ radians
s_3	1 0	$+135°$ or $3\pi/4$ radians
s_4	1 1	$+45°$ or $\pi/4$ radians

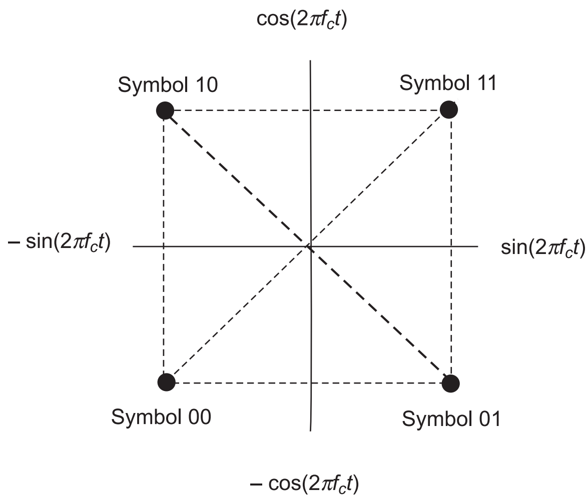


Figure 3.2.1 Constellation Diagram of QPSK Signal

Interpretation of Constellation Diagram

It is seen that the transmitted QPSK signal has to go to zero amplitude momentarily as it makes 180° transition (from 11 to 00 or from 10 to 01 or vice versa). Whereas it makes only 90° transition (from 11 to 10, 10 to 00, or 00 to 01, or 01 to 11, or vice versa).

Bandwidth Considerations of QPSK Signal

The rate of change at the output (baud) is equal to one-half the input bit rate $f_b/2$ (i.e., two input bits produce one output phase change). As a result, each bit in QPSK can be transmitted using half the bandwidth that of required to transmit BPSK signal, i.e., QPSK signal requires minimum Nyquist transmission bandwidth equal to input bit rate f_b only, a significant improvement as compared to that of BPSK.[12]

[12] QPSK signal can carry twice as much data in the same bandwidth as can a single-bit system, provided the SNR is high enough. QPSK needs more power and higher SNR (approximately 3 dB) than BPSK, to obtain the same performance for same probability of error.

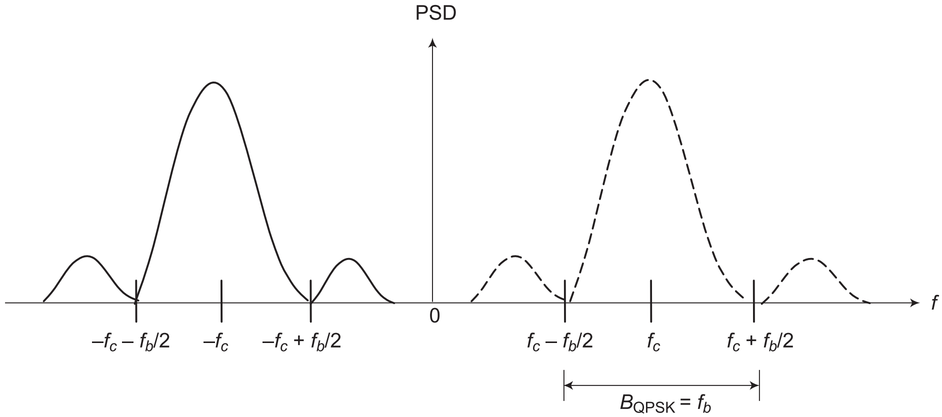


Figure 3.2.2 PSD of binary PSK Signal

Interpretation of PSD of QPSK Signal Waveform

If there is any phase change, it occurs at minimum duration of T_b only. The signal transitions are abrupt and unequal and this causes large spectrum dispersion. Practically, pulse shaping is carried out at baseband to provide proper filtering at the QPSK modulator output. A bandpass filter is used at the output of a QPSK modulator which confines the power spectrum of QPSK signal within the allocated band, prevents spill-over of signal energy into adjacent channels, and removes out-of-band spurious signals generated during modulation process.

QPSK Modulator

In QPSK, the input binary data sequence $d(t)$ is divided into odd and even bit sequences, that is, $d_1(t)$ and $d_2(t)$ respectively. Each symbol in both of these bit sequences has a period of $T_s = 2T_b$ seconds. The input is a data stream of binary digits with a data rate of $f_b = 1/T_b$, where T_b is the bit duration. This data stream is applied to polar NRZ encoder. It is followed by conversion into two separate bit streams by taking alternate bits at the rate of $f_b/2$ bps.

Figure 3.2.3 shows a simplified functional block schematic of QPSK modulator.

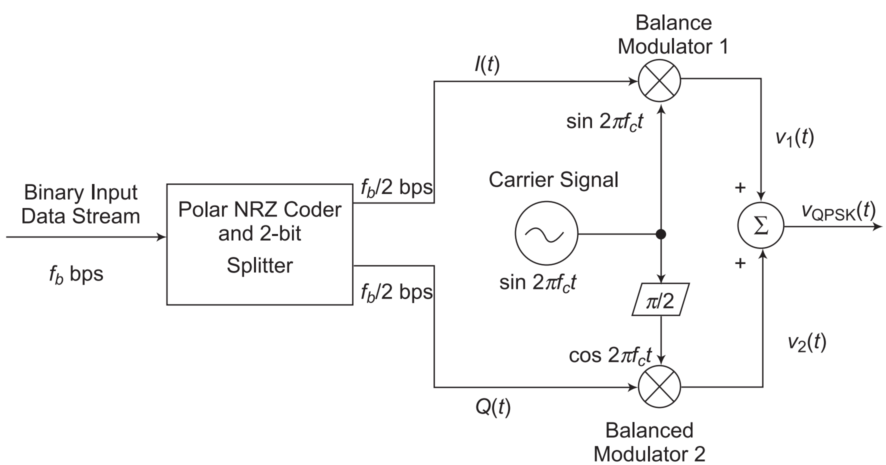


Figure 3.2.3 Functional Block Schematic of QPSK Modulator

Functional Description of QPSK Modulator

QPSK is characterized by two parts of the baseband data signal: the in-phase signal $I(t)$ and the Quadrature signal $Q(t)$. QPSK modulator is a combination of two BPSK modulators. Thus, at the output of the I-balanced modulator 1, there are two types of phases produced, which are $+\cos(2\pi f_c t)$ and $-\cos(2\pi f_c t)$. Similarly, at the output of the Q-balanced modulator 2, again two phases are produced as $+\sin(2\pi f_c t)$ and $-\sin(2\pi f_c t)$. When the linear adder combines these two groups of orthogonal BPSK modulated signals, there will be four possible phases which are

$$+\cos(2\pi f_c t) + \sin(2\pi f_c t),\ -\cos(2\pi f_c t) + \sin(2\pi f_c t),$$
$$+\cos(2\pi f_c t) - \sin(2\pi f_c t),\ \text{and}\ -\cos(2\pi f_c t) - \sin(2\pi f_c t).$$

- The binary input data stream of 1s and 0s is encoded into a polar non-return-to-zero (NRZ) stream of 1s and –1s.
- The *bit splitter* (also known as 2-bit demultiplexer) segregates the odd and even indexed polar binary digits.
- All the odd-indexed bits are directed to the upper in-phase channel (I-channel) where they phase-modulate the in-phase carrier signal $\sin(2\pi f_c t)$ using a balanced modulator 1.
- All the even-indexed bits are directed to the lower quadrature-phase channel (Q-channel) where they phase-modulate the quadrature-phase carrier signal $\cos(2\pi f_c t)$ which is $\pi/2$-shift version of the in-phase carrier signal $\sin(2\pi f_c t)$ using a balanced modulator 2.
- The outputs of balanced modulators, $v_1(t)$ and $v_2(t)$ are basically two BPSK signals which are linearly added in summer.
- The output QPSK signal can be expressed as:

$$v_{\text{QPSK}}(t) = \frac{1}{\sqrt{2}}\left[I(t)\sin(2\pi f_c t) - Q(t)\cos(2\pi f_c t)\right]$$

Note... *When $d_1(t)$ and $d_2(t)$ signals modulate the sinusoidal carrier signals of frequency f_c, the PSD is shifted to $\pm f_c$.*

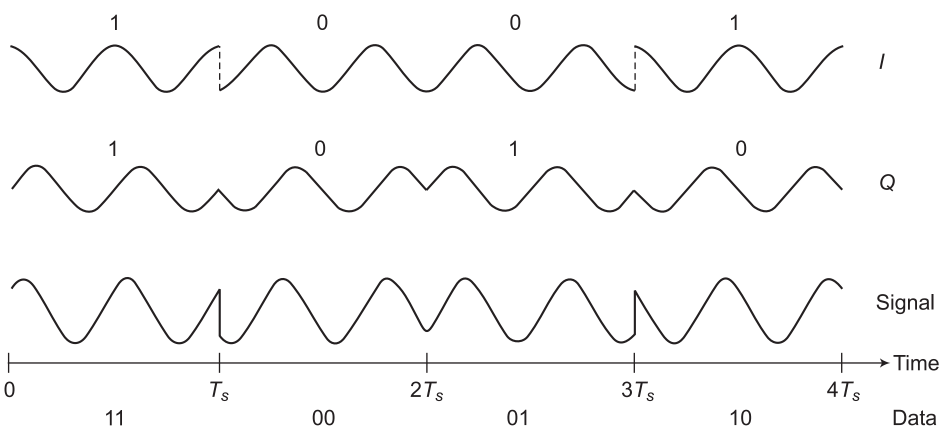


Figure 3.2.4 QPSK Signal in Time Domain *(Ts: Symbol duration)*

The binary data shown here is 1 1 0 0 0 1 1 0. The odd bits contribute to the in-phase (I-channel) component as **<u>1</u>** 1 **<u>0</u>** 0 **<u>0</u>** 1 **<u>1</u>** 0, and the even bits contribute to the quadrature-phase (Q-channel) component as 1 **<u>1</u>** 0 **<u>0</u>** 0 **<u>1</u>** 1 **<u>0</u>**. Note that there are abrupt changes in phase at some of the symbol-period boundaries.

The in-phase and quadrature channels of the coherent QPSK detector are typical BPSK coherent detectors.[13]

Figure 3.2.5 shows a simplified functional block schematic of coherent QPSK detector.

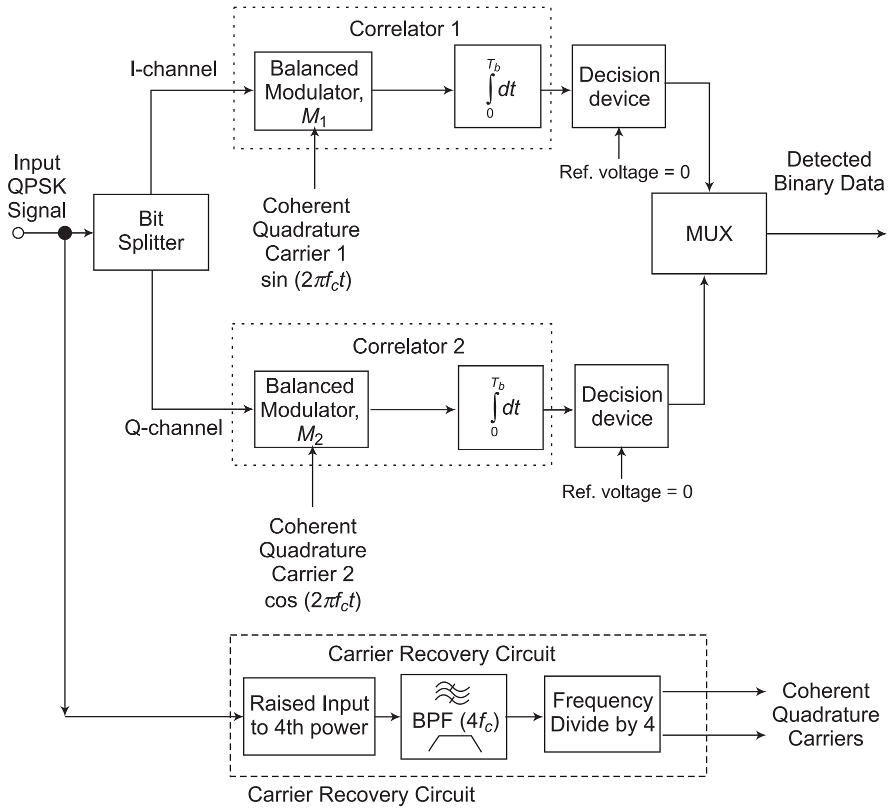


Figure 3.2.5 Functional Block Schematic of Coherent QPSK Detector

Functional Description of Coherent QPSK Detector

- The bit splitter directs the in-phase and quadrature channels of the input received QPSK signal to the *I* and *Q* synchronous correlators, respectively.
- The input received QPSK signal is also applied to the *carrier recovery circuit*, which reproduces the original transmit carrier signal which must be in frequency as well as phase coherence. The carrier recovery circuit is similar to that employed in BPSK except that the incoming signal is raised to the fourth power after which filtering recovers a waveform at four times the carrier frequency and finally frequency division by four regenerates the required carrier signals sin ($2\pi f_c t$) and cos ($2\pi f_c t$).
- The recovered coherent carrier signals are applied to two synchronous *correlators*, each comprising of *balanced modulator* and an *integrator*.
- The *integrator* circuits integrate the signals over two bit intervals, that is $T_s = 2T_b$.

[13] The bandwidth required by QPSK is one-half that of BPSK for the same BER. The transmission data rate in QPSK is higher because of reduced bandwidth. The variation in QPSK signal amplitude is not significant; hence, carrier power almost remains constant. A typical differential QPSK (DQPSK) uses phase shifts with respect to the phase of the previous symbol.

Coherent QPSK Detection

- If the output of the *correlator 1* is greater than reference voltage of 0 in the decision device, then the in-phase channel decision device decides that the detected bit is a binary 1, and if it is less than 0, then it decides that the detected bit is a binary 0.
- Similarly, if the output of the *correlator* 2 is greater than reference voltage of 0 in the decision device, then the quadrature channel's decision device decides that the detected bit is a binary 1, and if it is less than 0, then it decides that the detected bit is a binary 0.
- These binary digits from the outputs of the decision devices in the two channels are then multiplexed in the bit combining circuit (MUX).
- The final output is detected binary data of coherent QPSK detector which is an estimate of the transmitted binary data sequence with minimum possible probability of error for additive white Gaussian noise (AWGN) since it is a correlation reception.

SOLVED EXAMPLE 3.2.1 QPSK Symbol Rate and Bandwidth

In a QPSK digital communication system, the bit rate of a bipolar NRZ data sequence is 1 Mbps and carrier frequency of transmission is 100 MHz. Determine the symbol rate of transmission and the bandwidth requirement of the communications channel.

Solution Given bit rate of a bipolar NRZ data sequence, $f_b = 1$ Mbps or 1×10^6 bps

Therefore, bit period, $T_b = \dfrac{1}{f_b} = \dfrac{1}{1\text{ Mbps}} = 1\,\mu\text{s}$

In QPSK digital communication system, two successive binary bits form one symbol.

That is, the symbol duration, $T_s = 2T_b = 2\ \mu$s

Therefore, symbol rate of transmission $= \dfrac{1}{T_s} = \dfrac{1}{2\,\mu\text{s}} = 500$ k symbols/second **Ans.**

Minimum bandwidth, $B_{\text{QPSK}} = f_b = 1$ MHz **Ans.**

SOLVED EXAMPLE 3.2.2 QPSK Bandwidth Requirement

Consider a QPSK system having a bit rate of 9600 bps. Determine the bandwidth required by the QPSK signal using raised-cosine filter with roll-off factor of 0.35 and 0.5.

Solution We know that transmitted signal bandwidth = Symbol rate × (1 + roll-off factor)

Given bit rate of a bipolar NRZ data sequence, $f_b = 9600$ bps

In QPSK, symbol rate = ½ × bit rate = ½ × 9600 bps = 4800 symbols per second

(a) For given roll-off factor of 0.35, required bandwidth = 4800 × (1 + 0.35)
 Hence, required bandwidth = 6480 Hz **Ans.**

(b) For given roll-off factor of 0.5, required bandwidth = 4800 × (1 + 0.5)
 Hence, required bandwidth = 7200 Hz **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 3.2.3 For a QPSK modulator, how many symbols are possible at its output? What is the phase difference between each symbol?

3.2.2 Offset QPSK (OQPSK)

Recall

In the QPSK, the phase changes of the carrier signal may be by ±90°, or sometimes even by ±180°, depending upon whether the sign change of the in-phase and quadrature components occurs simultaneously or not. From the signal constellation diagram of QPSK, it can be seen that

- Whenever adjacent dibits in the binary data sequence differ only in one of the digits (e.g., 1 1 and 0 1, or 1 1 and 1 0, or 0 0 and 1 0, or 0 0 and 0 1), the transition involves a change of carrier phase by ±90° only.
- On the other hand, whenever adjacent dibits in the binary data sequence differ in both the binary digits (e.g., 1 1 and 0 0, or 0 1 and 1 0), the transition involves a change of carrier phase by ±180°.

Such sudden changes in the phase of the carrier signal can result in

- Reduction of the amplitude of the QPSK signal when it is passed through a low-pass filter during transmission (before detection).
- The resulting amplitude reduction of the QPSK signal can lead to errors during detection process.
- The changes in carrier phase by ±180° in particular, cause considerable reduction in the envelope amplitude and need to be avoided.
- Large amplitude fluctuation causes significant reduction in the desired quality in digital communication systems.

Need for OQPSK

Offset Quadrature Phase Shift Keying (OQPSK) is a variant of quadrature phase shift keying modulation using four different values of the phase to transmit. It is sometimes called Staggered Quadrature Phase Shift Keying (SQPSK). By offsetting the timing of the odd and even bits by one bit-period, or half a symbol-period, the in-phase and quadrature-phase components will never change at the same time. So, the carrier-phase changes are confined to ±90° only. The signal doesn't cross zero, because only one bit of the symbol is changed at a time. So the extent of amplitude fluctuations is reduced, thereby reducing the probability of occurrence of symbol errors in the detection process. Theoretically, the average probability of symbol error is exactly the same for QPSK and OQPSK for coherent detection.

Define

Offset QPSK (OQPSK) is a modified form of QPSK where the bit waveforms on the I and Q channels are offset or shifted in phase from each other by one-half of a bit interval. This means that OQPSK is obtained by introducing a shift or offset equal to one bit delay (T_b) in the bit stream for the quadrature component with respect to the in-phase component bit stream.

Mathematical Expression

Mathematically, the OQPSK signal can be expressed as

$$S_{\text{OQPSK}}(t) = \frac{1}{\sqrt{2}}\left[I(t)\cos(2\pi f_c t) - Q(t)\sin(2\pi f_c t)\right]$$

Interpretation

The changes in the I-channel occur at the midpoints of the Q-channel bits and vice versa. There is never more than a single bit change in the dibit code. There is never more than a 90° phase shift in the output phase. In other words, the modulated OQPSK signal transitions are ±90 degree maximum. It has no 180 degree phase shift and this result in reduction of out-of-band radiations. However, the abrupt phase transitions still remain.

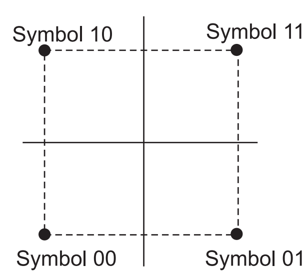


Figure 3.2.6 Constellation Diagram of OQPSK Signal

Interpretation

The changes in the output phase occur at twice the data rate in either *I* or *Q* channels.

Bandwidth Considerations

The minimum bandwidth is twice that of a QPSK for a given transmission data rate.

Figure 3.2.7 shows a simplified functional block schematic of a OQPSK modulator.

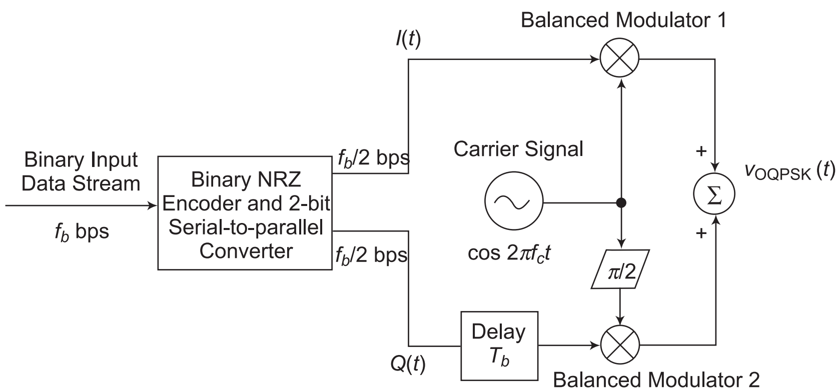


Figure 3.2.7 Functional Block Schematic of OQPSK Modulator

Functional Description of OQPSK Modulator

- The binary input data stream is applied to a binary NRZ encoder for line encoding.
- Then it is separated into odd bit and even bit.
- The quadrature data sequence will start with a delay of one bit period after the first odd bit is available in the in-phase data sequence.
- This delay is called *offset*, which leads to *offset QPSK*.
- The in-phase data sequence and carrier signal are applied to *I*-path balanced modulator.
- The quadrature data sequence and $\pi/2$-phase shifted version of carrier signal are applied to *Q*-path balanced modulator.
- Their outputs are then combined together to generate OQPSK signal.

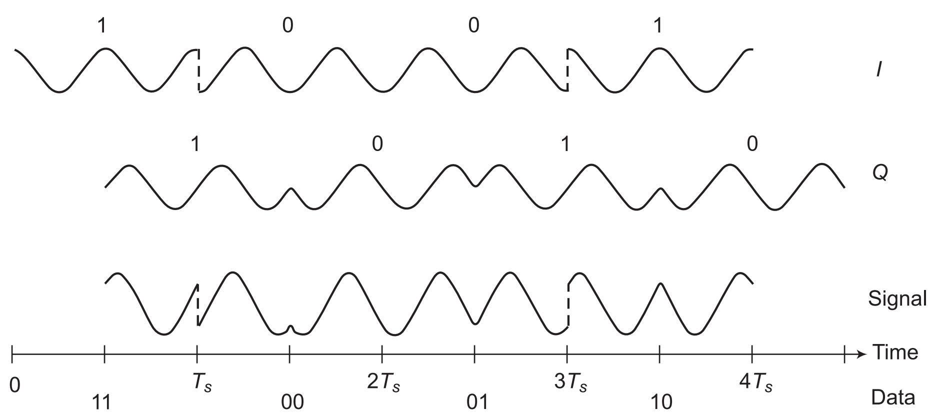


Figure 3.2.8 OQPSK Signal in Time Domain (Ts: Symbol duration)

The binary data shown here is 1 1 0 0 0 1 1 0. The odd bits contribute to the in-phase (I-channel) component as **1** 1 **0** 0 **0** 1 **1** 0, and the even bits with the half-period offset contribute to the quadrature-phase (Q-channel) component as 1 **1** 0 **0** 0 **1** 1 **0**. Note that there are abrupt changes in phase at some of the symbol-period boundaries but confined to ±90° only.

IMPORTANT! The modulated OQPSK signal has no 180° phase shift. Instead, it has ±90° maximum transition, which may result in reduction of out-of-band radiation. However, abrupt phase transitions are still present. Moreover, the minimum transmission bandwidth is twice that of QPSK for a given transmission data rate because the changes in the output phase occur at twice the data rate in either I or Q channels.[14]

3.2.3 π/4-Phase Shift QPSK

Recall In offset QPSK, the sudden change in the carrier-phase of ±90° (180° in conventional QPSK) lead to reduction in the amplitude of the envelope of the carrier and these can cause detection errors.

Define The $\pi/4$-*phase shift QPSK* can be regarded as a modification to QPSK and has carrier phase transitions that are restricted to $\pm\pi/4$ and $\pm 3\pi/4$ radians. So, it does not allow ±90° carrier-phase changes to occur.

Interpretation In fact, $\pi/4$-*QPSK* is a compromise between QPSK and OQPSK in terms of the allowed maximum phase transitions. Like OQPSK, the carrier phase does not undergo instantaneous 180° phase transition as well as no transition requires the signal amplitude to go through zero.

Here two separate constellations with identical Gray coding are shown but each is rotated by 45° ($\pi/4$ radians, hence the name $\pi/4$-QPSK) with respect to each other. Usually, either the even or odd symbols are used to select points from one of the constellations or the other symbols select points from the other constellation. This also reduces the phase-shifts from a maximum of 180°, but only to a maximum of 135° and so the amplitude fluctuations of $\pi/4$-QPSK are between OQPSK and conventional QPSK.

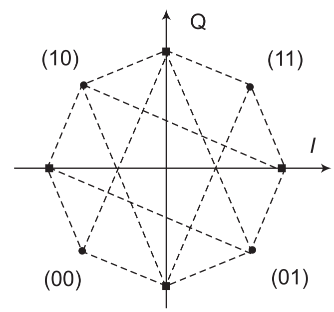


Figure 3.2.9 Constellation Diagram of $\pi/4$-QPSK Signal

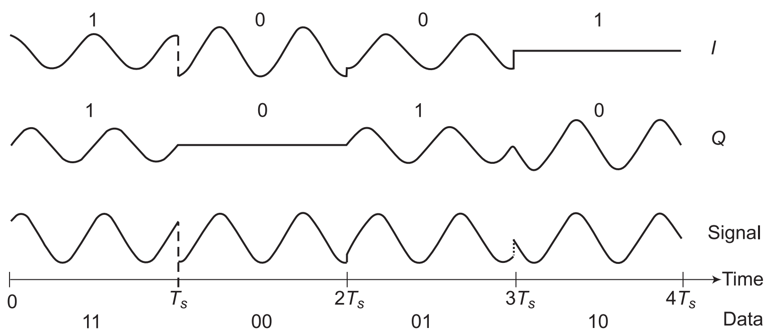


Figure 3.2.10 $\pi/4$–QPSK Signal in Time Domain (Ts: Symbol duration)

[14] OQPSK ensures that the $I(t)$ and $Q(t)$ signals have signal transitions at the time instants separated by $T_b/2$, where T_b is the bit period. Each bit in the in-phase or quadrature bit sequence will be held for a period of 2 T_b seconds. Thus, every symbol contains two bits ($T_s = 2\ T_b$). Theoretically, the average probability of symbol error is exactly the same for QPSK and OQPSK for coherent detection.

Actually, a $\pi/4$-QPSK signal constellation consists of symbols corresponding to eight phases. These eight phase points are formed by superimposing two QPSK signal constellations, offset by $\pi/4$ radians relative to each other. During each symbol period, a phase angle from only one of the two QPSK constellations is transmitted. The two constellations are used alternatively to transmit every pair of bits, called *dibits*. Thus, successive symbols have a relative phase difference that is one of four phases, namely +45°, +135°, –45°, and –135°. In $\pi/4$-QPSK, the set of constellation points are toggled each symbol, so transitions through zero cannot occur. This results into lower envelope variations as compared to that of QPSK and OQPSK.

Interpretation

It is observed that conventional QPSK has transitions through zero, that is, 180° phase transition. In offset-QPSK, the transitions on the *I*- and *Q*-channels are staggered in such a way that transitions through zero do not occur. Phase transitions are limited to 90°. The $\pi/4$-QPSK, unlike QPSK and OQPSK, can be detected non-coherently. Moreover, like QPSK signals, $\pi/4$-QPSK signals can be differentially encoded. All QPSK modulation schemes require linear power amplifiers. However, highly linear amplifiers are required in conventional QPSK.

IMPORTANT!

Table 3.2.2 provides a comprehensive comparison between conventional QPSK, offset QPSK, and $\pi/4$-QPSK digital modulation techniques.

Table 3.2.2 Comparison of QPSK, OQPSK, and $\pi/4$-QPSK

S. No.	Parameter	QPSK	OQPSK	$\pi/4$-QPSK
1.	Maximum phase shift	±180°	±90°	±45° or ±135°
2.	Amplitude variations at the instants of abrupt phase changes	Large	Small	Medium
3.	Simultaneous change of in phase and quadrature phase or even and odd bits	Yes	No	No
4.	Offset between in phase and quadrature phase or even and odd bits	No	Yes, by T_b seconds	Yes, by $T_b/2$ seconds
5.	Preferred method of demodulation	Coherent	Coherent	Coherent or non-coherent
6.	Minimum bandwidth	f_b	f_b	f_b
7.	Symbol duration	$2T_b$	$2T_b$	$2T_b$
8.	Receiver design complexity	Yes	Yes	No in case of non-coherent

In $\pi/4$-QPSK, there will always be a phase change for each input symbol. This enables the receiver to perform timing recovery and synchronization. The $\pi/4$-QPSK digital modulation technique is used for the North American TDMA cellular system, Personal Communication Services (PCS) system, and Japanese Digital Cellular (JDC) standards.

Application

Note... *Any number of phases may be used to construct a PSK constellation but 8-PSK is usually the highest order PSK constellation deployed. With more than 8 phases, the error-rate becomes too high and there are better, though more complex, modulations available such as quadrature amplitude modulation (QAM). Although any number of phases may be used, the fact that the constellation must usually* *deal with binary data means that the number of symbols is usually a power of 2 which allows an equal number of bits-per-symbol. Figure 3.2.11 shows constellation diagram for 8-PSK.*

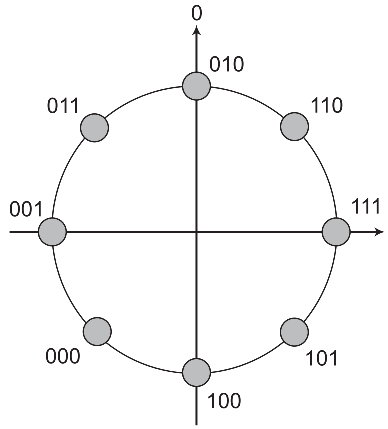


Figure 3.2.11 Constellation Diagram for 8-PSK

MATLAB simulation exercises on *M*-ary digital modulation techniques,

Scan the QR code given here OR visit: http://qrcode.flipick.com/index.php/129

ATTENTION QPSK and its variants are the most widely used digital modulation technique in modern digital wireless communication systems. This topic is important from examination point of view too.

3.2.4 Minimum Shift Keying (MSK)

Recall Coherent binary FSK receivers use the phase information contained in the received signal only for synchronization, not for improved detection.

Define *Minimum Shift Keying (MSK)* is a special case of binary continuous-phase FSK modulation technique in which the change in carrier frequency from symbol *0* to symbol *1* or vice versa is exactly equal to one-half the bit rate of input data signal.

Mathematical Expression As a form of binary FSK, the MSK signal can be expressed by

$$\boxed{v_{\text{MSK}}(t) = \begin{cases} \sqrt{\dfrac{2E_b}{T_b}} \cos\,[2\pi f_1\, t + \theta(0)] & \text{for binary 1} \\ \sqrt{\dfrac{2E_b}{T_b}} \cos\,[2\pi f_2\, t + \theta(0)] & \text{for binary 0} \end{cases}}$$

where E_b is the transmitted signal energy per bit, T_b is the bit duration, the phase $\theta(0)$ denotes the value of the phase at time $t = 0$.

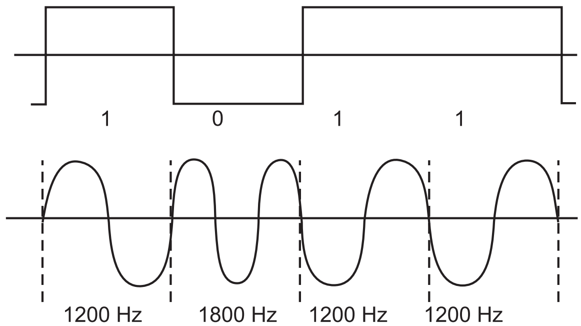


Figure 3.2.12 MSK Signal at 1200 baud for NRZ data 1 0 1 1

As an example, a 1200 baud baseband MSK data signal is composed of 1200 Hz and 1800 Hz frequencies for a binary 1 and 0, respectively. It can be seen that the modulated carrier contains no phase discontinuities and frequency changes occur at the carrier zero crossings only. Thus, MSK is a continuous phase modulation technique having modulation index ($\Delta f \times T_b$) = 0.5, where Δf is the difference in frequency for binary 1 and 0, and T_b is the bit duration.

Bandwidth of MSK Signal

The two frequencies f_1 and f_2 satisfy the following equations:

$$f_1 = f_c + \frac{1}{4T_b}; \text{ and } f_2 = f_c - \frac{1}{4T_b}$$

Therefore, bandwidth of MSK signal, $BW_{MSK} = |f_1 - f_2|$

IMPORTANT!

MSK signal produces orthogonal signaling and the phase information of the received signal is used in the detection process to improve the noise performance.

The PSD of MSK signal is shown in Figure 3.2.13.

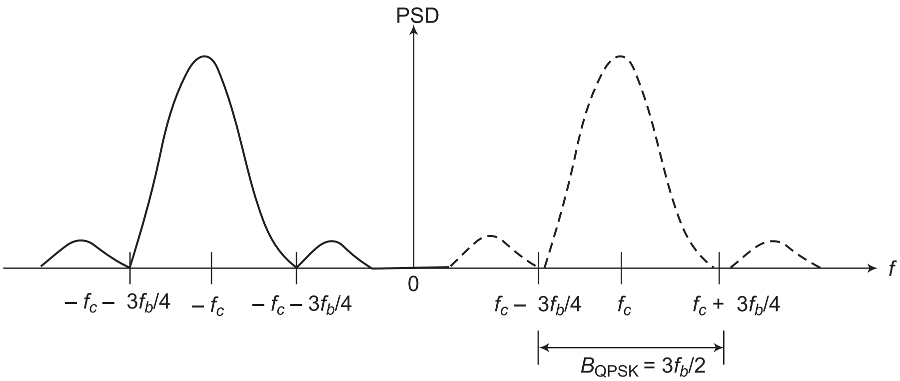


Figure 3.2.13 PSD of MSK Signal

Interpretation

The PSD of the MSK signal falls off as the inverse fourth power of frequency, as compared to the inverse square of the frequency in case of QPSK. Thus, the out-of-band radiation is less in case of MSK as compared to QPSK.

Offset MSK

MSK can also be viewed as a modified form of OQPSK, known as *Offset MSK* (*OMSK*) or "shaped QPSK". In MSK, the carrier signal is multiplied by a sinusoidal function. Mathematically, the MSK signal can then be expressed as

$$v_{\text{MSK}}(t) = I(t)\cos\left(\frac{\pi t}{2T_b}\right)\cos(2\pi f_c t) + Q(t - T_b)\sin(2\pi f_c t)$$

Significance of MSK

- MSK is derived from OQPSK by replacing the rectangular pulse with a half-cycle sinusoidal pulse.
- The in-phase and quadrature signals are delayed by intervals T_b from each other.
- The MSK signal has a constant envelope.
- For a modulation bit rate of f_b, the high-frequency, $f_1 = f_c + (0.25 f_b)$ when binary level is high (1), and the low-frequency, $f_2 = f_c - (0.25 f_b)$ when binary level is low (0).
- The difference frequency, $f_d = f_1 - f_2 = 0.5 f_b$, or $1/(2T_b)$, where T_b is the bit interval of the NRZ signal.
- MSK modulation makes the phase change linear and limited to $\pm\pi/2$ over a bit interval T_b. This enables MSK technique to provide a significant improvement over QPSK.

Difference between MSK and QPSK

- There is a similarity between the constellation of the QPSK system and the MSK system except that in MSK, there are only two possible phase changes at any one time depending on the initial phase value (as compared to that of QPSK in which there are four possible phase changes corresponding to one of the four dibits).
- In MSK, the baseband waveform that multiplies the quadrature carrier is much smoother than the abrupt rectangular waveform of QPSK.
- Although the PSD of MSK has a main lobe which is 1.5 times as wider as the main lobe of QPSK, the side lobes in MSK are relatively much smaller in comparison to the main lobe, making filtering much easier.
- The waveform of MSK signal exhibits phase continuity, that is, there are no abrupt phase changes as in QPSK.
- As a result, the intersymbol interference caused by nonlinear amplifiers are lower in MSK as compared to that of in QPSK.[15]

SOLVED EXAMPLE 3.2.4 **MSK Bandwidth Requirement**

Consider a digital message having a data rate of 8 kbps. Compute the transmission bandwidth required if the message is transmitted through MSK digital modulator.

Solution Input data rate of digital message, f_b = 8 kbps (Given)

In MSK, transmission bandwidth = 1.5 f_b

Hence, required transmission bandwidth = 1.5 × 8 kbps = 12 kHz **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 3.2.5 For a digital message having a data rate of 4 kbps, show that transmission bandwidth required of MSK signal is 6 kHz.

[15] It turns out that in MSK, 99% of the signal power is contained in a bandwidth of about 1.2 f_b while in QPSK, the corresponding bandwidth is about 8 f_b. We can also say that QPSK signals are in time quadrature while MSK signals are in frequency quadrature.

MSK and Gaussian MSK (GMSK) are particular cases of continuous phase modulation (CPM). Indeed, MSK is a particular case of the sub-family of CPM known as continuous-phase frequency-shift keying (CPFSK) which is defined by a rectangular frequency pulse (i.e., a linearly increasing phase pulse) of one symbol-time duration.

Application

3.2.5 Gaussian Minimum Shift Keying (GMSK)

Recall

MSK is a continuous phase modulation scheme where the modulated carrier contains no phase discontinuities and frequency changes occur at the carrier zero crossings. MSK is unique due to the relationship between the frequencies of a logical 0 and 1: the difference between the frequency of a logical 0 and a logical 1 is always equal to half the data rate. In other words, the modulation index is 0.5 for MSK.

Why GMSK?

The fundamental problem with MSK is that the spectrum is not compact enough to realize data rates approaching the RF channel bandwidth. A plot of the spectrum for MSK reveals side lobes extending well above the data rate. For wireless data transmission systems which require more efficient use of the RF channel bandwidth, is necessary to reduce the energy of the MSK upper side lobes. As we have seen that the power spectral density of MSK falls off as the fourth power of the frequency. But it is still not adequate for multi-user wireless communication applications, which require very stringent specifications with regard to adjacent channel interference.

- So the PSD of an MSK signal is modified by passing the NRZ binary data stream through a pulse-shaping low-pass filter such as Gaussian filter before carrier modulation.
- The *Gaussian filter* has a narrow bandwidth with sharp cut-off characteristics and negligible overshoot in its impulse response.

In addition, this filter should permit the carrier-modulated signal in that it has a carrier phase of either 0 or π radians at even multiples of T_b, and either $+\pi/2$ or $-\pi/2$ radians at odd multiples of T_b.

Therefore, the modified MSK using such a Gaussian filter, is called 'Gaussian-filtered MSK, or simply '*Gaussian MSK* (GMSK)'.

Define

Gaussian Minimum Shift Keying (GMSK) is a special case of MSK in which a pre-modulation low-pass Gaussian filter is used as a pulse-shaping filter to reduce the bandwidth of the baseband signal before it is applied to the MSK modulator.

Significance of using GMSK

The use of a filter with Gaussian characteristics with the MSK approach achieves the requirement of

- reduction in the transmitted bandwidth of the signal
- uniform envelope
- spectral containment, i.e., reduction of the side lobe levels of the power spectrum
- reduction in adjacent channel interference
- suppression of out-of-band noise

Bandwidth of using GMSK Signal

The relationship between the pre-modulation filter bandwidth $f_{3\text{dB}}$ and the bit period, T_b defines the bandwidth of the system. That is,

$$B \times T_b = \frac{f_{3dB}}{\text{Bit Rate}}$$

Hence, for a bit rate = 9.6 kbps and $(B \times T_b) = 0.3$, the 3 dB cut-off frequency of Gaussian filter is 2880 Hz. It may be noted that

- If $f_{3\text{dB}} > 1/T_b$, then the output signal waveform is essentially MSK.

- If $f_{3dB} < 1/T_b$, then the intersymbol interference occurs since each data pulse overlaps with the data pulse in the next position in the symbol duration.

We know that the product of bandwidth and bit duration (BT_b) plays an important role in the shape of the response of the pulse-shaping filter.

Since the pulse-shaping Gaussian filter acts on the input binary data which has been encoded in the form of polar NRZ waveforms, and is further used for frequency-shift keying, the shape of the filter response varies for different values of (BT_b) product.

The pre-modulation Gaussian filtering introduces marginal intersymbol interference in the finally modulated signal. However, the degradation in performance can be minimized by selecting the appropriate 3 dB bandwidth-bit duration product (BT_b) of the filter.

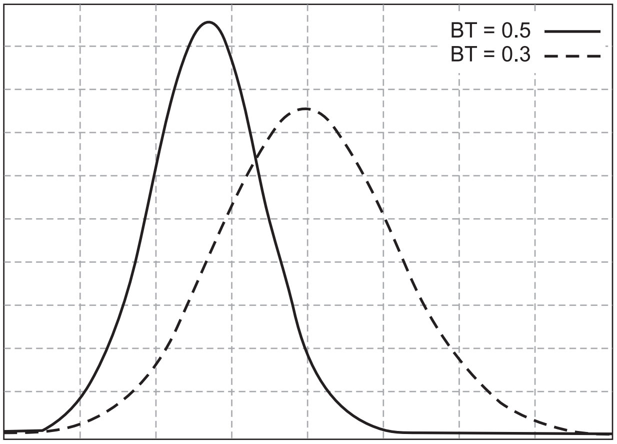


Figure 3.2.14 Gaussian Filter Impulse Response for $B \times T_b$ (or, BT) = 0.3 and 0.5

PSD of GMSK Signal

For wireless data transmission systems, which require more efficient use of the RF channel, it is necessary to reduce the energy of the side lobes. We have discussed that the spectrum of MSK has side lobes which extend well above the data rate. The use of pre-modulation low-pass Gaussian filter having a narrow bandwidth with a sharp cut-off frequency, the side lobe energy is reduced. The resultant modulation scheme is known as Gaussian MSK.

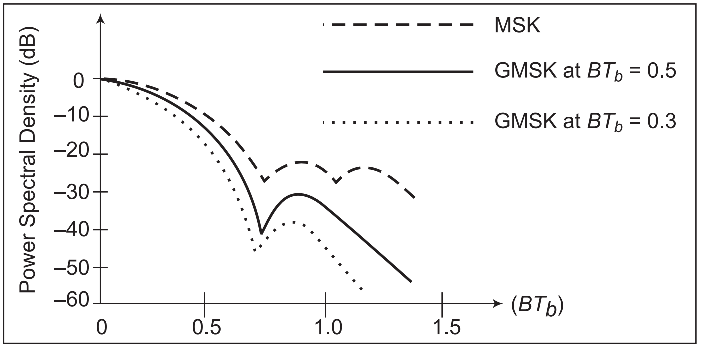


Figure 3.2.15 PSD Comparison of MSK and GMSK

The PSD of MSK signal has low side lobes because of the effect of the linear phase change. This enables to control adjacent channel interference. The power spectrum of GMSK with a (BT_b) value of infinity is equivalent to that of MSK. As the (BT_b) product decreases, the side lobe levels fall off very rapidly. For example, for a (BT_b) = 0.5, the peak of the second power lobe is more than 30 dB below the main power lobe. However, reducing (BT_b) beyond a certain value, increases the irreducible error rate generated by the Gaussian filter due to ISI. For (BT_b) = 0.3, adjacent bits or symbols will interfere with each other more than for (BT_b) = 0.5. In other words, MSK does not intentionally introduce ISI. But greater ISI allows the spectrum to be more compact, making demodulation more difficult. Hence, spectral compactness is the primary trade-off while selecting Gaussian pre-modulation filtered MSK. It implies that channel spacing can be closer for GMSK as compared to that for MSK for the same adjacent channel interference.

Interpretation

The channel spacing can be tighter for GMSK when compared to MSK for same adjacent channel interference.

IMPORTANT!

Considering GMSK as a special case of FSK, it can be generated simply using a frequency modulator. Also considering GMSK as a special case of PSK, it can be generated simply using phase modulator. Both types of GMSK modulators are briefly described here.

GMSK Modulator

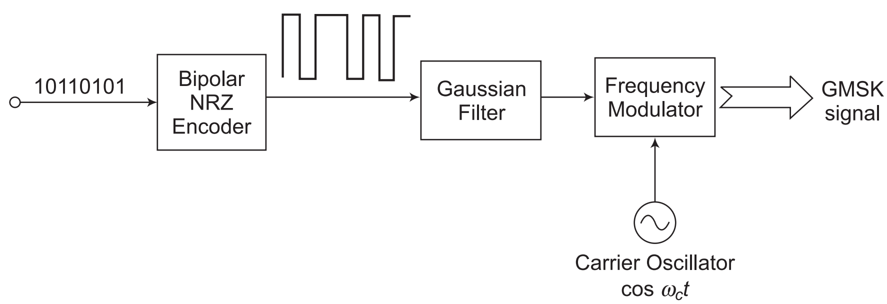


Figure 3.2.16 GMSK Modulator using Frequency Modulator

- The binary data sequence is encoded using a bipolar NRZ encoder.
- The resulting data stream is then applied through a Gaussian low-pass filter whose characteristics are Gaussian in nature.
- The filtered signal acts as modulating signal which modulates the carrier signal in frequency modulator.
- The output of the frequency modulator is a GMSK signal.

Functional Desrciption of GMSK Modulator using Frequency Modulator

- The bipolar NRZ encoded information binary digital data sequence is converted into two parallel data sequences using a serial-to-parallel converter.
- The resulting data streams are then applied to in-phase path and quadrature path respectively, each comprising of an integrator and Gaussian filter.
- Their outputs are applied to independent balanced modulators whose carrier signals are having a phase shift of 90° with each other. This operation is similar to phase modulation.
- Both outputs are added to produce the desired GMSK signal.

Functional Description of GMSK Modulator using Phase Modulator (Refer Fig. 3.2.17)

It may be noted that in GMSK, each transmitted symbol spans several bit periods.

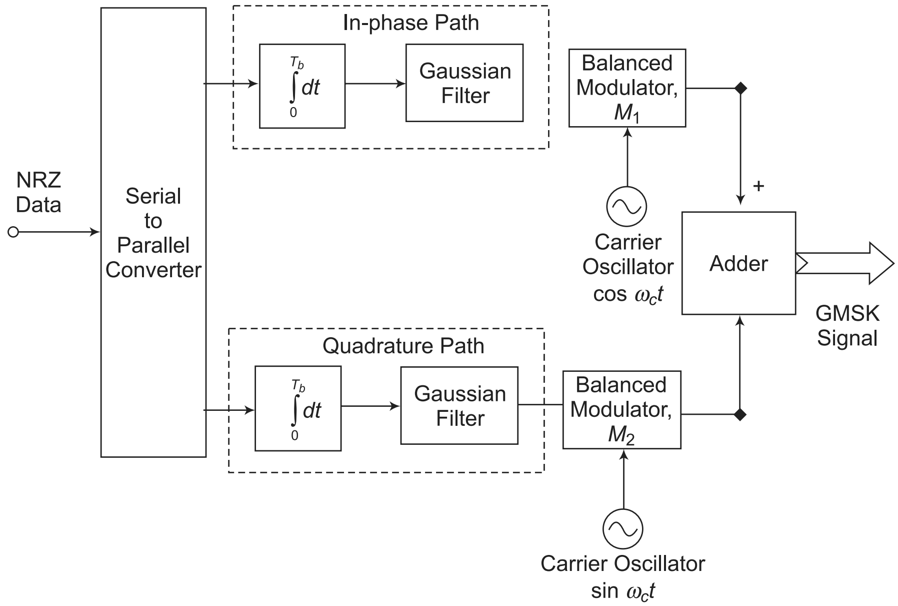


Figure 3.2.17 GMSK Modulator using Phase Modulator

GMSK Demodulator GMSK can be non-coherently detected as in FSK demodulator, or coherently detected as in MSK demodulator.

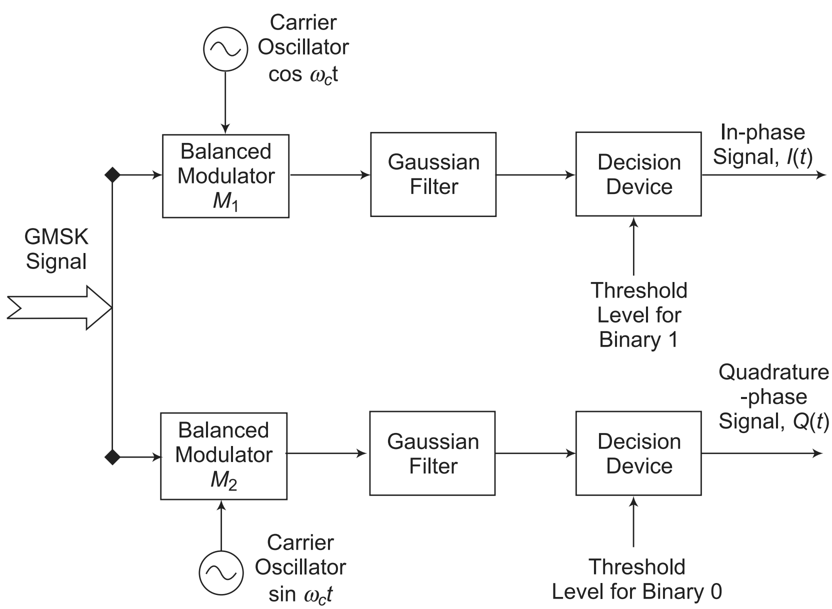


Figure 3.2.18 GMSK Demodulator

Functional Description of GMSK Demodulator

- The received GMSK signal is applied to two balanced modulators, whose carrier signals have a phase shift of 90° with each other.
- The output of balanced modulator is applied to Gaussian low-pass filter.
- The detection of bit 1 or 0 is done by the decision device.

Application

GMSK provides high spectrum efficiency, excellent power efficiency, and a constant amplitude envelope. It allows Class C power amplifiers for minimum power consumption. GMSK is widely used in the GSM cellular radio and PCS systems. A choice of $(BT_b) = 0.3$ in GSM is a compromise between BER and out-of-band interference since the narrow filter increases the ISI and reduces the signal power.

SOLVED EXAMPLE 3.2.6 **Gaussian Filter Pulse Waveforms**

In GMSK signal, low values of (BT_b) product create significant Intersymbol Interference (ISI). If (BT_b) product is less than 0.3, some form of combating the ISI is required. Illustrate it with the help of GMSK pulse waveform in comparison to that of MSK pulse waveform.

Solution In MSK, the value of (BT_b) product is infinity, whereas it is 0.5 or less in case of GMSK.

Figure 3.2.19 depicts the GMSK pulse shapes and intersymbol interference (ISI).

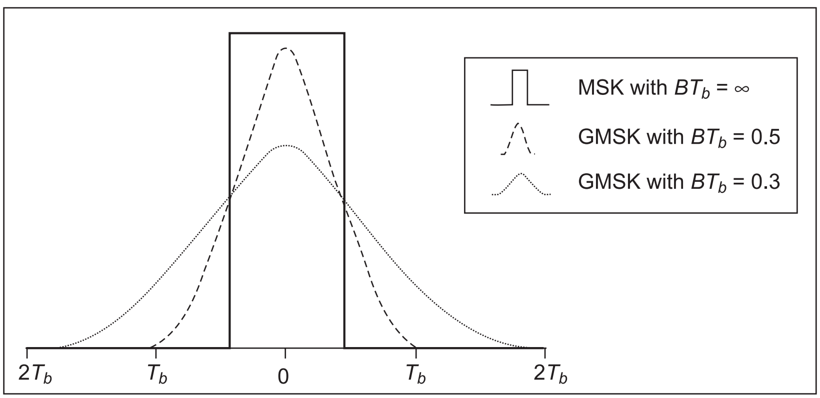


Figure 3.2.19 Gaussian Filter Waveforms

It is observed that

- GMSK with BT_b = infinity (∞) is equivalent to MSK.
- Gaussian filter's response for $BT_b = 0.3$ shows that a bit is spread over approximately 3 bit periods.
- Gaussian filter's response for $BT_b = 0.5$ shows that a bit is spread over approximately 2 bit periods.

This means that adjacent symbols will interfere with each other (intersymbol interference) for $BT_b = 0.3$ more than that for $BT_b = 0.5$. Therefore, it is concluded that in GMSK, low values of (BT_b) product create significant Intersymbol interference (ISI).

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 3.2.7 The 3 dB bandwidth of Gaussian filter is given by $f_{3-dB} = (BT_b)f_b$, where f_b is the bit rate in bps, and T_b is the bit duration in seconds. Calculate the 3-dB cut-off frequency of the Gaussian filter (since Gaussian filter is a low-pass filter, its 3 dB bandwidth is same as that of its 3 dB cut-off frequency) for $BT_b = 0.5$ and a data rate of 9600 bps.

3.2.6 Quadrature Amplitude Modulation (QAM)

Recap

In multilevel or M-ary digital modulation techniques, one of the M possible signal elements are transmitted during each bit interval of T_b seconds. The number of possible signal levels, M is given by $M = 2^n$, where n is an integer which represents the number of bits in each symbol.

In *baseband M-ary modulation* (normally called *baseband M-ary signaling*), the M signal elements are distinguished by M different pulse amplitude levels (e.g., M-ary PAM). In *bandpass M-ary digital modulation*, the M signal elements can differ in their amplitude levels, as in the case of *M-ary ASK*, or in their frequencies, as in the case of *M-ary FSK*, or in their phases, as in the case of *M-ary PSK.*

What is QAM?

In *bandpass M-ary digital modulation* in hybrid form, the M signal elements can differ in their amplitude levels as well as in their phases, resulting in *M-ary quadrature amplitude modulation* (*QAM*).

In M-ary ASK the average transmitted power sets a limit to the achievable transmission rate while maintaining a specified probability of error. We have seen that in BPSK, QPSK and even M-ary PSK, all the signals transmitted during a bit interval have the same amplitude and they all differ in their phase.

It is quite obvious that there is an advantage to have amplitude difference as well as the phase difference between the M possible signals of an M-ary modulation systems because the detection error in the presence of noise depends on the distance between any two adjacent message points.

This leads to a hybrid type of M-ary bandpass amplitude and phase modulation techniques, known as M-ary quadrature amplitude modulation (or, simply *QAM*), also known as M-ary quadrature amplitude and phase shift keying (*QAPSK*, or simply *QASK*).

Define

Quadrature Amplitude Modulation (QAM), also called *Quadrature Amplitude Shift Keying* (*QASK*), is a form of digital modulation technique similar to PSK except that the digital data is contained in both the amplitude and the phase of the modulated signal.

Interpretation

QAM can either be considered a logical extension of QPSK or a combination of ASK and PSK. In QAM/QASK, two carrier signals of the same frequency but in-phase quadrature (i.e., the phase of one carrier signal is shifted by the other by 90°) are independently amplitude modulated by discrete amplitude levels of input data stream.

Mathematical Expression

The QAM signal can be expressed as

$$S_{QAM}(t) = d_1(t) \cos (2\pi f_c t) + d_2(t) \sin (2\pi f_c t)$$

where $d_1(t)$ and $d_2(t)$ represents the binary data (0 and 1).

An Example of 16-ary QAM

As an example of a QAM system, let us consider to transmit a symbol for every 4 bits. There are then $2^4 = 16$ different possible symbols which shall have to be able to generate 16 distinguishable signals. We may view the M message points in the two-dimensional signal constellation of the 16-ary QAM with $M = 16$ as having been generated by the Cartesian product of the two coordinate sets of the message points of an 4-ary ASK with one carrier and another 4-ary ASK with the other carrier.

Thus, the ASK and PSK are combined in such a way that the positions of the signaling elements on the constellation diagram are optimized to achieve the largest possible distance between elements. This reduces the likelihood of one element being misinterpreted as another element. This enables to reduce the probability of occurrence of errors. For a given system, there are finite numbers of allowable amplitude-phase combinations.

- If 2-level ASK is used then the combined data stream can be in one of the possible 4 states. This is essentially *QPSK*.
- If 4-level ASK is used then the combined data stream can be in one of the possible 16 states. It is called *16-QAM*.

Clearly, more the number of states, higher is the data rate that can be supported within a given bandwidth. But probability of occurrence of error also increases.

IMPORTANT!

The complexity to implement M-ary QAM also increases and requires the use of linear amplifiers. We can say that the M-ary QAM having more than 16 levels are more efficient in terms of transmission bandwidth than BPSK, QPSK or 8-PSK, but it is also more susceptible to noise due to amplitude variations.

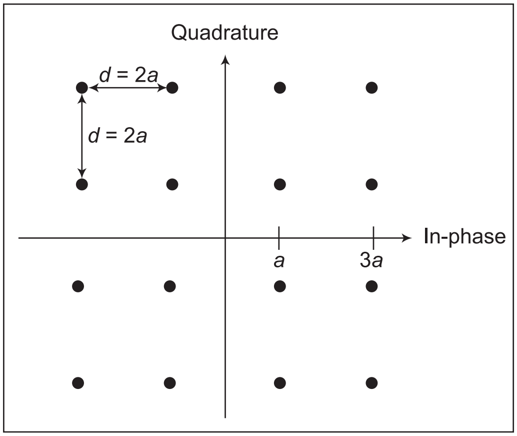


Figure 3.2.20 16-QAM Signal Constellation Diagram

Interpretation

There are four different amplitude levels, two each on positive and negative sides. These four amplitude levels are distributed in 12 different phases, 3 in each quadrant. Thus, each transmitted symbol represents four bits. In this configuration each signal point is equally distant from its nearest neighbors, the distance being $d = 2a$.

Bandwidth of M-ary QAM Signal

Let us assume that all 16 signals are equally likely and symmetrically placed about the origin of the signal space. The bandwidth of M-ary QAM signal is given by

$$B_{M\text{-ary QAM}} = \frac{2}{n} \times f_b$$

where n is the number of bits and f_b is the input bit rate.

- For $M = 8$, using $M = 2^n$, we get $n = 3$. Therefore, $B_{8\text{-QAM}} = \frac{2}{3} \times f_b$
- For $M = 16$, using $M = 2^n$, we get or $n = 4$, $B_{16\text{-QAM}} = \frac{1}{2} \times f_b$ [16]

QAM Modulator Figure 3.2.21 shows the simplified functional block schematic of QAM modulator.

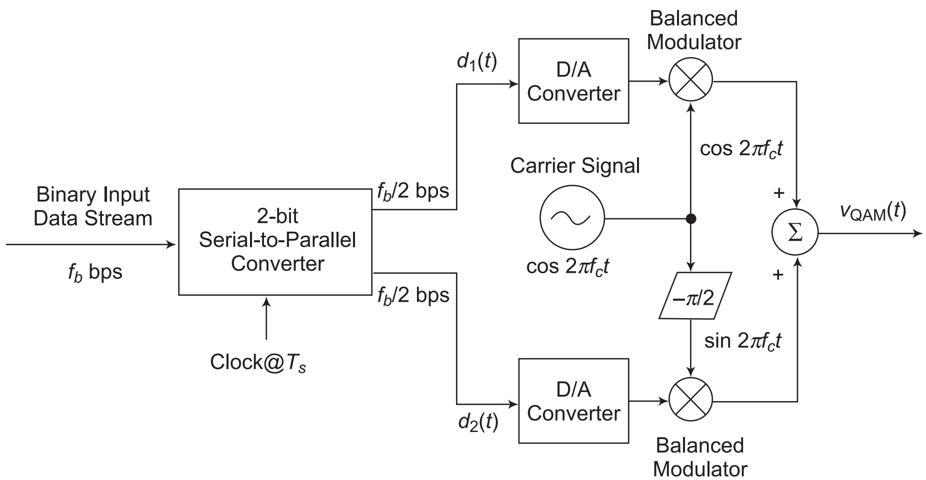


Figure 3.2.21 QAM Modulator

Functional Description of QAM Modulator

- The input is a data stream of binary digits at a rate of f_b bps.
- This data stream is converted into two separate data streams of $f_b/2$ bps each, by taking alternate bits for the two data streams.
- One data stream is ASK modulated on a carrier frequency f_c.
- Other data stream is ASK modulated by the same carrier signal shifted by 90°.
- The two modulated signals are then added and transmitted.

QAM Coherent Demodulator Figure 3.2.22 shows the functional block schematic of QAM coherent demodulator. That is similar to the QPSK coherent demodulator.

Functional Description of Coherent QAM demodulator

- A local set of quadrature carrier signals is recovered for synchronous detection by raising the received signal to the fourth power.
- The component at frequency $4f_c$ is extracted using a narrow bandpass filter tuned at $4f_c$ and then dividing the frequency by 4.
- The available quadrature carrier signals are applied to two correlators, each comprising of a balanced modulator and an integrator.
- The integration is done over time interval equal to the symbol time T_s.
- Symbol synchronizers can be used for synchronization.

[16] The number of levels could be increased indefinitely with a noiseless channel, but a practical limit is reached when the difference between adjacent levels becomes too small. It becomes difficult to detect reliably in the presence of noise and signal distortion.

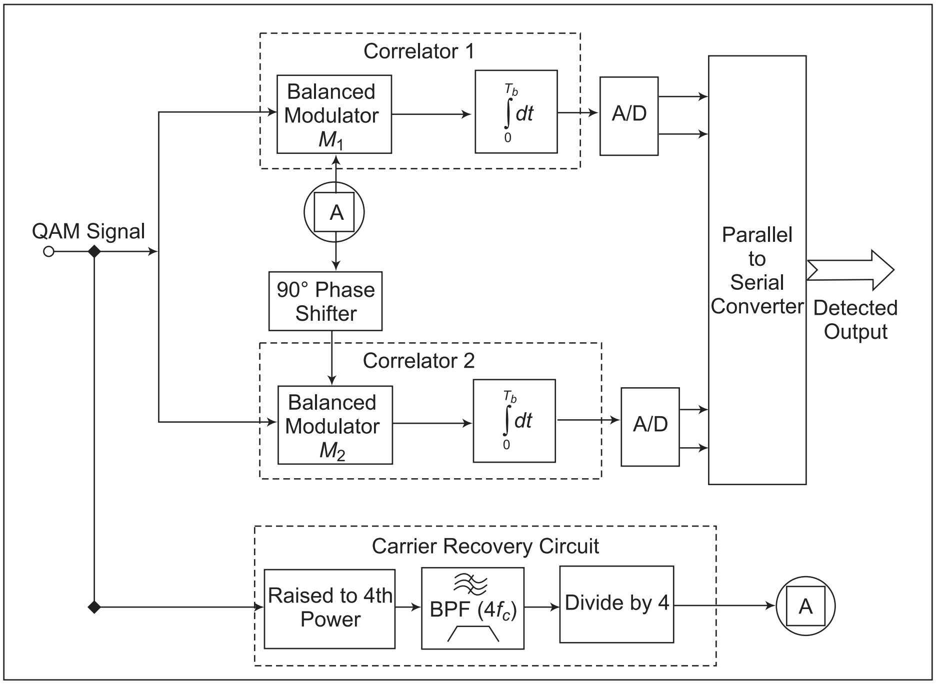


Figure 3.2.22 QAM Coherent Demodulator

- The original information bits are then recovered by using A/D converter and parallel-to-serial converter.

Application

QAM is an efficient way to achieve high data rates with a narrowband channel by increasing the number of bits per symbol, and uses a combination of amplitude and phase modulation. Higher level QAM (64-QAM and 256-QAM) are more bandwidth efficient and are used for high data rate transmission applications in fixed terrestrial microwave digital radio, digital video broadcast cable, and modems.

SOLVED EXAMPLE 3.2.8 **Bits per Symbol in QAM**

A QAM modulator uses 4 different amplitudes and 16 different phase angles. How many bits does it transmit for each symbol?

Solution We know that the number of possible levels per symbol is the multiplication of the number of different amplitude levels and number of different phase angles in QAM.

For given 4 different amplitudes and 16 different phase angles,

The number of possible states per symbol, $M = 4 \times 16 = 64$

The number of bits per symbol, $n = \log_2 M$

Hence, the number of bits transmitted for each symbol, $n = \log_2 64 = 6$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 3.2.9 A QAM modulator converts groups of 6 bits into symbols. Determine the symbol rate and the number of different symbols if the data rate is 12000 bps.

Self-Assessment Exercise linked to LO 3.2

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/130

Q3.2.1 List some of the key aspects of QPSK digital modulation.

Q3.2.2 Bit splitter is basically a shift-register with controlled output. Describe the functional block schematic of QPSK modulator with implementation of bit splitter at the input of balanced modulators.

Q3.2.3 Outline the possible reasons for higher probability of errors in QPSK and minimum probability of errors in offset QPSK.

Q3.2.4 Highlight the unique feature in $\pi/4$-QPSK as compared to BPSK and QPSK.

Q3.2.5 For an 8-PSK modulator with an analog carrier signal frequency of 70 MHz and an input bit rate of 10 Mbps, design the minimum double-sided Nyquist bandwidth requirement.

Q3.2.6 For a 16-QAM modulator with an analog carrier signal of 70 MHz and an input bit rate of 10 Mbps, calculate the minimum double-sided Nyquist bandwidth.

If you have been able to solve the above exercises then you have successfully mastered

LO 3.2: *Describe M-ary digital modulation techniques including QPSK, Offset QPSK, π/4-QPSK, MSK, GMSK, and QAM.*

Mid-Chapter Check

So far you have learnt the following:

- Binary Digital Modulation Techniques such as ASK, FSK and PSK
- M-ary Digital Modulation Techniques such as QPSK, OQPSK, π/4-QPSK, MSK, GMSK and QAM

Therefore, you are now skilled to complete the following tasks:

MQ3.1 Implement the operation of binary ASK, FSK and PSK for the binary data 1 1 0 0 1 0 1.

MQ3.2 Compare and contrast the bandwidth requirements of binary ASK, FSK and PSK signals.

MQ3.3 Draw the PSD waveforms of binary ASK, FSK and PSK signals and interpret them to bring out the pros and cons of these digital modulation techniques.

MQ3.4 Distinguish between coherent and non-coherent FSK detection.

MQ3.5 "Coherent detection of DBPSK signal is said to be more reliable". Justify this statement with the help of suitable example data depicting clearly as how a DBPSK receiver decode the incoming symbols into binary bits.

LO 3.1

MQ3.6 List and differentiate various types of QPSK modulation techniques.

MQ3.7 For a QPSK modulator, determine the symbol rate at its output for an input data rate of 3600 bps.

MQ3.8 In a digital communication system, the bit rate of a bipolar NRZ data sequence is 1 Mbps and carrier frequency of transmission is 100 MHz. Determine the symbol rate of transmission and the bandwidth requirement of the communications channel for (a) 8-ary PSK system; and (b) 16-ary PSK system.

MQ3.9 Is GMSK better than conventional FSK? Give the significance of (BT_b) product in GMSK.

MQ3.10 For an 16-QAM modulator with an input bit rate of 10 Mbps and an analog carrier frequency of 60 MHz, determine the minimum double-sided Nyquist bandwidth.

LO 3.2

Sure of what you have learnt so far? For answers, scan the QR code

OR

visit http://qrcode.flipick.com/index.php/125

Be ready for the next sections!

LO 3.3

3.3 COHERENT AND NON-COHERENT DETECTION

Recall

We have earlier discussed coherent and non-coherent detectors/demodulators for various digital modulation techniques including ASK, FSK, PSK, QPSK, QAM, and M-ary PSK. When the receiver exploits knowledge of the incoming carrier signal's phase to detect the digitally modulated signals, the process is called *coherent detection*. Whereas when the receiver does not utilize the knowledge of the incoming carrier signal's phase as reference to detect the digitally modulated signals, the process is called *non-coherent detection*. Thus, coherent systems need carrier phase information at the receiver and they use matched filters to detect and decide what data was sent, while non-coherent systems do not need carrier phase information and use methods like square law to recover the data.

What We Discuss Here

The concept of coherent and non-coherent detection of digitally bandpass modulated signals.

- Various methods of carrier signal recovery from digital modulated received signals.
- Clock signal recovery approach to obtain symbol synchronization for coherent detection of digital modulated signals.

Coherent Detection

In ideal *coherent detection* at the receiver, a prototype of each possible received signal is available and the receiver multiplies and integrates (correlates) the incoming signal with each of its prototype replicas. This implies that the receiver is phase-locked to the incoming signal. Practically, the local carrier signal generated at the receiver is phase-locked with the carrier signal at the transmitter. It is also called *synchronous detection*, which implies that the detection is carried out by correlating received noisy signals and locally generated carrier signal.

Non-Coherent Detection

Non-coherent detection refers to systems employing demodulators that are designed to operate without knowledge of the absolute value of the phase of the incoming carrier signal. So phase estimation is not required. Although non-coherent systems have reduced complexity in comparison to coherent systems but probability of error is more. Practically, the non-coherent detection process does not require locally generated receiver carrier signal to be phase locked with transmitter carrier signal. It is simpler in design but error probability increases.

Differential Coherent Detection

There is another variant of coherent detection, known as *differential-coherent detection*, in which the phase information of the prior symbol is utilized as a phase reference for detecting the current symbol. Thus, it does not require a reference in phase with the received carrier signal.

How is Coherent Detection possible?

The process of making coherent detection possible and then maintaining it throughout the reception of digital modulated signals with optimum error performance is known as *carrier recovery* or *carrier synchronization*. In an ideal form of coherent detection, exact replica of the received signals is available at the receiver. It implies that the receiver has complete knowledge of the phase reference of the carrier signal used at the modulator. In such a case, the receiver is said to be phase-locked to its transmitter. Coherent detection is performed by cross-correlating the possible incoming received signal with each one of its replica. Then a decision is made based on comparisons with pre-defined threshold levels.

Methods of Carrier Recovery

There are two basic methods of carrier recovery:

- Knowledge of frequency as well as the phase of the carrier signal is necessary. This method is called *carrier signal recovery*, or *carrier synchronization*.
- The receiver has to know the instants of time at which the modulation has changed its state so as to determine the instant when to sample. This method of marking the beginning

and end times of the individual symbols is called *clock signal recovery,* or *symbol synchronization*.[17]

M^{th} Power Loop Carrier Recovery

The synchronization loop that produces coherent reference signals that are independent of the modulation is a M^{th} power loop carrier recovery.

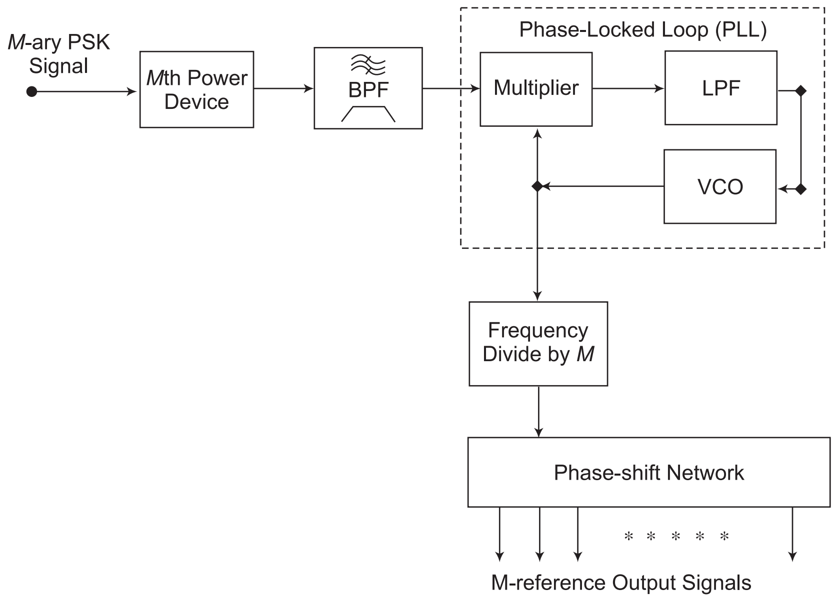


Figure 3.3.1 M^{th} Power Loop Carrier Recovery

Functional Description

Step I The received signal is raised to the M^{th} power before passing it through a bandpass filter which is used to minimize the effects of noise.

Step II The M^{th} harmonic of the carrier thus produced is then tracked by a Phase-Locked Loop (PLL).

Step III A PLL consists of a VCO, a loop filter, and a multiplier that are connected together in the form of a negative feedback system.

Step IV The resulting sinusoidal output is applied to a frequency divide-by-*M* circuit that yields the first reference cosine signal.

Step V By applying it to a 90° phase shift network, the second reference sinusoidal signal is obtained.

Step VI Thus, *M*-reference output signals are obtained for carrier synchronization purpose.

Costas-Loop Carrier Recovery

The Costas loop, named after its inventor, is an alternative method by which the carrier of a BPSK (or even QPSK) received signal may be recovered. This method works on the principle of phase locked loop (PLL) which mainly comprises of a Voltage Controlled Oscillator (VCO) and the loop filter. It may be recalled that PLL has the ability to lock to the frequency of the incoming signal, track it and smoothen frequency or phase noise. In general, the PLL is a

[17] Carrier signal recovery and clock signal recovery methods of synchronization can occur either together, or sequentially one after another. Carrier signal recovery method is primarily used for coherent detection of digital modulated signal. Naturally, carrier signal recovery is of no concern in a non-coherent detection.

feedback mechanism by which phase error between an input signal and locally generated signal is minimized. The Costas loop circuit involves two PLLs employing a common VCO and loop filter.

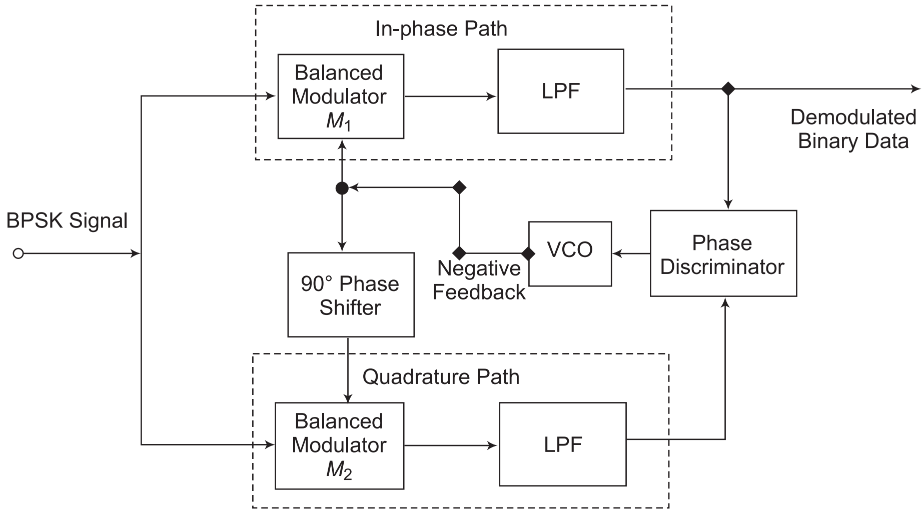


Figure 3.3.2 Costas-Loop Carrier Recovery

Functional Description of Costas Loop

The Costas-loop carrier recovery method consists of in-phase and Quadrature paths. These are coupled together via a common VCO and loop filter to form a negative feedback loop.

- The received BPSK signal is applied at the inputs of balanced modulators M_1 and M_2 of in-phase and Quadrature paths, respectively.
- The low pass filters (LPF) used in in-phase and quadrature paths help to remove the double frequency terms generated in the balanced modulators.
- The outputs of both low-pass filters are applied to phase discriminator which produces the signal corresponding to the received signal which is making transitions at random times between ±1.
- The input signal to the VCO and loop filter will serve to keep the VCO oscillating at the carrier frequency.
- The VCO output is directly applied to balanced modulator M_1 of in-phase path whereas it is 90° phase shifted before applying to balanced modulator M_2 of quadrature path as their carrier signals.
- When synchronization is attained, the demodulated data waveform is available at the output of the in-phase path.

Clock Signal Recovery

Clock signal recovery is important in digital communication to generate important timing information and bit or symbol synchronization. It can be processed either along with carrier signal recovery or independently. There are various approaches to obtain symbol synchronization for coherent detection of digital modulated signal such as by transmitting a clock signal along with modulated signal in multiplexed form.

At the receiver, the clock signal is extracted by appropriate filtering of the received modulated signals. Thus, the time required for carrier/clock signal recovery is minimized. However, a fraction of the transmitted power is consumed for transmission of clock signal. In order to avoid any wastage of transmitted power, firstly a non-coherent detector is used to extract the clock signal in each clocked interval, which is followed by processing the demodulated baseband signals.

Symbol Synchronization

A symbol synchronizer is used in order to ensure that even in the presence of noise, the timing offset is acceptably small. Early-late gate synchronizer is basically a symbol synchronizer which uses PLL loop in an intelligent way. It contains a pair of gated integrators, called early and late gates. Each gate performs its integration over a time interval equal to half of the symbol duration. Gate intervals adjoin each other, but do not overlap.

- If the estimated and incoming data transitions coincide with each other, then the outputs of two integrators will be equal. This will result in zero error voltage. The error voltage is also zero in case the data transition is missing.
- If the estimated and incoming data transitions does not coincide with each other, then a timing and data transition occurs within the operation interval of one of the gates.

The comparative magnitude of the two integrators gives the error voltage which is used to control the VCO frequency after passing through low-pass filter. When a pulse signal $0 \leq t \leq T_s$ is passed through a matched filter, the output of the filter is maximum at $t = T_s$. If the filter output is sampled early, that is at $t = T - \delta$, or later at $t = T + \delta$ (δ being the time offset), the sampled values are not maximum. In the presence of additive white Gaussian noise (AWGN), it implies that the average sampled value may result in wrong symbol decision. Thus, a symbol synchronization scheme attempts to ensure that even in presence of noise, the time offset is acceptably small.[18]

Note... *Coherent and non-coherent detection methods for various digital modulation techniques have been described in the previous sections.*

Self-Assessment Exercise linked to LO 3.3

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/121

Q3.3.1 Distinguish between coherent and non-coherent detection of digital modulated signals at the receiver.

Q3.3.2 Interpret the unique feature of synchronous detection of digital modulated bandpass signals.

Q3.3.3 List two prominent methods of recovery of carrier signals at digital communication receiver.

Q3.3.4 Illustrate the Costas-loop carrier signal recovery method as employed in digital demodulator.

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 3.3: *Understand coherent and non-coherent detection including carrier recovery methods.*

[18] Early-late gate synchronizer is specifically necessary for demodulating narrowband carrier modulated signals. Because of its simplicity in implementation and less sensitivity towards dc offset early-gate gate is preferred even for on-board Telecommunication system. Early-late gate synchronizer can be used for any data rate depending on the sampling speed supported by the device.

LO 3.4

3.4 PERFORMANCE OF DIGITAL MODULATION TECHNIQUES

What We Discuss Here ....

A digital communication system should utilize minimum possible spectrum bandwidth for required transmission data rate while maintaining the specified limits of the probability of error.

Firstly, we describe *bandwidth efficiency*—one of the major performance-determining criteria of binary as well as *M*-ary digitally modulated signals.

Then, we discuss *error probabilities*—bit error rate and symbol rate as applied to digital modulation techniques with specific example of *M*-ary PSK.

Finally, we present a *comparative study of performance* determining parameters of various digital modulation techniques in terms of bandwidth efficiency, power versus bandwidth requirement, and error probabilities.

Define

Bandwidth efficiency (also known as *spectral efficiency*) is defined as the transmission bit rate per unit bandwidth.

By definition, bandwidth efficiency, $B_\eta = \dfrac{R_b}{B}$

where R_b is the bit rate in bits per second and B is the transmission bandwidth utilized in Hz.

Bandwidth Efficiency—Units

Bandwidth efficiency is measured in *bps/Hz*. It is generally normalized to a 1 Hz bandwidth. It indicates the number of bits that can be transmitted for each Hz of channel bandwidth.

IMPORTANT!

A high rate of transmission requires a large bandwidth. It is recommended that a digital communication system should utilize minimum possible spectrum bandwidth for required transmission data rate while maintaining the specified limits of the probability of error. The parameter "bandwidth efficiency" is quite often used to compare the performance of different digital modulation techniques.

Bandwidth Efficiency of M-ary FSK Signal

Consider an *M*-ary FSK signal that consists of an orthogonal set of *M* frequency-shifted signals. When the orthogonal signals are detected coherently, the adjacent signals need only be separated from each other by a frequency difference $1/(2T_s)$ Hz, where T_s is the symbol duration. Hence, the transmission bandwidth required to transmit *M*-ary FSK signal is given as

$$B_{\text{MFSK}} \geq M \times \frac{1}{2T_s}$$

The symbol duration T_s of *M*-ary FSK signal is given as

$$T_s = T_b \log_2 M$$

where T_b is the bit duration. Therefore, the transmission bandwidth required to transmit *M*-ary FSK signals is given as

$$B_{\text{MFSK}} = M \times \frac{1}{2T_b \log_2 M}$$

Using $R_b = 1/T_b$, we get

$$B_{\text{MFSK}} = M \times \frac{R_b}{2 \log_2 M}$$

Hence, the bandwidth efficiency, which is the ratio of the bit rate (R_b) to the transmission bandwidth, of M-ary FSK signals, is given by

$$B_{\eta\text{-MFSK}} = \frac{R_b}{B_{MFSK}} = \frac{R_b}{(R_b M)/(2\log_2 M)}$$

$$\boxed{B_{\eta\text{-MFSK}} = \frac{2\log_2 M}{M}}$$

Interpretation

This expression clearly indicates that as M increases, the bandwidth efficiency of M-ary FSK decreases. For example, it is equal to 1 for $M = 4$, and it is only 0.5 for $M = 16$.

Bandwidth Efficiency of *M*-ary PSK Signal

We know that the transmission bandwidth corresponding to the main lobe of PSD of M-ary PSK signal is given by ($2/T_s$), where T_s is the symbol duration, which is related to the bit duration T_b by the relationship $T_s = T_b \log_2 M$. Therefore, the transmission bandwidth required to transmit M-ary PSK signals is given as

$$B_{M\text{-ary PSK}} = \frac{2}{T_b \log_2 M}$$

Using $R_b = 1/T_b$, the transmission bandwidth of M-ary PSK signals is redefined as

$$B_{M\text{-ary PSK}} = \frac{2R_b}{\log_2 M}$$

Hence, the bandwidth efficiency of M-ary PSK signals is given as

$$B_{\eta(M\text{-ary PSK})} = \frac{R_b}{B_{M\text{-ary PSK}}} = \frac{R_b}{(2R_b)/(\log_2 M)}$$

$$\boxed{B_{\eta(\text{M-ary PSK})} = \frac{\log_2 M}{2}}$$

It may be noted that by considering 3 dB bandwidth of M-ary PSK signal as $1/T_s$, its bandwidth efficiency is given as $\log_2 M$.

Interpretation

In either case, this expression clearly indicates that as M increases, the bandwidth efficiency of M-ary PSK system improves. For example, it is equal to 1 for $M = 4$, and it increases to 2 for $M = 16$.[19]

Bandwidth Efficiency of *M*-ary QAM Signal

We know that the binary ASK has a bandwidth efficiency of 1, and that of quadrature-carrier ASK is twice that of binary ASK, i.e., 2.

We have discussed earlier that M-ary QAM comprises of two L-ary ASK systems on quadrature carriers, where $L = \sqrt{M}$. Since an L-ary ASK will have a bandwidth efficiency that is $\log_2 L$ times is given by

$$B_{M\text{-ary QAM}} = 2\log_2 L = 2\log_2 \sqrt{M}$$

Hence, $\boxed{B_{M\text{-ary QAM}} = \log_2 M}$

[19] In case of M-ary PSK, as the value of M increases, the distance between message signal points on the circumference of the circle on its constellation diagram decreases and so the probability of error increases. This can be compensated by increasing the radius of the circle, i.e., by increasing the transmitted power.

SOLVED EXAMPLE 3.4.1 **Bandwidth Efficiency of *M*-ary FSK Signals**

Compute and tabulate the results for bandwidth efficiency of *M*-ary FSK signals for $M = 2, 4, 8, 16, 32$, and 64.

Solution

We know that $B_{\eta\text{-MFSK}} = \dfrac{2\log_2 M}{M}$ bps/Hz

For given $M = 2$; $B_{\eta\text{-MFSK}} = \dfrac{2\log_2 2}{2} = 1.0$ bps/Hz

For given $M = 4$; $B_{\eta\text{-MFSK}} = \dfrac{2\log_2 4}{4} = \dfrac{2\log_2 2^2}{4} = 1.0$ bps/Hz

For given $M = 8$; $B_{\eta\text{-MFSK}} = \dfrac{2\log_2 8}{8} = \dfrac{2\log_2 2^3}{8} = 0.75$ bps/Hz

For given $M = 16$; $B_{\eta\text{-MFSK}} = \dfrac{2\log_2 16}{16} = \dfrac{2\log_2 2^4}{16} = 0.5$ bps/Hz

For given $M = 32$; $B_{\eta\text{-MFSK}} = \dfrac{2\log_2 32}{32} = \dfrac{2\log_2 2^5}{32} = 0.3125$ bps/Hz

For given $M = 64$; $B_{\eta\text{-MFSK}} = \dfrac{2\log_2 64}{64} = \dfrac{2\log_2 2^6}{32} = 0.1875$ bps/Hz

From the above results, it is observed that the bandwidth efficiency of *M*-ary FSK tends to decrease with increasing values of *M*.

Table 3.4.1 gives the bandwidth efficiency for given values of *M* for *M*-ary FSK signal.

Table 3.4.1 Bandwidth Efficiency for *M*-ary FSK Signal

M	2	4	8	16	32	64
$B_{\eta-\text{MFSK}}$ (bps/Hz)	1.0	1.0	0.75	0.5	0.3125	0.1875

SOLVED EXAMPLE 3.4.2 **Bandwidth Efficiency of *M*-ary PSK Signals**

Compute and tabulate the results for bandwidth efficiency of *M*-ary PSK signals for $M = 2, 4, 8, 16, 32$, and 64.

Solution

We know that $B_{\eta\,(M\text{-ary PSK})} = \dfrac{\log_2 M}{2}$

For given $M = 2$; $B_{\eta(\text{BPSK})} = \dfrac{\log_2 2}{2} = \dfrac{1}{2} = 0.5$ bps/Hz

For given $M = 4$; $B_{\eta(\text{QPSK})} = \dfrac{\log_2 4}{2} = \dfrac{\log_2 2^2}{2} = 1.0$ bps/Hz

For given $M = 8$; $B_{\eta(\text{8PSK})} = \dfrac{\log_2 8}{2} = \dfrac{\log_2 2^3}{2} = 1.5$ bps/Hz

For given $M = 16$; $B_{\eta(16\text{PSK})} = \dfrac{\log_2 16}{2} = \dfrac{\log_2 2^4}{2} = 2.0$ bps/Hz

For given $M = 32$; $B_{\eta(32\text{PSK})} = \dfrac{\log_2 32}{2} = \dfrac{\log_2 2^5}{2} = 2.5$ bps/Hz

For given $M = 64$; $B_{\eta(64\text{PSK})} = \dfrac{\log_2 64}{2} = \dfrac{\log_2 2^6}{2} = 3.0$ bps/Hz

From the above results, it is observed that the bandwidth efficiency of M-ary PSK tends to increase with increasing values of M.

Table 3.4.2 gives the bandwidth efficiency for different values of M for M-ary PSK signal.

Table 3.4.2 Bandwidth Efficiency for M-ary PSK Signal

M	2	4	8	16	32	64
$B_{\eta(\text{M-ary PSK})}$ (bps/Hz)	0.5	1.0	1.5	2.0	2.5	3.0

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 3.4.3 Show that M-ary PSK signals are spectrally more efficient than M-ary FSK signals for value of M greater than 4.

Ex 3.4.4 Compute and tabulate the results for bandwidth efficiency of M-ary QAM signals for M = 2, 4, 8, 16, 32, and 64.

SOLVED EXAMPLE 3.4.5 Bandwidth Efficiency Comparison

Compare and contrast the values of bandwidth efficiency (in terms of bps/Hz) for M-ary PSK systems with that of M-ary FSK systems for different values of M.

Solution

We know that $B_{\eta\ (M\text{-ary PSK})} = \dfrac{\log_2 M}{2}$, and $B_{\eta\ (M\text{-ary FSK})} = \dfrac{2\log_2 M}{M}$

Table 3.4.3 gives the bandwidth efficiency for M-ary PSK and M-ary FSK systems for different values of M including BPSK and BFSK systems.

Table 3.4.3 *Bandwidth Efficiency for M-ary PSK and M-ary FSK Systems*

S. No.	Value of M	$B_{\eta\,(\text{M-ary FSK})} = \dfrac{\log_2 M}{2}$	$B_{\eta\,(\text{M-ary FSK})} = \dfrac{2\log_2 M}{M}$
1.	2 (Binary)	0.5 bps/Hz	1.0 bps/Hz
2.	4-ary	1.0 bps/Hz	1.0 bps/Hz
3.	8-ary	1.5 bps/Hz	0.75 bps/Hz
4.	16-ary	2.0 bps/Hz	0.5 bps/Hz
5.	32-ary	2.5 bps/Hz	0.3125 bps/Hz
6.	64-ary	3.0 bps/Hz	0.1875 bps/Hz

Hence, it is concluded that as the value of M is increased, bandwidth efficiency of M-ary PSK systems increases whereas that of M-ary FSK systems decreases. Thus, M-ary PSK systems are spectrally more efficient for $M \geq 8$.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 3.4.6 Compare the minimum bandwidth requirement and bandwidth efficiency for ASK, FSK, M-ary PSK and M-ary QAM digital modulation techniques in a tabular form.

Error Probabilities—Significance

The binary and M-ary digital modulation techniques have an unequal number of symbols. These two systems can be compared if they use the same amount of energy to transmit each bit of information, E_b. Since it is important that it is the total amount of energy needed to transmit the complete information satisfactorily, not the amount of energy needed to transmit a particular symbol. Ideally, FSK and PSK signals have a constant envelope which makes them impervious to amplitude non-linearities, and are much more widely used than ASK signals. So a comparison between binary and M-ary FSK and PSK signals can be made.

Concept of Probability of Error

One of the main criteria for comparing the performance parameters of digital modulation techniques is the average *probability of error*, P_e expressed as a function of bit-energy-to-noise density ratio, E_b/N_o. The statistical characteristics of the correlation receiver requires the evaluation of its *noise performance* in terms of probability of error, P_e.

Two Methods of Encoding

We know that if binary and M-ary digital modulation systems are to be able to transmit information at the same data rate, then it is mandatory that $T_s = n \times T_b$. Accordingly, there are two different methods for binary and M-ary modulation systems—the first one is described as *bit-by-bit encoding*, and the second one is *symbol-by-symbol encoding*. It is pertinent to determine which method provides the higher probability that the complete information is correctly received.

Bit Error Rate—Its Definition and Significance

Bit Error Rate (BER) is defined as the ratio of total number of bits received in error and total number of bits received over a large session of information transmission. It is assumed that total number of bits received is same as total number of bits transmitted from the source. It is an average figure.

The system parameter BER signifies the quality of digital information delivered to the receiving end user. It also indicates the quality of service provided by a digital communication system.

Symbol Error Rate—Its Definition and Significance

Symbol Error Rate (SER) is defined as the ratio of total number of symbols detected by demodulator in error and total number of symbols received by the receiver over a large session of information transmission. It is assumed that total number of symbols received is same as total number of symbols transmitted by the modulator.

It is also an average figure. It is used to describe the performance of a digital communication system.

Bit Error Probability

If the information data is transmitted through bit-by-bit encoding, the probability that the information is correctly received is the probability that all of the n bits of an information data are correctly received. If *bit-error probability* is the probability that bit is received in error, then the probability that the complete information data is correctly received is $P_i = (1 - P_{eb})^n$; where P_{eb} is the probability of bit error or bit error rate (BER).

So it is clear that in bit-by-bit encoding, for fixed P_{eb}, the probability of correctly receiving the information decreases with increasing n, and therefore the probability that there is some error in the detected information increases with n. For example, in case of binary PSK,

Interpretation

$$\boxed{P_{eb}(\text{BPSK}) = \frac{1}{2} erfc\sqrt{\frac{E_b}{N_0}}}$$

$$P_{c\text{-information}}(\text{BPSK}) = \left(1 - \frac{1}{2} erfc\sqrt{\frac{E_b}{N_0}}\right)^m$$

If a symbol consists of a single bit, as in BPSK, then the bit error probability, P_{eb} is same as the symbol error probability, P_{es}. Generally, a digital system encodes n number of bits into M symbols with the relationship $M = 2^n$. When an n-bit symbol is received with error, it may be that 1 bit or 2 bits or all n bits are received in error. Assuming that the probability, P_{es} of receiving any of these symbols in errors is the same, then it is possible to relate P_{es} with P_{eb}.

IMPORTANT!

As an example, consider M-ary PSK in which $M = 4$; $n = 2$, that means QPSK signal. With such a two-bit transmitted symbol, there are three possible received signals after detection which could be in error such as

Probability of bit-Error for *M*-ary PSK

- One such received signal may have an error in the first bit
- A second received signal may have an error in the second bit
- The third received signal may have errors in both bits

The probability of a bit-error P_{eb} is then the weighted average, that is,

$$P_{eb}(\text{QPSK}) = \frac{\frac{1}{2}P(1^{\text{st}}\,\text{bit-error}) + \frac{1}{2}P(2^{\text{nd}}\,\text{bit-error}) + \frac{2}{2}P(\text{two\_bit-errors})}{3}$$

Since the three bit-symbol probabilities are the same, therefore,

$$P_{eb}(\text{QPSK}) = \frac{4/2}{3} P_{es}$$

Or, $$P_{eb}(\text{QPSK}) = \frac{4/2}{(4-1)} P_{es}$$

As in QPSK, $M = 4$; the relationship between bit-error probability and symbol-error probability can be expressed in the general form as

$$\boxed{P_{eb}(M\text{-ary PSK}) = \frac{M/2}{(M-1)} P_{es}}$$

This expression applies to M-ary PSK (or even *M-ary FSK*) in which the M orthogonal symbols are equiprobable.

Actually, errors in received symbols which involve many bit errors and which contribute heavily to bit-error probability are less likely than received symbols with fewer bit errors. Hence, bit-error probability is an overestimate, that is, an upper bound. In case of an 8-ary PSK in which symbols are encoded in such a way that closest neighboring symbols differ by only a single bit. When one of M ($M = 8$) possible symbols is received and is erroneously decoded, it will be misinterpreted as one or another of only a limited number of the other ($M - 1$) symbols. This gives a lower bound on bit-error probability P_{eb} as

Interpretation

$$P_{eb}(M\text{-ary PSK}) = \frac{P_{es}}{\log_2 M}\text{ ; for M} \geq 2$$

$$P_{eb}(M\text{-ary PSK}) = \frac{P_{es}}{\log_2 2^n} = \frac{P_{es}}{n}$$

Assuming that every signal has an equal likelihood of transmission, the bit-error probability is bounded as follows:

$$\frac{P_{es}}{n} \leq P_{eb}(M\text{-ary PSK}) \leq \frac{M/2}{M-1} P_{es}$$

Note that for large M, the bit-error probability approaches the limiting value $P_{es}/2$.

Table 3.4.4 Summary of P_{eb} for Coherent Digital Modulation Systems

S. No.	Digital Modulation Technique	Bit-error Probability, P_{eb}
1.	ASK	$\frac{1}{2} erfc\sqrt{\frac{E_b}{4N_0}}$
2.	Coherent BFSK	$\frac{1}{2} erfc\sqrt{\frac{E_b}{2N_0}}$
3.	Non-coherent BFSK	$\frac{1}{2}\exp\left(-\frac{E_b}{2N_0}\right)$
4.	M-ary FSK	$\leq \frac{M-1}{2} erfc\sqrt{\frac{E_s}{2N_0}}$
5.	Coherent BPSK	$\frac{1}{2} erfc\sqrt{\frac{E_b}{N_0}}$
6.	Coherent DBPSK	$erfc\sqrt{\frac{E_b}{N_0}} - \frac{1}{2} erfc^2\sqrt{\frac{E_b}{N_0}}$
7.	Non-coherent DBPSK	$\frac{1}{2}\exp\left(-\frac{E_b}{N_0}\right)$
8.	Coherent QPSK	$erfc\sqrt{\frac{E_b}{N_0}} - \frac{1}{4} erfc^2\sqrt{\frac{E_b}{N_0}}$
9.	M-ary PSK (for large E_b/N_0 and $M \geq 4$	$erfc\left(\sqrt{\frac{E_s}{N_0}}\sin\left(\frac{\pi}{M}\right)\right)$
10.	Coherent MSK	$erfc\sqrt{\frac{E_b}{N_0}} - \frac{1}{4} erfc^2\sqrt{\frac{E_b}{N_0}}$

Figure 3.4.1 depicts BER versus curves $\frac{E_b}{N_0}$ (dB) for non-coherent BFSK, coherent BFSK, coherent BPSK, DPSK, coherent QPSK, and coherent MSK modulation techniques for comparative analysis.

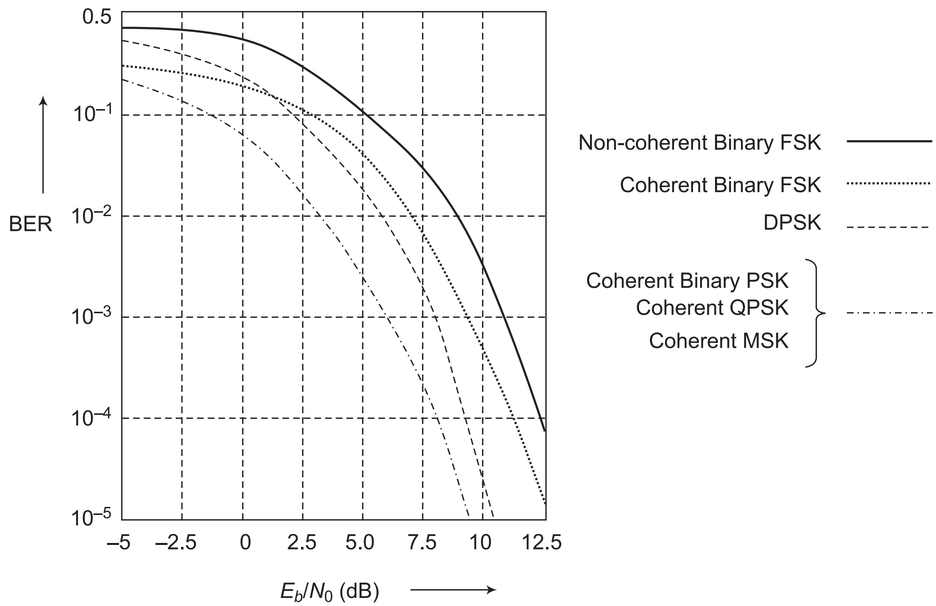


Figure 3.4.1 BER versus E_b/N_0 curves for BFSK, M-ary PSK, and MSK

Interpretation

- The probability of error decreases monotonically with increasing values of E_b/N_0 for all digital modulation systems.
- For any value of E_b/N_0, coherent BPSK modulation produces a low probability of error than any of the other systems.
- Coherent BPSK modulation requires an E_b/N_0 value which is 3 dB less than the corresponding values for coherent BFSK.
- Increasing the value of M in orthogonal M-ary FSK systems has the opposite effect to that in non-orthogonal M-ary PSK systems.
- For same E_b/N_0 value in M-ary PSK system, P_{eb} increases as M increases whereas in M-ary FSK, E_b/N_0 value decreases as M increases.
- In M-ary FSK or M-ary PSK systems, as M increases, the system approaches the Shannon limit of $E_b/N_0 = -1.6$ dB.
- Increasing M is equivalent to increased bandwidth requirement.
- For same P_{eb}, less E_b/N_0 is required in case of coherent FSK as compared to non-coherent FSK.

Application Criteria

Each of the digital modulation techniques is suited to specific applications. In general, techniques that rely on more than two levels (e.g., QPSK, QAM) require better signal-to-noise ratios (SNR) than two-level techniques (e.g., BPSK) for similar BER performance. Additionally, in a wireless environment, multi-level techniques generally require greater power amplifier linearity than two-level techniques. The fact that GMSK uses a two-level continuous phase modulation (CPM) format has contributed to its popularity. Another point in its favor is that it allows the use of class C power amplifiers (relatively non-linear) and data rates approaching the channel bandwidth (dependent on filter bandwidth and channel spacing).

SOLVED EXAMPLE 3.4.7 Error Performance of 8-PSK System

For an 8-PSK system operating at 20 Mbps with a carrier-to-noise power ratio of 11 dB, determine the minimum bandwidth required to achieve a probability of error of 10^{-6}. The corresponding minimum E_b/N_o ratio for an 8-PSK system is 14 dB.

Solution The energy per bit-to-noise power density ratio, E_b/N_0 is simply the ratio of the energy of a single bit to the noise power present in 1 Hz of bandwidth.

It is used to compare two or more digital modulation systems that use different modulation schemes (FSK, PSK, QAM), or encoding techniques (M-ary) operating at different transmission rates (bit rates).

Thus, it normalizes all multiphase modulation techniques to a common noise bandwidth, thereby resulting into a simpler and accurate comparison of their error performance.

Mathematically, $\frac{E_b}{N_o} = \frac{C/f_b}{N/B} = \frac{C}{N} \times \frac{B}{f_b}$; where C/N is carrier-to-noise power ratio, and B/f_b is noise bandwidth-to-bit rate ratio.

Expressing this expression in dB,

$$\frac{E_b}{N_o}(\text{dB}) = \frac{C}{N}(\text{dB}) + \frac{B}{f_b}(\text{dB})$$

Or,
$$\frac{B}{f_b}(\text{dB}) = \frac{E_b}{N_o}(\text{dB}) - \frac{C}{N}(\text{dB})$$

For given C/N = 11 dB and E_b/N_o= 14 dB, we have

$$\frac{B}{f_b}(\text{dB}) = 14 - 11 = 3 \text{ dB} \Rightarrow \frac{B}{f_b} = \text{antilog}\left(\frac{3}{10}\right) = 2$$

For given f_b = 20 Mbps, B = 2 × 20 Mbps = 40 MHz **Ans.**

SOLVED EXAMPLE 3.4.8 Power and Bandwidth Requirements for *M*-ary PSK

Tabulate the relative values of power and bandwidth requirements for M-ary PSK signals with respect to BPSK signal for symbol-error probability of 10^{-4}.

Solution Table 3.4.5 depicts the relative values of power and bandwidth requirements for coherent M-ary PSK signals for with respect to BPSK signal symbol-error probability of 10^{-4} under identical noise environment.

Table 3.4.5 Power and Bandwidth Requirement of *M*-ary PSK

S. No.	Value of M	$\frac{(\text{Average Power})_{M\text{-ary PSK}}}{(\text{Average Power})_{\text{BPSK}}}$	$\frac{B_{M\text{-ary PSK}}}{B_{\text{BPSK}}}$
1.	4	0.34 dB	½ or 0.5
2.	8	3.91 dB	1/3 or 0.333
3.	16	8.52 dB	1/4 or 0.25
4.	32	13.52 dB	1/5 or 0.2

Clearly, for $M > 8$, the power requirements for *M*-ary PSK systems become excessive.

Table 3.4.6 shows the comparison of spectral efficiency and the required SNR (for BER of 1 in 10^6) for PSK and MSK digital modulation techniques.[20]

Table 3.4.6 Performance Comparison of PSK and MSK

Digital Modulation Technique	Spectral Efficiency	Required SNR
BPSK	1 bps/Hz	11.1 dB
QPSK	2 bps/Hz	14.0 dB
16-PSK	4 bps/Hz	26.0 dB
2-MSK	1 bps/Hz	10.6 dB
4-MSK	2 bps/Hz	13.8 dB

Table 3.4.7 summarizes various digital modulation techniques adopted in second-generation cellular and cordless telephone systems.

Table 3.4.7 Spectral Efficiency of QPSK and GMSK

Digital Modulation Technique	Channel Bandwidth, kHz	Data Rate, kbps	Spectral Efficiency, bps/Hz	Application
$\pi/4$-QPSK	30	48.6	1.62	US Digital Cellular
$\pi/4$-QPSK	25	42.0	1.68	Japanese Digital Cellular
GMSK ($BT_b = 0.5$)	1728	1572.0	0.67	Digital Enhanced Cordless Telephony
GMSK	100	72.0	0.72	Cordless Telephony (CT-2)
GMSK ($BT_b = 0.3$)	200	270.8	1.35	GSM

[20] Spectral efficiency influences the spectrum occupancy in a mobile radio system. Theoretically, an increase in the number of modulation levels results into higher spectral efficiency. But higher SNR is required to achieve same BER performance. $\pi/4$-QPSK is the most bandwidth efficient modulation, having moderate hardware complexity. GMSK modulation offers constant envelope, narrow power spectra, and good error rate performance.

Table 3.4.8 shows typical application areas of various bandwidth-efficient digital modulation techniques.

Table 3.4.8 Applications of Digital Modulation Techniques[21]

S. No.	Digital Modulation Technique	Typical Application Areas
1.	Frequency Shift Keying (FSK)	Paging Services, Cordless Telephony
2.	Binary Phase Shift Keying (BPSK)	Telemetry
3.	Quaternary Phase Shift Keying (QPSK)	Cellular Telephony, Satellite Communications, Digital Video Broadcasting
4.	Octal Phase Shift Keying (8-PSK)	Satellite Communications
5.	16- or 32-level Quadrature Amplitude Modulation (16-QAM or 32-QAM)	Microwave Digital Radio Links, Digital Video Broadcasting
6.	64-level Quadrature Amplitude Modulation (64-QAM)	Digital Video Broadcasting, Set-Top Boxes, MMDS
7.	Minimum Shift Keying (MSK)	Cellular Telephony

Self-Assessment Exercise linked to LO 3.4

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/126

Q3.4.1 Define the terms: bandwidth efficiency, bit error rate, symbol error rate, error probability.

Q3.4.2 Compare and contrast the minimum bandwidth requirement in terms of input bit rate f_b, and bandwidth efficiency in bps/Hz for binary PSK, QPSK, 8-PSK, 16-PSK, 32-PSK, 4-QAM, 8-QAM, 16-QAM, 32-QAM and 64-QAM digital modulation techniques.

Q3.4.3 Derive an expression for probability of bit error for M-ary PSK signal. Show that for large value of M, the bit-error probability approaches the limiting value.

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 3.4: *Develop performance determining criteria in terms of bandwidth efficiency and error probabilities of various digital modulation techniques.*

[21] Orthogonal Frequency Division Multiplexing (OFDM) modulation is based on the idea of a Frequency Division Multiplexing (FDM), but is utilized as a digital modulation technique. The bit stream is split into several parallel data streams, each transferred over its own sub-carrier using some conventional digital modulation technique. The sub-carriers are summed into an OFDM symbol. OFDM is considered as a modulation technique rather than a multiplexing technique, since it transfers one bit stream over one communication channel using one sequence of so-called OFDM symbols. OFDM can be extended to multi-user multiple access technique, allowing several users to share the same physical medium by giving different sub-carriers to different users. It is discussed in Chapter 6.

Key Concepts

- amplitude-shift keying (ASK)
- coherent detection
- constellation diagram
- differential binary phase-shift keying (DBPSK)
- dibit
- digital modulation
- frequency-shift keying (FSK)
- Gaussian minimum-shift keying (GMSK)
- modulation
- non-coherent detection
- phase-shift keying (PSK)
- quad bit
- quadrature phase-shift keying (QPSK)
- quadrature amplitude-shift keying (QAM)
- symbol
- synchronous detection
- tribit

Learning Outcomes

LO 3.1

- Digital transmission uses amplitude, frequency, and phase variations of the analog carrier signal, just as in analog transmission.
- FSK uses two transmitted frequencies to achieve modest data rates with reasonably good error performance.

LO 3.2

- Most PSK systems such as QPSK, DQPSK, OQPSK, π/4-QPSK use four phase angles for slightly higher data rates than are achievable with FSK.
- Gaussian minimum-shift keying (GMSK) is a special case of FSK that achieves the minimum bandwidth possible for a binary FSK system at a given data rate.
- GMSK is considered the most promising digital modulation technique although linear digital modulation techniques offer better spectral efficiency.
- QAM achieves higher data rates than FSK or PSK by using a combination of amplitude and phase digital modulation.

LO 3.3

- When the detection process does not require locally generated receiver carrier signal to be phase locked with transmitter carrier signal, it is known as non-coherent detection of digital modulated signal.
- When the detection is carried out by correlating received noisy signals and locally generated carrier signal, it is known as coherent detection of digital modulated signal.
- In coherent detection, the local carrier signal generated at the receiver is phase-locked with the carrier signal at the transmitter.

LO 3.4

- The maximum data rate for a communications channel is a function of digital modulation technique, bandwidth, and signal-to-noise ratio.
- Generally, more complex digital modulation technique can achieve higher data rates, but only when the *S*/*N* is high.

Hands-on Projects

3.1 The IC MAX1472 is a crystal-referenced phase-locked loop (PLL) VHF/UHF transmitter designed to transmit ASK/OOK data in the 300 MHz to 450 MHz frequency range. It supports data rates up to 100 kbps, and adjustable output power to more than +10 dBm into a 50 Ω load. Design, fabricate and test a typical application using it.

3.2 Design and fabricate wireless transmitter and receiver operating at 433 MHz using ASK RF modules alongwith IC HT12E encoder and IC HT12D decoder.

3.3 Design a circuit to realize binary frequency shift-keying (BFSK) demodulator using PLL with standard IC 565. Apply its output signal to 3-stage *RC* low-pass filter and then use a comparator to obtain binary data output waveform.

3.4 Design a complete FSK communication system comprising of square-wave generator, FSK modulator and demodulator, comparator and display unit, using various linear ICs such as Timer IC555, PLL IC565, Comparator IC 741/351 and digital ICs as display drivers and LCD display.

3.5 Digital satellite broadcast reception set top box uses QPSK demodulation error correction IC CXD19611Q alongwith IC CXA3108Q channel selection and IC CXA3038N quadrature demodulation. Study its functional block schematic and design the complete circuit.

3.6 Baseband MSK is a robust means of transmitting data in wireless systems where the data rate is relatively low compared to the channel bandwidth. Design and implement MSK modulation and demodulation circuits using MX-COM's devices such as MX429 and MX469 as single chip solution. As an alternative method, you may realize MSK modulation circuit by directly injecting NRZ data into a frequency modulator with its modulation index set for 0.5.

Objective-Type Questions

For Interactive Quiz with answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/83

3.1 Identify the type of digital modulation technique that allows more bits per symbol and, therefore, greater speed in a given bandwidth than other digital modulation techniques.
(a) OQPSK
(b) π/4-QPSK
(c) 16-PSK
(d) QAM

3.2 Which parameter of an analog sine-wave carrier signal can be modulated by digital data information, leading to the most popular type of digital modulation technique?
(a) Amplitude
(b) Frequency
(c) Phase
(d) Amplitude and phase

3.3 How many bits per symbol are needed to operate a modem operating at 9600 bps over the telephone system?
(a) 2 bits per symbol
(b) 4 bits per symbol
(c) 8 bits per symbol
(d) 16 bits per symbol

3.4 Check the different symbols that are possible at the output of a 16-QAM modulator.
(a) 8
(b) 16
(c) 64
(d) 256

3.5 *Statement I:* Conventional QPSK has transitions through zero, that is, 180° phase transitions.
Statement II: In offset QPSK, the transitions on the *I* and *Q* channels are staggered and the phase transitions are therefore limited to 45°.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

3.6 Exemplify that the __________ type of digital modulation technique gives minimum probability of error.
(a) ASK
(b) FSK
(c) PSK
(d) Differential binary PSK

3.7 Typical value of (BT_b) product in MSK digital modulation is
(a) 0.3
(b) 0.5
(c) 1.0
(d) ∞

3.8 *Statement I:* The ratio of bit rate to channel bandwidth, expressed in bits per second per Hz, is used as a figure of merit in digital radio systems.
Statement II: As the value of *M* increases, the phase difference corresponding to various symbol decreases and therefore it is difficult for *M*-ary PSK demodulator to distinguish between symbols received, especially in the presence of noise and interference.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

3.9 In a dibit system, the symbol rate or baud rate is _________ the bit rate.
(a) half
(b) same as
(c) double
(d) four times

3.10 One of the following type of digital modulation technique is used for high-speed telephone modems.
(a) QPSK
(b) 8-PSK
(c) QAM
(d) GMSK

Short-Answer-Type Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/166

3.1 What could be the limiting factor for FSK modems for use over standard analog telephone lines? Is FSK generally used for high-performance digital radio systems? ○○●

3.2 Bring out clearly as how can OQPSK digital modulation be derived from conventional QPSK digital modulation scheme. ●●●

3.3 Discriminate differential binary PSK digital modulation from that of binary PSK. ○●●

3.4 Give sufficient reasons so as to justify that GMSK modulation offers distinct advantage over MSK. ○●●

3.5 Identify at least four essential features of GMSK digital modulation technique. Mention cell phone system in which GMSK is employed. ○○●

3.6 Discuss the main benefit of varying amplitude along with phase in M-ary PSK digital modulation system. ○○●

3.7 List the advantages of constant envelope family of digital modulation techniques. ○○●

3.8 Define bit error rate and symbol error rate in digital modulation. ○○●

3.9 State the reason as why the error performance of M-ary PSK demodulator is poor. ○○●

3.10 Give examples of non-linear (constant envelope), linear, and combined non-linear plus linear digital modulation techniques. ●●●

3.11 What do you mean by figure of merit in digital radio systems? Discuss them by considering the example of QAM digital modulation. ○●●

Discussion Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/124

3.1 We wish to transmit binary data over bandpass channels having limited bandwidth. It is mandatory to have digital communication link which is immune to noise and interference. Suggest suitable digital modulation technique. Discuss the functional block schematics of its modulator and coherent detector. **[LO 3.1]**

3.2 In $\pi/4$-QPSK, there will always be a phase change for each input symbol. This enables the receiver to perform synchronization and timing recovery at the detector. Describe Offset QPSK modulator and bring out the changes needed for $\pi/4$-QPSK modulator. **[LO 3.2]**

3.3 Compare and contrast multilevel (M-ary) digital modulation techniques such as 16-QAM, 16-PSK, and 16-APSK in terms of their respective geometrical representation (constellation diagrams). **[LO 3.2]**

3.4 Carrier signal recovery method is primarily used for coherent detection of digital modulated signal. Draw the functional block schematic diagram of M^{th} power loop carrier recovery method and describe its operation. **[LO 3.3]**

3.5 A digital modulation technique used for mobile environment should utilize the transmitted power and RF channel bandwidth as efficiently as possible because mobile radio channel is both power- and bandlimited. Paraphrase the concept of bandwidth efficiency and analyze critically the bandwidth efficiency in different binary and M-ary digital modulation techniques. **[LO 3.4]**

Problems

3.1 Represent the binary data sequence 1 0 1 1 0 0 1 0 with the resulting waveform using ASK, FSK and PSK digital modulation techniques.

3.2 For a binary FSK signal with a mark frequency of 49 kHz, and a space frequency of 51 kHz, calculate the minimum bandwidth requirement and baud rate for an input data rate of 2 kbps.

3.3 For a QPSK modulator, how many symbols are possible at its output? Check that the phase difference between each symbol is 90°.

3.4 Consider an 8-PSK modulator with an input data rate of 10 Mbps and an analog carrier frequency of 80 MHz. Find the minimum Nyquist bandwidth and the resultant baud rate.

3.5 Show that a four-level QAM system would require 3.4 dB less E_b/N_o ratio as compared to that of an 8-PSK system for a specified probability of error = 10^{-6}. The corresponding minimum E_b/N_o ratio for an 8-PSK system is 14 dB, and for a four-level QAM system is 10.6 dB.

3.6 For an 8-PSK system operating at 10 Mbps with a carrier-to-noise power ratio of 11.7 dB, determine the minimum bandwidth required to achieve a probability of error of 10^{-7}.The corresponding minimum E_b/N_o ratio for an 8-PSK system is specified as 14.7 dB.

3.7 A terrestrial microwave radio system uses 256 QAM digital modulation scheme. Estimate the number of bits per symbol used.

3.8 For a 16-PSK system with a transmission bandwidth of 10 kHz, design the system for maximum bit rate.

3.9 We know that bandwidth efficiency is given as the transmission bit rate per unit bandwidth occupied. Determine the bandwidth efficiency for QPSK system operating at the bit rate of 20 Mbps. Assume B = 10 MHz.

For answer keys, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/134

Critical Thinking Questions

3.1 When we use a carrier signal of exactly the same frequency (and phase) as the carrier signal used for modulation, the method of recovering the baseband digital data is known as synchronous detection, or coherent detection. The binary FSK can be viewed as two interleaved ASK signals with carrier frequencies f_{c0} and f_{c1}, respectively. Illustrate and explain the coherent binary FSK detection method. **[LO 3.1]**

3.2 In Quadrature Phase Shift Keying (QPSK) digital modulation, we combine two successive bits in a bit stream to form a symbol. With two bits, there are four possible conditions: 11, 10, 01, and 00.Write the expression for QPSK signal and draw the constellation diagram depicting the relationship between symbol, binary input and phase shift. **[LO 3.2]**

3.3 Gaussian minimum shift keying (GMSK) is a special case of MSK in which a pre-modulation low-pass Gaussian filter is used as a pulse-shaping filter to reduce the bandwidth of the baseband signal before it is applied to MSK modulator. Highlight the performance improvements achieved with the use of a filter with Gaussian characteristics with the MSK approach. Show the filter characteristics taking ($B \times T_b$) = 0.3, 0.5 and infinity. **[LO 3.2]**

3.4 In an ideal form of coherent detection, exact replica of the received digital modulated signals is available at the receiver. It implies that the receiver has complete knowledge of the phase reference of the carrier signal used at the modulator. Compare and contrast two basic methods of carrier signal recovery–carrier synchronization and symbol synchronization. **[LO 3.3]**

3.5 The binary and *M*-ary digital modulation techniques can be compared in terms of average probability of error if they use the same amount of energy to transmit each bit of information. Elucidate bit error rate and symbol error rate for binary phase shift keying (BPSK) digital modulation. How does the probability of error vary with increasing values of E_b/N_o for any digital modulation system? Draw the BER versus E_b/N_o curves for coherent binary FSK and coherent binary FSK and infer the results. **[LO 3.4]**

References for Further Reading

[Cou00] Couch, L; *Digital and Analog Communication Systems. Upper Saddle River, NJ: Prentice Hall, 2000*.

[SI12] Singal, TL; *Analog and Digital Communications.* Tata McGraw-Hill, 2012.

4 Information Theory

Learning Objectives

To master information theory, you must cover these milestones:

- **LO 4.1** Understand discrete and continuous messages, message sources, amount of information and its measure
- **LO 4.2** Discuss the probabilistic behavior of a source of information
- **LO 4.3** Illustrate the properties of discrete memoryless channel and mutual information
- **LO 4.4** Analyze the intrinsic ability of the communication channel to convey information reliably

Reliable Transmission of Information – An Essence

A reliable transmission of information is the basic requirement of any electronic communication system. We have discussed digitization of analog information, followed by transmission of baseband as well as passband (modulated) digital signals in the previous chapters. It is important that the information detected at the receiver end is error-free, or if any error occurs due to channel noise, the receiver must be able to detect, locate, and correct it. We know that the performance of electronic communication system is determined by the available signal power, limited transmission bandwidth, and the presence of undesirable noise along with the received signal. So, the question that arises in our mind is how to represent the information for reliable transmission over the noisy channel.

INTRODUCTION

Historical Aspects

In 1948, *Claude Shannon* developed *A Mathematical Theory of Communication* in which he emphasized that the message should be represented by its *information* rather than by the signal. Recognizing this fact, his pioneered work was renamed *Information Theory*, which deals with mathematical modeling and analysis of a communication system rather than with physical sources and channels.

Three Basic Concepts of Information Theory

Information theory provides the fundamental limits on the performance of digital communication system by specifying the minimum number of bits per symbol to fully represent the source information or message. Information theory describes three basic concepts:
(a) the rate at which the source generates the information;
(b) the maximum rate at which reliable transmission of information is possible over a given channel with an arbitrarily small error; and
(c) a scheme for efficient utilization of the channel capacity for information transfer.

Interpretation The first concept is related to the measure of source information, the second one, to the information capacity of a channel, and the third one to coding.

Entropy and Capacity

It is true that a greater amount of information is conveyed when the receiver correctly identifies a less likely message. The average information rate is zero for both extremely likely and extremely unlikely messages. The probabilistic behavior of a source of information is known as *entropy*, and the intrinsic ability (noise characteristics) of the communication channel to convey information reliably is known as *capacity*.

Good to Know!

A remarkable result that emerges from information theory is that if the entropy of the source is less than the capacity of the channel then error-free communication over the channel can be achieved by employing a coding technique. There is a trade-off between information capacity, channel bandwidth, and signal-to-noise ratio (S/N) of communication channel which defines the maximum rate at which transmission of information can take place over the channel.

A PRIMER

Recap

In digital communication systems, the original source messages may be in the form of discrete (digitized data) or continuous (analog data). The analog data is converted to digital pulses, known as digitized analog data, prior to transmission and converted back to analog form at the receiver end. Digitizing an analog information signal often results in improved transmission quality, with a reduction in signal distortion and an improvement in signal-to-noise power ratio.

IMPORTANT!

The goals of the system designers for any digital communication system include the following parameters:

- Required transmission bandwidth—To be minimized.
- Transmission bit rate—To be maximized.
- Required signal power—To be minimized.
- Bit-energy to noise-power-spectral-density ratio—To be minimized.
- Probability of error—To be minimized.
- Reliable user services (minimum delay and maximum resistance to interference)—To be provided.
- System complexity, computation load, and system cost—To be minimized.

A Typical Digital Communication System

Figure 4.0 illustrates a typical functional block schematic of a digital communication system, depicting the flow of information from source to destination via noisy communication channel.

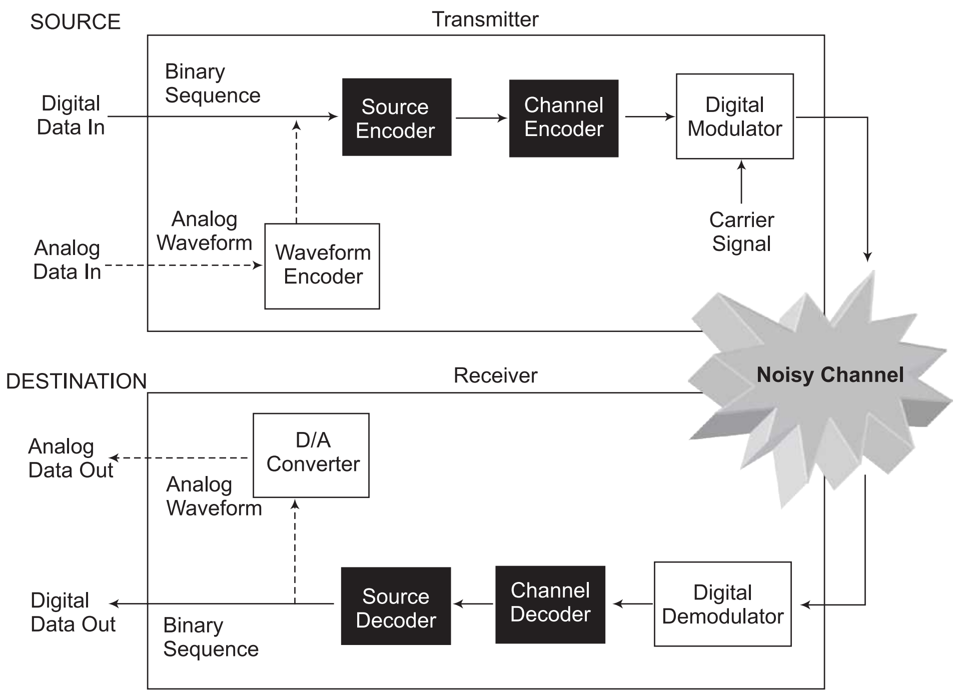


Figure 4.0 A Typical Digital Communication System

Functional Description

The analog information is sampled and digitized using a *waveform encoder* to make the final source output to be in digital form. In other words, a source produces messages in the form of signals, which may be either discrete-time, or continuous-time in nature. The digital pulse waveforms are propagated between source and destination through various levels of coding as required by different system applications.

- The *source encoder/decoder* is used to match the message source having the source information rate within the channel capacity.
- The *channel encoder/decoder* performs the function of error-control coding with an objective of providing an equivalent noiseless channel with a well-defined information capacity.

- The *digital modulator/demodulator* is required for wireless transmission of encoded original information. The original information or message may be available in an analog form or in digital form.

In this chapter...

LO 4.1
- We begin with types of messages such as discrete and continuous, different message sources like discrete memoryless sources (DMS) and Markov. It is followed by description of amount of information contained in a long message and the units of source information.

LO 4.2
- Next, we discuss the average information per individual message generated by a source, and is known as the *entropy* (probabilistic behavior), and their properties for a binary- and multilevel discrete memoryless source of information.

LO 4.3
- Then the discussion is carried forward to describe the characteristics of communication channel through which the information passes from transmitter to receiver.

LO 4.4
- Finally, we analyze the noise characteristics of the channel with an objective of conveying information reliably using Shannon's channel coding theorem. Trade-off between information capacity, channel bandwidth, and S/N of communication channel given by Shannon–Hartley theorem is also discussed.

LO 4.1

4.1 DISCRETE AND CONTINUOUS MESSAGES

Recall
From Chapters 1 and 2

We know how to convert an analog signal to a digital data (sequence of binary symbols 1s and 0s) either by using waveform encoders such as PCM, DM, or by using linear predictive encoders such as vocoders for speech signal. The digital output comprising of a long sequence of binary symbols 1s and 0s is neither uniform nor suitable for direct transmission over the channel. These binary symbols are required to be converted into electrical pulses or waveforms having finite voltage levels so as to make it compatible for transmission over the channel. It is in a *line code* that a binary stream of data takes on an electrical representation.

Analog Signals

Analog signals are those signals whose amplitude levels can be specified at any value in a continuous range.

- In *analog discrete-time signals*, the amplitude levels may be specified only at discrete points of time.

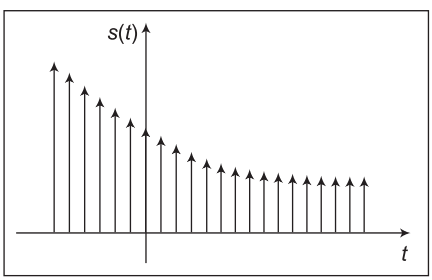


Figure 4.1.1 An Analog Discrete-Time Signal

- In *analog continuous-time signals,* the amplitude levels may be specified an infinite number of values.[1]

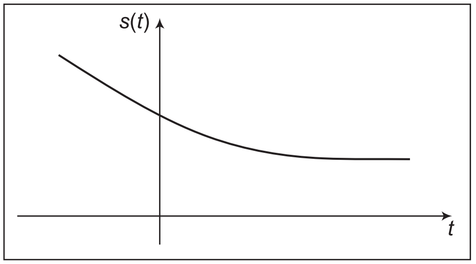


Figure 4.1.2 An Analog Continuous-Time Signal

Digital Signals

Digital signals are those signals whose amplitude levels can take on only a finite number of values.[2]

- In *digital discrete-time signals*, the amplitude levels may be specified only at discrete points of time.
- In *digital continuous-time signals,* the amplitude levels may be specified on a continuous time axis.

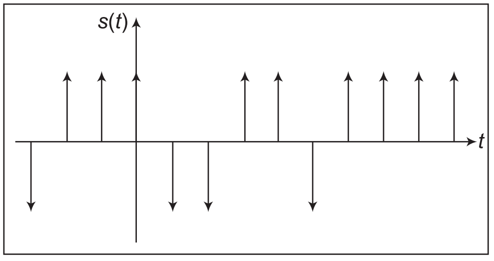


Figure 4.1.3 A Digital Discrete-Time Signal

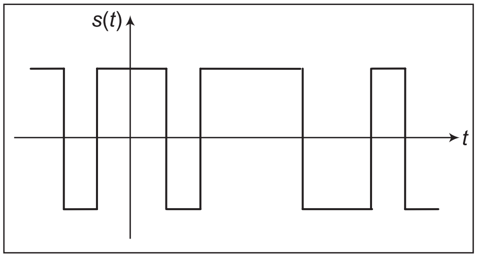


Figure 4.1.4 A Digital Continuous-Time Signal

[1] The term 'analog' describes the nature of the signal along the amplitude axis, and the terms 'discrete time' and 'continuous time' qualify the nature of the signal along the time axis. Analog signals need not be continuous-time signals only. Daily temperature average values and monthly rainfall average values of a particular place are examples of analog-discrete time signals. Speech and audio/video signals are examples of analog continuous-time signals.

[2] A binary signal is a special case of digital signal which takes only two values for logic 0 and 1. There can be multilevel digital signals referred as M-ary digital signals, where M = 2, 3, 4, …. Signals generated by analog-to-digital converters are examples of digital discrete-time signals. Signals associated with a digital computer are examples of digital binary continuous-time signals.

IMPORTANT! It may be noted that typical analog signals usually have a finite bandwidth because of smooth variations. Digital signals usually have unlimited bandwidth because of discrete nature. However, the useful bandwidth for digital signals can be determined as the range of frequencies that contains significant energy of the signal.

4.1.1 Message Sources—DMS and MARKOV

Define

We know that a *message source* is where the message (information) to be transmitted, originates.

- A *zero-memory source* or *memoryless source* implies that each message generated is independent of the previous message(s).
- If a message source generates statistically independent successive symbols 0 and 1 only, then such a source is known as a *binary memoryless source*.
- A message source which generates discrete messages is generally referred to as *discrete memoryless sources (DMS)*. Basically, it generates discrete-time random processes that take only discrete values. It also generates statistically independent symbols during successive signaling intervals.
- The symbol generated at any particular instant of time is independent of previously generated symbols. All discrete sources generate outputs which are sequences of a finite number of symbols called *alphabets*. For example, a binary source has only two finite numbers in its alphabet which are binary digits 0 and 1.

Figure 4.1.5 shows a functional block schematic of a digital communication system depicting discrete memoryless source (DMS) generating binary messages 0 and 1, binary source and channel encoder/decoder, and binary symmetric channel.

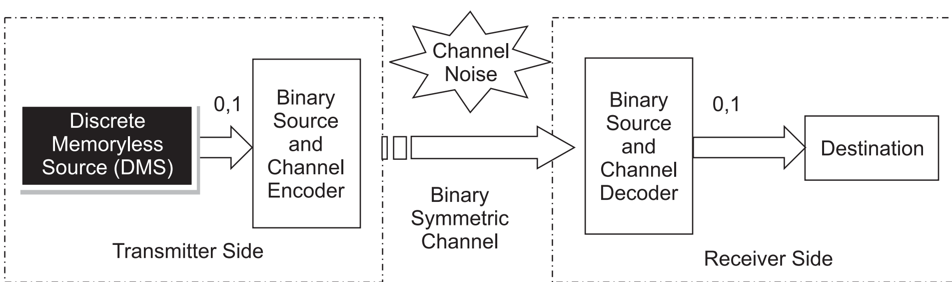


Figure 4.1.5 A Digital Communication System with DMS

Discrete Memoryless Source

A discrete source is said to be *memoryless* if the symbols generated by the source are statistically independent. This implies that the current output symbol is statistically independent from all past and future output symbols. In particular, this means that for any two symbols taken at a time, the joint probability of the two elements is simply the product of their respective probabilities.

Markov Source

A discrete source is said to have memory if the source elements composing the sequence are not independent. The dependency between symbols means that in a sequence of M symbols, there is reduced uncertainty about the M^{th} symbol when the previous $(M-1)$ symbols are known. Such a source is also called *Markov source*.

- If only the immediate past symbol influences the current symbol, the source is of first order.

- On the other hand, if the past two symbols have influence on the selection of the current symbol, the source is of second order, and so on. For example, written text.

It is possible to evolve a probabilistic model of discrete or binary memoryless source. Thus, we shall hereafter consider our message (information) source to be a discrete memoryless source unless and otherwise stated.

IMPORTANT!

4.1.2 Amount of Information

In an electronic communication system, the *information* is the message that is transmitted from the source to the destination, which was previously not known to the end user.

Recall

The *amount of information* is determined by measurements carried out according to some well-defined scientific principles. It appears to us that uncertainty, surprise, and information are all related to each other. Before the occurrence of the event, there is an amount of *uncertainty*. In fact, the basis of information theory lies on the fact that removing uncertainty on any event may be considered to be equivalent to providing information on that event. Therefore, it measures the amount of information given by the occurrence of an event in terms of the amount of uncertainty removed by its occurrence. At the instant of occurrence of an event, there is an amount of *surprise* about the content and amount of information. After the occurrence of the event, there is definitely gain in the amount of *information*.

Correlation between Information and Uncertainty

Generally, a message source generates information that is uncertain, unpredictable, and contains element of surprises. So, the *amount of uncertainty* regarding the occurrence of an event is related to the *probability* of its occurrence. Rather, it is inversely dependent on the probability, i.e., smaller the probability of occurrence of an event, means larger is the uncertainty associated with its occurrence. Thus, information contained in the occurrence of an event also depends *inversely* on its probability of occurrence. That is, if the probability of occurrence of an event is smaller then it contains more amounts of information and vice versa. Moreover, the probability of outcome for a particular event will *not* be the same for all occurrences of event. The probability of occurrence of each possible outcome is known such that the given set of probabilities satisfies the condition given as

Relationship with Probability

$$\sum_{k=1}^{K} p_k = 1, \text{ where } k = 1, 2, \ldots, K.$$

The information content of an event is defined in such a way that it monotonically decreases with increasing probability of its occurrence and approaches to zero for a probability of unity. This means that if the probability of occurrence is 1, i.e., if the outcome is 100% sure (for example, the Sun rises in the East), then such an event has zero amount of information associated with it. Hence, we find that the probability of occurrence of an event can be used to measure the amount of information associated with the occurrence of that event.

Interpretation

It is required to measure the extent of uncertainty in the information so as to know the amount of information contents contained in the signal. Consider the event which describes the generation of symbol s_k by the discrete memoryless source with probability p_k. As mentioned earlier, three distinct conditions may exist such as the following:

Three Conditions

- If $p_i = 0$ and $p_k = 1$ for all $i \neq k$ then the outcome of the source is already known. So there is no information generated.

- If the source symbols occur with low probability then there is more amount of uncertainty and surprise, and therefore, more amount of information.
- If the source symbols occur with higher probability then there is less amount of uncertainty and surprise, and therefore, less amount of information.

Question

Now, the question arises how to measure the amount of information gained after the occurrence of the event? At least we know that the amount of information is related to the inverse of the probability of occurrence. But, what function of probability?

Logarithmic Function is the Answer

Let us examine it with certain properties associated with information. It is quite obvious that the information must be a continuous as well as decreasing function of probability. If two independent events occur in sequence then the total information content should be equal to the *sum* of the individual information contents of two events. And, if we wish to obtain the same outcome in occurrence of both events happening in sequence then the total information content of the composite event is the *product* of two individual information contents. Therefore, the definition of information must be such that when probabilities are multiplied together, the corresponding contents are added. Finally, information must be non-negative, i.e., information $I \geq 0$ for $0 \leq p \leq 1$, where p stands for probability. Clearly, the *logarithm function* satisfies these requirements which can be used as a measure of information.

Physical Interpretation

Now, we are in a position to define the amount of information as function of $\log_b(1/p)$ with an occurrence of an event whose probability of occurrence is p. The amount of information in an event or a message depends only on its probability of occurrence rather than its actual content or physical interpretation. Although the probability is a dimensionless quantity, the unit of information depends on the nature of logarithm base 'b', which may be 2 (the unit of I is bit), or 10 (the unit of I is decit), or e (the unit of I is nat).

Mathematical Representation

Let a discrete memoryless source generate a number of independent messages $s_1, s_2, s_3, ..., s_K$ with their probabilities of occurrence $p_1, p_2, p_3, ..., p_K$, respectively, and the messages are correctly received. The amount of information $I(s_k)$ gained after the event s_k occurs with the probability of occurrence p_k, is defined as the logarithmic function given by Shannon as follows:

$$I(s_k) = \log_2\left(\frac{1}{p_k}\right) \qquad \text{for } k = 1, 2, \ldots, K$$

$$\Rightarrow \qquad I(s_k) = \log_2\left(\frac{1}{p_k}\right) \text{ bits} = -\log_2(p_k) \text{ bits}$$

Hence, total amount of information can be expressed as

$$\boxed{I = \sum_{k=1}^{K} \log_2\left(\frac{1}{p_k}\right) \text{ bits}}$$

Inference

The probability is always less than or equal to unity, and logarithmic function of a number which is less than one, is negative. Therefore, the amount of information is always positive. Here, the logarithmic base is mentioned as 2, so the unit of information is binary digit (*bit*).

Properties of Information

1. For a symbol with probability approaching its maximum value 1, the amount of information in it should approach its minimum value 0.

$$I(s_k) = 0; \qquad \text{for } p_k = 1$$

2. Information is always positive.

$$I(s_k) \geq 0; \quad \text{for } 0 \leq p_k \leq 1$$

3. If the probability of occurrence of an event is less, the information gain will be more and vice versa.

$$I(s_k) > I(s_i); \text{ for } p_k < p_i$$

$$I(s_k) < I(s_i); \text{ for } p_k > p_i$$

4. Total information conveyed by two statistically independent symbols is the sum of their respective information contents.

$$I(s_k\, s_l) = I(s_k) + I(s_l)$$

where s_k and s_l are statistically independent symbols.

Measure of Information

As mentioned previously, the amount of information can be measured using different units of information which can be defined corresponding to different bases of logarithms.

- ***Bit*** When the base of logarithm is '2'; the information is expressed in the unit of a bit (or, *bi*nary digi*t*). We know that the base 2, also known as binary system, is of particular importance in digital communication.
- ***Nat*** When the base of logarithm is '*e*'; the information is expressed in unit of a nat (or, *nat*ural unit).
  1 nat = $\log_2 e$ = 1.44 bits
- ***Decit*** When the base of the logarithm is '10'; the information is expressed in the unit of a decit (or, *dec*imal digi*t*), also called *Hartley*.
  1 Decit or 1 Hartley = $\log_2 10$ = 3.32 bits

IMPORTANT!

It may be noted that when no base of logarithm is specified, the unit of information is generally taken as *binit*. If the base of logarithm is assumed to be 2 for the unspecified base then the unit of information is obviously a *bit*.

One Bit of Information

Let us now examine what we mean by '1 bit' of information. Consider the example of tossing of a coin. We know that the probability of occurrence of 'heads' is equal to the probability of occurrence of 'tails'. So the probability of occurrence of each one is ½. Hence, the information associated with an event of a single tossing of a coin with the outcome as 'head' or 'tail' is given as

For $p_{(\text{head})}$ or $p_{(\text{tail})} = 1/2$, $I = \log_2\left(\dfrac{1}{1/2}\right) \text{ bit} = \log_2(2) \text{ bit} = 1 \text{ bit}$ [3]

This implies that one *bit* amount of information is gained when one of two possible and equally likely (equiprobable corresponding to p_k = ½) events occur.

Unit Conversion Formulae

The information expressed in one unit (bit, decit, or nat) can be represented in other units by the following general relationship:

$$\log_2(x) \text{ bit} = \frac{\log_e(x)}{\log_e(2)} \text{ nat} = \frac{\log_{10}(x)}{\log_{10}(2)} \text{ decit}$$

$$\Rightarrow \quad \log_2(x) \text{ bit} = 1.443 \times \log_e(x) \text{ nat} = 3.32 \times \log_{10}(x) \text{ decit}$$

$$\Rightarrow \quad \log_e(x) \text{ nat} = 2.3 \times \log_{10}(x) \text{ decit}$$

[3] If various messages are equally probable, the information content of each message is exactly equal to the minimum integer number of bits required to send the message. This is why base 2 logarithm is used, and the unit of information is called the bit of information.

Dear student... *After reading this material, you should be able to answer how to relate the amount of information with the probability of its occurrence. This is illustrated with the help of following examples.*

SOLVED EXAMPLE 4.1.1 **Maximum Probability, Zero Information**

Show that the information gained is zero at maximum probability of the outcome of an event.

Solution We know that information,

$$I(s_k) = \log_2\left(\frac{1}{p_k}\right)$$

The maximum probability of the outcome of an event, $p_k = 1$.

Therefore, $I(s_k) = \log_2(1) = 0$

Thus, the information gained is zero at maximum probability of the outcome of an event.

SOLVED EXAMPLE 4.1.2 **Amount of Information**

A communication system consists of six independent messages with probabilities 1/8, 1/8, 1/8, 1/8, 1/4, and 1/4, respectively. Find the information content in each message.

Solution We know that information,

$$I(s_k) = \log_2\left(\frac{1}{p_k}\right)$$

For given $p_1 = 1/8$, $I(s_1) = \log_2\left(\frac{1}{p_1}\right) = \log_2\left(\frac{1}{1/8}\right) = \log_2(8) = \log_2(2^3) = 3$ bits

For given $p_2 = 1/8$, $I(s_2) = \log_2\left(\frac{1}{p_2}\right) = \log_2\left(\frac{1}{1/8}\right) = \log_2(8) = \log_2(2^3) = 3$ bits

For given $p_3 = 1/8$, $I(s_3) = \log_2\left(\frac{1}{p_3}\right) = \log_2\left(\frac{1}{1/8}\right) = \log_2(8) = \log_2(2^3) = 3$ bits

For given $p_4 = 1/8$, $I(s_4) = \log_2\left(\frac{1}{p_4}\right) = \log_2\left(\frac{1}{1/8}\right) = \log_2(8) = \log_2(2^3) = 3$ bits

For given $p_5 = 1/4$, $I(s_5) = \log_2\left(\frac{1}{p_5}\right) = \log_2\left(\frac{1}{1/4}\right) = \log_2(4) = \log_2(2^2) = 2$ bits

For given $p_6 = 1/4$, $I(s_6) = \log_2\left(\frac{1}{p_6}\right) = \log_2\left(\frac{1}{1/4}\right) = \log_2(4) = \log_2(2^2) = 2$ bits

It is observed that as the probability of occurrence of message increases (from 1/8 to 1/4), the amount of information decreases (from 3 bits to 2 bits). Thus, the amount of information is inversely proportional to the probability of occurrence of an event.

SOLVED EXAMPLE 4.1.3 — Total Amount of Information

An analog signal, bandlimited to 4 kHz, is quantized in 8 levels of PCM with respective probabilities as given in Table 4.1.1.

Table 4.1.1 Data for Example 4.1.3

Quantized Level, k	1	2	3	4	5	6	7	8
Probability, p_k (k = 1 to 8)	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{20}$	$\frac{1}{20}$	$\frac{1}{20}$

Find the total amount of information presented as a PCM signal.

Solution We know that total amount of information is given as

$$I = \sum_{k=1}^{K} \log_2\left(\frac{1}{p_k}\right)$$

For the given value of $K = 8$, we can write the expression as

$$I = \log_2\left(\frac{1}{p_1}\right) + \log_2\left(\frac{1}{p_2}\right) + \log_2\left(\frac{1}{p_3}\right) + \log_2\left(\frac{1}{p_4}\right) + \log_2\left(\frac{1}{p_5}\right) + \log_2\left(\frac{1}{p_6}\right) + \log_2\left(\frac{1}{p_7}\right) + \log_2\left(\frac{1}{p_8}\right)$$

Substituting the values of p_k as given in Table 4.1.1, we get

$$I = \log_2\left(\frac{1}{1/4}\right) + \log_2\left(\frac{1}{1/5}\right) + \log_2\left(\frac{1}{1/5}\right) + \log_2\left(\frac{1}{1/10}\right) + \log_2\left(\frac{1}{1/10}\right) + \log_2\left(\frac{1}{1/20}\right) + \log_2\left(\frac{1}{1/20}\right) + \log_2\left(\frac{1}{1/20}\right)$$

$$I = \log_2(4) + \log_2(5) + \log_2(5) + \log_2(10) + \log_2(10) + \log_2(20) + \log_2(20) + \log_2(20)$$

$$I = 2 + 2.32 + 2.32 + 3.32 + 3.32 + 4.32 + 4.32 + 4.32 = 26.24 \text{ bits}$$ **Ans.**

SOLVED EXAMPLE 4.1.4 — Units of Information

A discrete memoryless source produces a binary symbol with a probability of 0.75. Determine the amount of information associated with the symbol in bits, nats, and decits.

Solution We know that the amount of information,

$$I = \log_2\left(\frac{1}{p}\right) \text{ bit}$$

For given binary symbol with $p = 0.75$, we have

$$I = \log_2\left(\frac{1}{0.75}\right) \text{ bit} = 3.32 \times \log_{10}\left(\frac{1}{0.75}\right) \text{ bit} = 0.415 \text{ bit}$$ **Ans.**

Also, the amount of information, $I = \log_e \left(\dfrac{1}{p} \right)$ nat

For given $p = 0.75$, $I = \log_e \left(\dfrac{1}{0.75} \right)$ nat $= 0.288$ nat **Ans.**

[Alternatively, using 1.44 bits = 1 nat; 0.415 bits = (1/1.44) × 0.415 = 0.288 nat]

Also, the amount of information, $I = \log_{10} \left(\dfrac{1}{p} \right)$ decit

For given $p = 0.75$, $I = \log_{10} \left(\dfrac{1}{0.75} \right)$ decit $= 0.125$ decit **Ans.**

[Alternatively, using 3.32 bits = 1 decit; 0.415 bits = (1/3.32) × 0.415 = 0.125 decit]

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 4.1.5 A discrete memoryless source produces a binary symbol with a probability of 3/4. Determine the amount of information associated with the symbol in bits.

Ex 4.1.6 An event has six possible outcomes with the probabilities $p_1 = \dfrac{1}{2}$, $p_2 = \dfrac{1}{4}$, $p_3 = \dfrac{1}{8}$, $p_4 = \dfrac{1}{16}$, $p_5 = \dfrac{1}{32}$, $p_6 = \dfrac{1}{32}$. Determine total amount of information if the event occurs six times with different outcome every time.

LET'S RECONFIRM OUR UNDERSTANDING!!

- Define Markov source.
- Write the conversion formula between bit and nat.

Self-Assessment Exercise linked to LO 4.1

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/138

Q4.1.1 Distinguish between (a) an analog signal and a digital signal, and (b) discrete memoryless source and Markov source. ○●●

Q4.1.2 Define information. Write an expression for total amount of information in terms of probability. ○○●

Q4.1.3 It is required to measure the extent of uncertainty in the information so as to know the amount of information contents contained in the signal. Hypothesize three distinct conditions for this requirement. ●●●

Q4.1.4 A discrete memoryless source produces a binary symbol with a probability of 0.4. Determine the amount of information associated with the symbol in bits, nats and decits. ○●●

Q4.1.5 A discrete memoryless source produces a binary symbol with a probability of 1/4. Determine the amount of information associated with the symbol in bits. ○●●

Q4.1.6 An analog signal, bandlimited to 4 kHz, is quantized in 8 levels of PCM with respective probabilities as given below: ●●●

Quantized Level, k	1	2	3	4	5	6	7	8
Probability, p_k (k = 1 to 8)	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{20}$	$\frac{1}{20}$	$\frac{1}{20}$

Find the total amount of information presented as PCM signal.

Note ○○● Level 1 and Level 2 Category
○●● Level 3 and Level 4 Category
●●● Level 5 and Level 6 Category

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 4.1: *Understand discrete and continuous messages, message sources, amount of information, and its measure.*

4.2 AVERAGE INFORMATION AND ENTROPY

LO 4.2

What is meant by Average Information?

The term '*average information*' is used to represent 'statistical average' and not 'arithmetic average' (which is applicable for the quantities that are deterministic in nature and hence are to be considered just once). However, the information associated with every transmitted message (also called symbol) is a non-deterministic (probabilistic, statistical) quantity and, hence, statistical average is considered to represent the average information.

Entropy of the Source

Entropy is that part of information theory which determines the maximum possible compression of the signal from the source without losing any information contents. Generally, entropy is defined in terms of the probabilistic behavior of a source of information, and it is a measure of the average amount of information per source symbol in a long message.

The average information per individual message generated by a source is known as the *entropy of the source*. It is usually expressed in *bits per symbol*.

Mathematical Representation

In order to show that the entropy of a source $\{s_k\}$ is a function of the message probabilities, consider a discrete memoryless source generating K number of different source messages (sometimes termed as source symbols) $s_1, s_2, s_3, \ldots, s_K$ having their respective probabilities of occurrences $p_1, p_2, p_3, \ldots, p_K$ such that $\sum_{k=1}^{K} p_k = 1$. Since a memoryless source implies that each message generated is independent of the previous message(s) then as per the definition of information, the information content of a message s_k with its probability of occurrence p_k is $I(s_k)$, and is given by

$$I(s_k) = \log_2\left(\frac{1}{p_k}\right) \text{ bits} = -\log_2(p_k) \text{ bits}$$

Therefore, the average information per message generated by the source is given by $\sum_{k=1}^{K} p_k I(s_k)$. As per the definition, the average information per message of a source is called its entropy, denoted by $H\{s_k\}$. Hence,

$$H\{s_k\} = \sum_{k=1}^{K} p_k I(s_k) \text{ bits/message}$$

But
$$I(s_k) = \log_2\left(\frac{1}{p_k}\right); \text{ for } k = 1, 2, \ldots, K$$

Representing $H\{s_k\}$ as H for simplification purpose to denote entropy of the source, we have

$$\Rightarrow \quad \boxed{H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right) \text{ bits/message}}$$

or,
$$\boxed{H = -\sum_{k=1}^{K} p_k \log_2(p_k) \text{ bits/message}}$$

Recap

Earlier we have thought of information as being synonymous with the amount of uncertainty, or surprise, associated with the event (or message or symbol). A smaller probability of occurrence implies more uncertainty about the occurrence of event. Hence, intuitively, the information associated with a message is a measure of the uncertainty of the message. Therefore, $\log_2\left(\frac{1}{p_k}\right)$ is a measure of the uncertainty of the message s_k, and the entropy $H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$ is the average uncertainty per message of the source that generates messages $s_1, s_2, s_3, ..., s_K$ having their respective probabilities of occurrences $p_1, p_2, p_3, ..., p_K$.

Interpretation

From engineering point of view, the information content of a message is equal to the minimum number of digits required to encode the message (for example, waveform encoding), and, therefore, the entropy H is equal to the minimum number of digits per message required for encoding. Both these interpretations lead to the fact that entropy may also be viewed as a function associated with a random variable $\{s_k\}$ that assumes values $s_1, s_2, s_3, ..., s_K$ having their respective probabilities of occurrences $p_1, p_2, p_3, ..., p_K$ as $H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$. Thus, we can associate an entropy with every discrete random variable.

ATTENTION

The entropy of a source is a function of the message probabilities. Since the entropy is a measure of uncertainty, the probability distribution that generates the maximum uncertainty will have the maximum entropy. On qualitative aspects, one can expect entropy to be maximum when all the messages are equiprobable.

Properties of Entropy

The entropy of a discrete memoryless $H = 0$, **if and only if** the probability $p_k = 1$ for some value of k, and the remaining probabilities in the set are all zero. In fact, this is the *lower limit* on the entropy which corresponds to no uncertainty.

The entropy of a discrete memoryless $H = \log_2 K$, **if and only if** the probability $p_k = 1/K$ (i.e., equiprobable) for all values of k. There is *upper limit* on the entropy, which corresponds to maximum uncertainty.

$H < \log_2 K$, if a message generated at any time is not independent of the previous message generated.

Lower Bound on Entropy

It states that the entropy $H = 0$, **if and only if** the probability $p_k = 1$ for some value of k, and the remaining probabilities in the set are all zero.

We know that entropy, $H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$

It is given that the value of probability p_k is less than or equal to unity, that is, $p_k \le 1$.

From the above equation, it implies that each term on the right-hand side (for any value of k from $k = 1$ to $k = K$) will always be positive (greater than or equal to zero).

Therefore, $H \ge 0$

Also, it is noted that the term $p_k \log_2\left(\frac{1}{p_k}\right)$ is zero, **if and only if** $p_k = 0$**,** or $p_k = 1$ (since $\log_2 1 = 0$).

Hence, the entropy $H = 0$, if and only if the probability $p_k = 0$**,** or $p_k = 1$. It implies that $p_k = 1$ for some value of k, and the remaining probabilities in the set are all zero.

Upper Bound on Entropy

It states that the entropy $H \le \log_2 K$, and $H = \log_2 K$, **if and only if** the probability $q_k = 1/K$ for all values of k.

Consider any two probability distributions:

$$\{p_1, p_2, p_3, ..., p_K\}, \text{ and } \{q_1, q_2, q_3, ..., q_K\}$$

on the alphabet $\{s_k\} = \{s_1, s_2, s_3, ..., s_K\}$ of a discrete memoryless source.

For one probability distribution, $H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$

For two probability distributions, $H = \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right)$

Converting $\log_2$ to natural $\log_e$, we have

$$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) = \sum_{k=1}^{K} p_k \frac{\log_e(q_k/p_k)}{\log_2 e}$$

$$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) = \frac{1}{\log_2 e}\sum_{k=1}^{K} p_k \log_e\left(\frac{q_k}{p_k}\right)$$

As a property of the logarithm, we know that

$$\log_e x \le (x-1); \qquad \text{for } x \ge 0$$

This is true because the straight line $y = (x - 1)$ always lies above the curve $y = \log_2 e$, and the equality $\log_2 e = (x - 1)$ holds true only at the point $x = 1$, where the line $y = (x - 1)$ is tangential to the curve $y = \log_2 e$.

By substituting $x = \left(\frac{q_k}{p_k}\right)$ in $\log_e x \le (x - 1)$, we get

$$\log_e\left(\frac{q_k}{p_k}\right) \le \left(\frac{q_k}{p_k} - 1\right)$$

Therefore, $\sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le \frac{1}{\log_2 e}\sum_{k=1}^{K} p_k\left(\frac{q_k}{p_k} - 1\right)$

$$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le \frac{1}{\log_2 e}\sum_{k=1}^{K} p_k\left(\frac{q_k - p_k}{p_k}\right)$$

$$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le \frac{1}{\log_2 e}\sum_{k=1}^{K} (q_k - p_k)$$

$$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le \frac{1}{\log_2 e}\left(\sum_{k=1}^{K} q_k - \sum_{k=1}^{K} p_k\right)$$

But sum of all probability is always equal to 1, that is,

$$\sum_{k=1}^{K} q_k = 1, \text{ and } \sum_{k=1}^{K} p_k = 1$$

$$\therefore \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le \frac{1}{\log_2 e}(1 - 1)$$

$$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le \frac{1}{\log_2 e} \times 0$$

$$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le 0$$

It can be seen that the equality holds good if $q_k = p_k$, or $(q_k/p_k) = 1$ for all values of k since $\log_2(1) = 0$.

Let $q_k = 1/K$, where $k = 1, 2, 3, \ldots, K$, corresponds to an alphabet $\{s_k\}$ with equiprobable symbols.

The entropy of a discrete memoryless source with equiprobable symbols is given by substituting $q_k = 1/K$ in the expression $H = \sum_{k=1}^{K} q_k \log_2\left(\frac{1}{q_k}\right)$, and we get

$$H = \sum_{k=1}^{K} q_k \log_2\left(\frac{1}{1/K}\right)$$

$\Rightarrow \qquad H = \sum_{k=1}^{K} q_k \log_2(K)$

$\Rightarrow \qquad H = \log_2(K)\sum_{k=1}^{K} q_k$

$\Rightarrow \qquad H = \log_2(K)\left(\frac{1}{K}+\frac{1}{K}+\frac{1}{K}+\ldots K \times \text{times}\right)$

$\Rightarrow \qquad H = \log_2(K)\left(K \times \frac{1}{K}\right)$

$H = \log_2(K)$

Thus, it is proved that $H = \log_2(K)$, if and only if the probability $q_k = 1/K$ for all values of k.

$\Rightarrow \qquad \sum_{k=1}^{K} q_k \log_2\left(\frac{1}{q_k}\right) = \log_2(K)$

Re-writing the equation $\sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le 0$, we have

$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{1/p_k}{1/q_k}\right) \le 0$

$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right) - \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{q_k}\right) \le 0$

$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right) \le \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{q_k}\right)$

Substituting p_k with q_k on the right-side term ($\because q_k = p_k$ for all values of k), we get

$\Rightarrow \qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right) \le \sum_{k=1}^{K} q_k \log_2\left(\frac{1}{q_k}\right)$

But $\qquad \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right) = H \qquad$ **… (By definition)**

and $\qquad \sum_{k=1}^{K} q_k \log_2\left(\frac{1}{q_k}\right) = \log_2(K) \qquad$ **… (As proved above)**

Therefore, $\qquad H \le \log_2(K)$

Hence, the entropy H is always less than or equal to $\log_2(K)$; the equality will hold good only if the symbols in the alphabet $\{s_k\}$ are equiprobable, that is, **if and only if** the probability $p_k = 1/K$ for all values of k.

A binary source is said to be memoryless when it generates statistically independent successive symbols 0 and 1. Consider a binary source for which the symbol 0 occurs with the probability p_0 and the symbol 1 occurs with the probability $p_1 = (1 - p_0)$. It simply implies that it is a typical case of two equiprobable events. Using $H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$, the entropy of binary memoryless source can be expressed as

Entropy of Binary Memoryless Source

$$\boxed{H = p_0 \log_2\left(\frac{1}{p_0}\right) + p_1 \log_2\left(\frac{1}{p_1}\right)}$$

$$\Rightarrow \qquad H = -p_0 \log_2(p_0) - p_1 \log_2(p_1)$$

Substituting $p_1 = (1 - p_0)$ as applicable in the case of a binary memoryless source, we have

$$\boxed{H = -p_0 \log_2(p_0) - (1 - p_0)\log_2(1 - p_0)} \text{ bits}$$

Entropy versus Symbol Probability

Let $H(p_0)$ is an entropy function of the prior probability p_0 defined in the interval [0, 1]. Mathematically, the entropy function $H(p_0)$ can be expressed as

$$H(p_0) = -p_0 \log_2(p_0) - (1 - p_0) \log_2(1 - p_0)$$

$$\Rightarrow \qquad H(p_0) = -\left[p_0 \log_2(p_0) + (1 - p_0)\log_2(1 - p_0)\right]$$

$$\Rightarrow \qquad H(p_0) = p_0 \log_2\left(\frac{1}{p_0}\right) + (1 - p_0)\log_2\left(\frac{1}{(1 - p_0)}\right)$$

In order to generalize the expression, we can replace p_0 with p.

Hence, $$H(p_0) = p \log_2\left(\frac{1}{p}\right) + (1 - p)\log_2\left(\frac{1}{(1 - p)}\right)$$

Entropy of Multilevel Memoryless Source

The results of a binary-level memoryless source can be extended for a multilevel source.

In case of binary level, the entropy becomes maximum (1 bit/symbol) when both the symbols are equiprobable.

Similarly, in case of multilevel, the entropy becomes maximum when all the symbols are equiprobable.

For equiprobable symbols, source entropy $H = \log_2(M)$ gives minimum number of bits needed to encode the symbol, where M represents the number of levels. This ensures the minimum transmission bandwidth as well as resolving the uncertainty at the receiver.

Entropy of Extended Discrete Memoryless Source

Consider a block of symbols generated by an extended discrete memoryless source. It has a source alphabet $\{s_k\}^n$ that has K^n distinct blocks, where K is the number of distinct symbols in the source alphabet $\{s_k\}$ of the original source, and n is the number of successive source symbols in each block.

In case of a discrete memoryless source, the source symbols are statistically independent. Hence, the probability of a source symbol in $\{s_k\}$ alphabet is equal to the product of the probabilities of the n source symbols in the source alphabet $\{s_k\}$.

It implies that the entropy of the extended discrete memoryless source is equal to n times H the entropy of the original discrete memoryless source. That is,

$$\boxed{H\{s_k\}^n = n \times H\{s_k\}}$$

Differential Entropy

Let a message source generate a continuous signal $x(t)$ having a finite bandwidth so that it can be completely described by its periodic sample values. We transmit this signal through a continuous channel. The set of all possible signals can then be considered an ensemble of waveforms generated by an ergodic random process. At any sampling instant, the collection of possible sample values constitutes the random variable X. If $p_X(x)$ is the corresponding probability density function of X then the average information per sample value of continuous signal $x(t)$, called the *differential entropy* of the continuous random variable, X is defined as

$$h(X) = \int_{-\infty}^{+\infty} p_X(x) \log_2 \left[\frac{1}{p_X(x)} \right] dx \text{ bits/sample}$$

By definition, X assumes a value in the interval $[x_k, x_k + \Delta x]$ with probability $p_X(x_k)\Delta x$. As Δx approaches zero, $-\log_2 \Delta x$ approaches infinity. This means that differential entropy of a continuous random variable may assume a value anywhere in the interval $(-\infty, +\infty)$ and the uncertainty associated with the variable is on the order of infinity. Since the information transmitted over a channel is actually the difference between two entropy terms that have a common reference, the information will be the same as the difference between the corresponding differential entropy terms, hence termed *differential entropy*.

Properties of Differential Entropy

- The differential entropy of a continuous random variable can be negative if a random variable X is uniformly distributed over the given interval.
- The differential entropy of a Gaussian random variable X is independent of the mean of X.
- The differential entropy of a Gaussian random variable X is uniquely determined by the variance of X.
- For a finite value of variance, the Gaussian random variable has the largest differential entropy which can be achieved by any random variable.

Joint Entropy

The *joint entropy* $H(X, Y)$ is the average uncertainty of the communication channel as a whole considering the entropy due to channel input as well as channel output.

$$H(X, Y) = \sum_{j=1}^{J} \sum_{k=1}^{K} p(x_j, y_k) \log_2 \left[\frac{1}{p(x_j, y_k)} \right]$$

where $p(x_i, y_j)$ is the joint probability of the average uncertainty of the channel input $H(X)$ and the average uncertainty of the channel output $H(Y)$.

The entropy functions for a channel with m inputs and n outputs are given as

$$H(X) = \sum_{i=1}^{m} p(x_i) \log_2 p(x_i)$$

$$H(Y) = \sum_{j=1}^{n} p(y_j) \log_2 p(y_j)$$

where $p(x_i)$ represents the input probabilities and $p(y_j)$ represents the output probabilities.

Conditional Entropy

The *conditional entropy* $H(X|Y)$ and $H(Y|X)$, also known as equivocation, is a measure of the average uncertainty remaining about the channel input after the channel output, and the channel output after the channel input has been observed, respectively.

$$H(X|Y) = \sum_{j=1}^{n} \sum_{i=1}^{m} P(x_i, y_j) \log_2 P(x_i | y_j)$$

$$H(Y|X) = \sum_{j=1}^{n}\sum_{i=1}^{m} P(x_i, y_j)\log_2 P(y_j|x_i)$$

Relationship between Joint and Conditional Entropy

The joint and conditional entropies are related with each other as

$$H(X, Y) = H(X \mid Y) + H(Y)$$

$$H(X, Y) = H(Y \mid X) + H(X)$$

Average Effective Entropy

The *average effective entropy* H_{eff} at the receiver is the difference between the entropy of the source and the conditional entropy of the message X, given Y. That is,

$$\boxed{H_{\text{eff}} = H(X) - H(X|Y)}$$

Information Rate

If a discrete memoryless source generates r messages per second, the *information rate* or the *average information per second* is defined as

$$R = rH = \frac{H}{T_s} \text{ bits/s}$$

where H is the source entropy and T_s is the time required to send a single message or symbol (=1/r). This expression indicates that a source is characterized by its entropy as well as its rate of information.

Let there be two sources of identical entropy H, generating r_1 and r_2 messages/second. The first source generates the information at a rate $R_1 = r_1H$ and the second source generates the information at a rate $R_2 = r_2H$. If $r_1 > r_2$, then $R_1 > R_2$. This means that that first source generates more information than the second source within a given time period even when the source entropy is the same.

Effective Information Bit Rate

The effective information bit rate, R_{eff} is defined as the product of actual transmitted bit rate (R) and the average effective entropy H_{eff}. That is, $R_{\text{eff}} = R \times H_{\text{eff}}$.

Coding of Information

Let a message comprising of four quantized levels of a bandlimited analog signal (obtained after sampling of the analog signal at the Nyquist rate, followed by quantizing the samples) be transmitted by binary PCM. We can represent each of the four quantized levels by a 2-bit binary code. For example, four independent quantized levels Q_1, Q_2, Q_3, and Q_4 with probabilities p_1 = 1/8, p_2 = 3/8, and p_3 = 3/8, and p_4 = 1/8, respectively can be encoded by using binary code as depicted in Table 4.2.1.

Table 4.2.1 Coding of Information

Quantization Level	Probability	Binary Code
Q1	1/8	0 0
Q2	3/8	0 1
Q3	3/8	1 0
Q4	1/8	1 1

Let us assume that the analog signal is bandlimited to B Hz. Then as per the Nyquist rate, we transmit 2B messages per second. Since each message requires 2 binits in a binary coding, we shall be transmitting information at the rate of $4B$ binits per second. It may be noted that when no base of logarithm is specified, the unit of information is generally taken as binit. We know that a binit is capable of transmitting 1 bit of information. It implies that we should be able to transmit $4B$ bits of information per second. However, with the given probabilities for each message, the actual information rate can be computed as below:

The information rate, $R = rH$ bits/second where r indicates number of messages per second, which is $2B$ in our case, and H is the average information per message or entropy of the system which is given as

$$H = \sum_{k=1}^{4} p_k \log_2\left(\frac{1}{p_k}\right) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right) + p_3 \log_2\left(\frac{1}{p_3}\right) + p_4 \log_2\left(\frac{1}{p_4}\right)$$

Using the given data as $p_1 = 1/8$, $p_2 = 3/8$, $p_3 = 3/8$, and $p_4 = 1/4$, we get

$$H = \frac{1}{8} \times \log_2\left(\frac{1}{1/8}\right) + \frac{3}{8} \times \log_2\left(\frac{1}{3/8}\right) + \frac{3}{8} \times \log_2\left(\frac{1}{3/8}\right) + \frac{1}{8} \times \log_2\left(\frac{1}{1/8}\right)$$
$$= 1.8 \text{ bits per message}$$

Therefore, the rate of information, $R = rH$ bits/s

Substituting $r = 2B$, $R = 2B \times 1.8 = 3.6B$ bits/s

This shows that we are actually substituting only $3.6B$ bits of information per second against the available $4B$ bits of information per second. It implies that we are not taking full advantage of the ability of the binary PCM to transmit information.

One Solution—Binary Coding

There are basically two ways to improve the situation. One way is to select different quantization levels such that each level is equally likely. This means, in the above example, $p_1 = p_2 = p_3 = p_4 = p = 1/4$. Then, we would find that the average information per message is

$$H = \sum_{k=1}^{4} p_k \log_2\left(\frac{1}{p_k}\right) = 4 \times p \log_2\left(\frac{1}{p}\right) = 4 \times \frac{1}{4} \log_2\left(\frac{1}{1/4}\right) = 2 \text{ bits/message}$$

and $\quad R = rH = 2B \times 2$ bits/s $= 4B$ bits/s

This is the same as the available $4B$ bits of information per second for binary coding. Thus, we find that for $M = 2^n$ messages, each message coded into n bits and are equally likely, the average information per message interval is $H = n$. In other words, since there are n bits in the message, the average information carried by an individual bit is $H/n = 2/2 = 1$ bit.[4]

Alternative Solution—Source Coding

If the messages are not equally likely and it is not feasible to change the message probabilities, then H is less than n, and each bit carries less than 1 bit of information. In such a situation, then the alternative way is to apply an appropriate *source-coding* scheme (rather than binary coding) in which, on the average, the number of bits per message is less than 2. In fact, all the messages having different probabilities are not encoded into the same number of bits. Instead, the more likely a message is, the fewer the number of bits that should be used in its code word and vice versa.[5]

[4] If the source is not memoryless, i.e., a message generated at any time is dependent on the previous messages generated, then the source entropy will be less than the entropy of the discrete memoryless source. This is because the dependence of a message on previous messages reduces its uncertainty.

[5] Several source coding schemes such as Shannon–Fano coding, Huffman coding, Lempel–Ziv coding are discussed in detail in the next chapter.

SOLVED EXAMPLE 4.2.1 **Average Information of English Language**

What is the average information in bits/character if each of the 26 characters in the English-language alphabet occur with equal probability? Neglect spaces and punctuations.

Solution The average amount of information per source message (alphabet), also known as entropy of the source, is expressed as

$$H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right) \text{ bits/character}$$

Since there are $K = 26$ characters in the English language occurring with equal probability, therefore, $p_k = \frac{1}{26}$.

$$\Rightarrow \quad H = \sum_{k=1}^{26} p_k \log_2\left(\frac{1}{p_k}\right) = \frac{1}{26}\log_2\left(\frac{1}{1/26}\right) + \frac{1}{26}\log_2\left(\frac{1}{1/26}\right) + \ldots \text{ total 26 times}$$

$$\Rightarrow \quad H = 26 \times \frac{1}{26}\log_2\left(\frac{1}{1/26}\right) = \log_2\left(\frac{1}{1/26}\right) \text{ bits/character}$$

Using $\log_2 (x) = 3.32 \times \log_{10} (x)$, we can write

Hence, $$H = 3.32 \times \log_{10}\left(\frac{1}{1/26}\right) = 4.7 \text{ bits/character}$$ **Ans.**

In actual practice, the alphabet characters do not appear with equal likelihood in the use of English language (or for that matter of any language). Thus, 4.7 bits/character represents the *upper bound limit* of average information content for the English language.

SOLVED EXAMPLE 4.2.2 **Average Information with Given Probabilities**

Calculate the average information content in bits/character for the English language if 4 letters occur with probability of 0.1; 5 letters occur with probability of 0.07; 8 letters occur with probability of 0.02, and remaining letters occur with probability of 0.01. Neglect spaces and punctuations.

Solution Neglecting the spaces and punctuations, the average information content (entropy) for 26 alphabets of the English language for given data can be calculated using $H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$ as

$$H = \sum_{k=1}^{4} 0.1 \times \log_2\left(\frac{1}{0.1}\right) + \sum_{k=1}^{5} 0.07 \times \log_2\left(\frac{1}{0.07}\right) + \sum_{k=1}^{8} 0.02 \times \log_2\left(\frac{1}{0.02}\right) + \sum_{k=1}^{9} 0.01 \times \log_2\left(\frac{1}{0.01}\right)$$

$$H = 4 \times 0.1 \times \log_2\left(\frac{1}{0.1}\right) + 5 \times 0.07 \times \log_2\left(\frac{1}{0.07}\right) + 8 \times 0.02 \times \log_2\left(\frac{1}{0.02}\right) + 9 \times 0.01 \times \log_2\left(\frac{1}{0.01}\right)$$

$$H = 0.4 \times \log_2\left(\frac{1}{0.1}\right) + 0.35 \times \log_2\left(\frac{1}{0.07}\right) + 0.16 \times \log_2\left(\frac{1}{0.02}\right) + 0.09 \times \log_2\left(\frac{1}{0.01}\right)$$

Using $\log_2(x) = 3.32 \times \log_{10} (x)$, we can write

$$H = 3.32 \times \left[0.4 \times \log_{10}\left(\frac{1}{0.1}\right) + 0.35 \times \log_{10}\left(\frac{1}{0.07}\right) + 0.16 \times \log_{10}\left(\frac{1}{0.02}\right) + 0.09 \times \log_{10}\left(\frac{1}{0.01}\right)\right]$$

$\Rightarrow \quad H = 4.17$ bits/character **Ans.**

As expected, it is less than the upper-bound limit of 4.7 bits/character for the English language.

SOLVED EXAMPLE 4.2.3 **Entropy of the Given System**

A communication system consists of six messages with probabilities 1/8, 1/8, 1/8, 1/8, 1/4, and 1/4, respectively. Determine the entropy of the system.

Solution We know that the entropy of the system,

$$H = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$$

For the given data, we have

$$H = \frac{1}{8}\times\log_2\left(\frac{1}{1/8}\right)+\frac{1}{8}\times\log_2\left(\frac{1}{1/8}\right)+\frac{1}{8}\times\log_2\left(\frac{1}{1/8}\right)$$
$$+\frac{1}{8}\times\log_2\left(\frac{1}{1/8}\right)+\frac{1}{4}\times\log_2\left(\frac{1}{1/4}\right)+\frac{1}{4}\times\log_2\left(\frac{1}{1/4}\right)$$

$$\Rightarrow \quad H = 4\times\frac{1}{8}\times\log_2\left(\frac{1}{1/8}\right)+2\times\frac{1}{4}\times\log_2\left(\frac{1}{1/4}\right)=\frac{1}{2}\times\log_2\left(\frac{1}{1/8}\right)+\frac{1}{2}\times\log_2\left(\frac{1}{1/4}\right)$$

Using $\log_2(x) = 3.32 \times \log_{10}(x)$, we can write

$$\Rightarrow \quad H = 3.32\times\left[\frac{1}{2}\times\log_{10}\left(\frac{1}{1/8}\right)+\frac{1}{2}\times\log_{10}\left(\frac{1}{1/4}\right)\right]$$

Hence, $H = 2.5$ bits/message **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 4.2.4 An analog signal is quantized into four levels which occur with probabilities $p_1 = p_2 = 0.375$ and $p_3 = p_4 = 0.125$. Determine the average information per level.

SOLVED EXAMPLE 4.2.5 **Entropy of Binary Memoryless Source**

Determine the entropy of a binary memoryless source for following three conditions:

(i) When the symbol 0 occurs with probability $p_0 = 0$.
(ii) When the symbol 0 occurs with probability $p_0 = 1$.
(iii) When the symbols 0 and 1 occur with equal probability.

Solution We know that in case of a binary memoryless source, the entropy is given by

$$H = -p_0 \log_2(p_0) - (1-p_0)\log_2(1-p_0) \text{ bits}$$

(i) Case I: $p_0 = 0$;

$$H = -0\times\log_2(0) - (1-0)\times\log_2(1-0) = 0 - \log_2(1) = -\log_2(2^0)$$

$$\Rightarrow \quad H = 0$$

(ii) Case II: $p_0 = 1$;

$$H = 1 \times \log_2 (1) - (1 - 1) \times \log_2 (1 - 1)$$

$$\Rightarrow \quad H = -1 \times \log_2(2^\circ) - (0) \times \log_2(0) = -1 \times 0 - 0$$

$$\Rightarrow \quad H = 0$$

(iii) Case III: The symbols 0 and 1 are equally probable, that is, $p_0 = p_1 = 1/2$

$$H = -\frac{1}{2} \times \log_2\left(\frac{1}{2}\right) - \left(1 - \frac{1}{2}\right) \times \log_2\left(1 - \frac{1}{2}\right)$$

$$\Rightarrow \quad H = -\frac{1}{2} \times \log_2(2^{-1}) - \left(\frac{1}{2}\right) \times \log_2(2^{-1})$$

$$\Rightarrow \quad H = -\frac{1}{2} \times (-1) - \left(\frac{1}{2}\right) \times (-1) = \frac{1}{2} + \frac{1}{2}$$

$$\Rightarrow \quad H = 1 \text{ bit}$$

It implies that the maximum entropy is 1 bit and it is achieved when the symbols 0 and 1 are equally probable.

SOLVED EXAMPLE 4.2.6 **Entropy of Binary Source**

Determine the entropy of a binary source having two independent messages, S_0 and S_1, with respective probabilities as given in the following cases:

(a) $p_0 = 0.01$ and $p_1 = 0.99$
(b) $p_0 = 0.4$ and $p_1 = 0.6$
(c) $p_0 = 0.5$ and $p_1 = 0.5$

Solution We know that the entropy of binary memoryless source is expressed as

$$H = p_0 \log_2\left(\frac{1}{p_0}\right) + p_1 \log_2\left(\frac{1}{p_1}\right)$$

(a) For given values $p_0 = 0.01$ and $p_1 = 0.99$, we get

$$H = 0.01 \times \log_2\left(\frac{1}{0.01}\right) + 0.99 \times \log_2\left(\frac{1}{0.99}\right) = 0.08$$

From the given data, it can be easily seen that the message S_1 with probability $p_1 = 0.99$ will occur most of the time as compared to the message S_0 with probably $p_0 = 0.01$. Thus, the uncertainty is very less, and, hence, the computed low value of entropy, $H = 0.08$, is justified.

(b) For given values of $p_0 = 0.4$ and $p_1 = 0.6$, we get

$$H = 0.4 \times \log_2\left(\frac{1}{0.4}\right) + 0.6 \times \log_2\left(\frac{1}{0.6}\right) = 0.97$$

From the given data, it can be seen that both the messages occur with their respective nearly equal probabilities of 0.4 and 0.6, it is quite difficult to estimate which message is likely to occur. Thus, the uncertainty is more, and hence the computed value of entropy, $H = 0.97$, is justified.

(c) For given values of $p_0 = 0.5$ and $p_1 = 0.5$, we get

$$H = 0.5 \times \log_2\left(\frac{1}{0.5}\right) + 0.5 \times \log_2\left(\frac{1}{0.5}\right) = 1.00$$

From the given data, it can be seen that both the messages occur with exactly same probabilities of 0.5 each, it is extremely difficult or almost impossible to estimate which message is likely to occur. Thus, the uncertainty is maximum, and, hence, the computed value of entropy, $H = 1.00$, is also maximum.

SOLVED EXAMPLE 4.2.7 **Entropy Function versus Symbol Probability**

Plot the variation of entropy function, $H(p)$ of a binary memoryless source with symbol probability, p.

Solution We know that the entropy function, $H(p)$ of a binary memoryless source is given by

$$H(p) = p\log_2\left(\frac{1}{p}\right) + (1-p)\log_2\left(\frac{1}{(1-p)}\right)$$

where p is the symbol probability which varies from 0 to 1.

Case I: When $p = 0$

$$H_{(p=0)} = 0 \times \log_2\left(\frac{1}{0}\right) + (1-0)\log_2\left(\frac{1}{(1-0)}\right)$$

$$H_{(p=0)} = 0 + 1 \times \log_2(1) = 0 + 1 \times \log_2(2^0)$$

$$H_{(p=0)} = 0 + 1 \times 0 = 0$$

Case II: When $p = 1/2$

$$H_{(p=1/2)} = \frac{1}{2} \times \log_2\left(\frac{1}{1/2}\right) + \left(1 - \frac{1}{2}\right)\log_2\left(\frac{1}{\left(1-\frac{1}{2}\right)}\right)$$

$$H_{(p=1/2)} = \frac{1}{2} \times \log_2(2^1) + \frac{1}{2} \times \log_2(2^1)$$

$$H_{(p=1/2)} = \frac{1}{2} \times 1 + \frac{1}{2} \times 1 = 1$$

Case III: When $p = 1$

$$H_{(p=1)} = 1 \times \log_2\left(\frac{1}{1}\right) + (1-1) \times \log_2\left(\frac{1}{(1-1)}\right)$$

$$H_{(p=1)} = 1 \times \log_2(2^0) + (0) \times \log_2\left(\frac{1}{0}\right)$$

$$H_{(p-1)} = 1 \times 0 + 0 = 0$$

The plot of the variation of entropy function $H(p)$ of a binary memoryless source with symbol probability p is shown in Figure 4.2.1.

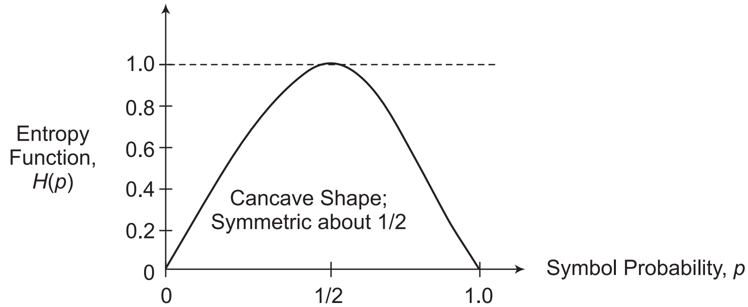


Figure 4.2.1 Plot of Entropy Function versus Symbol Probability

The following observations can be made:

(i) The value of $H(p_0)$ varies from the minimum value 0 being observed at $p_0 = 0$ as well as at $p_0 = 1$; and maximum value 1 at $p_0 = p_1 = 1/2$ (the symbols 0 and 1 are equally probable).

(ii) $H(p_0)$ increases from 0 to 1 during the value of p_0 from 0 to 1/2, and then $H(p_0)$ decreases from 1 to 0 during the value of p_0 from 1/2 to 1.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 4.2.8 Consider the binary source that generates independent symbols 0 and 1, with probabilities equal to p and $(1 - p)$, respectively. If $p = 0.1$, determine the entropy of the binary source.

SOLVED EXAMPLE 4.2.9 **Entropy of Multilevel Source**

Prove that the entropy of a multilevel discrete memoryless source is maximum, that is, $H = \log_2(M)$ when all the symbols occur with equal probability.

Solution Let there be M symbols, each having probability $p = 1/M$.

Then the entropy of a discrete multilevel memoryless source with equiprobable symbols is given by substituting $p_k = 1/M$ in the expression $H = \sum_{k=1}^{M} p_k \log_2\left(\frac{1}{p_k}\right)$. That is,

$$H = \sum_{k=1}^{M} p_k \log_2\left(\frac{1}{1/M}\right)$$

$$\Rightarrow \qquad H = \sum_{k=1}^{M} p_k \log_2(M)$$

$$\Rightarrow \qquad H = \log_2(M)\sum_{k=1}^{M} p_k$$

$$\Rightarrow \qquad H = \log_2(M)(p_0 + p_1 + p_2 + \ldots M \text{ times})$$

$$\Rightarrow \qquad H = \log_2(M)\left(\frac{1}{M}+\frac{1}{M}+\frac{1}{M}+\ldots M \text{ times}\right)$$

$$\Rightarrow \qquad H = \log_2(M)\left(M\times\frac{1}{M}\right)$$

$$\Rightarrow \qquad H = \log_2(M)$$

Thus, it is proved that maximum $H = \log_2 (M)$, **if and only if** the probability of all symbols is equal.

SOLVED EXAMPLE 4.2.10 **Number of Encoded Bits in a Multilevel Source**

Determine the number of bits required to encode multilevel discrete memoryless source if (a) $M = 2$; (b) $M = 4$; (c) $M = 8$.

Solution We know that in case of a multilevel discrete memoryless source, the maximum value of entropy occurs if and only if the probability of all symbols is equal.

(a) $M = 2$. A discrete memoryless source with two equiprobable symbols requires at least $\log_2 2 = \log_2 2^1 = 1$ binary bit to encode these symbols.

(b) $M = 4$. A discrete memoryless source with four equiprobable symbols requires at least $\log_2 4 = \log_2 2^2 = 3$ binary bits to encode these symbols.

(c) $M = 8$. A discrete memoryless source with eight equiprobable symbols requires at least $\log_2 8 = \log_2 2^3 = 3$ binary bits to encode these symbols.

SOLVED EXAMPLE 4.2.11 **Entropy of Multilevel Source**

Let there be 4 independent symbols, each having respective probabilities as $p_0 = 1/8$, $p_1 = 3/8$, $p_2 = 3/8$, and $p_3 = 1/8$. Determine the entropy of multilevel memoryless source.

Solution We know that the entropy of multilevel memoryless source is given as

$$H = \sum_{k=1}^{M} p_k \log_2\left(\frac{1}{p_k}\right); \text{ where } M \text{ is the number of levels.}$$

For given value of $M = 4$, we have

$$H = p_0\log_2\left(\frac{1}{p_0}\right)+p_1\log_2\left(\frac{1}{p_1}\right)+p_2\log_2\left(\frac{1}{p_2}\right)+p_3\log_2\left(\frac{1}{p_3}\right)$$

For given values of $p_0 = 1/8$, $p_1 = 3/8$, $p_2 = 3/8$, and $p_3 = 1/8$; we get

$$\Rightarrow \qquad H = \frac{1}{8}\log_2\left(\frac{1}{1/8}\right)+\frac{3}{8}\log_2\left(\frac{1}{3/8}\right)+\frac{3}{8}\log_2\left(\frac{1}{3/8}\right)+\frac{1}{8}\log_2\left(\frac{1}{1/8}\right)$$

$$\Rightarrow \qquad H = \frac{1}{8}\log_2(8)+\frac{3}{8}\log_2\left(\frac{8}{3}\right)+\frac{3}{8}\log_2\left(\frac{8}{3}\right)+\frac{1}{8}\log_2(8)$$

$\therefore$ $\qquad H = 1.8$ bits/message **Ans.**

SOLVED EXAMPLE 4.2.12 **Entropy of Extended Source**

Consider a discrete memoryless source with source alphabet $\{s_k\} = \{s_0, s_1, s_2\}$ with respective probabilities $p_0 = 1/4$, $p_1 = 1/4$ and $p_2 = 1/2$. Show that $H\{s_k\}^n = n \times H\{s_k\}$, where $H\{s_k\}^n$ is the entropy of the extended discrete memoryless source, n is the source symbols in the source alphabet $\{s_k\}$, and $H\{s_k\}$ is the entropy of the original discrete memoryless source.

Solution

(1) Entropy for First-order Discrete Memoryless Source: The first-order discrete memoryless source generates independent individual symbols, one at a time only. With the source alphabet $\{s_k\}$ consisting of given three symbols $\{s_k\} = \{s_0, s_1, s_2\}$, it follows that the first-order source alphabet $\{s_k\}^{(1)}$ has three symbols only.

We know that $$H\{s_k\} = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$$

For given values of $k = 0$, 1, and 2; we have

$$\Rightarrow \quad H\{s_k\} = p_0 \log_2\left(\frac{1}{p_0}\right) + p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right)$$

For given probabilities $p_0 = 1/4$, $p_1 = 1/4$ and $p_2 = 1/2$; we get

$$\Rightarrow \quad H\{s_k\} = \frac{1}{4} \times \log_2\left(\frac{1}{1/4}\right) + \frac{1}{4} \times \log_2\left(\frac{1}{1/4}\right) + \frac{1}{2} \times \log_2\left(\frac{1}{1/2}\right)$$

$$\Rightarrow \quad H\{s_k\} = \frac{1}{4} \times \log_2(4) + \frac{1}{4} \times \log_2(4) + \frac{1}{2} \times \log_2(2)$$

$$\Rightarrow \quad H\{s_k\} = \frac{1}{4} \times \log_2(2^2) + \frac{1}{4} \times \log_2(2^2) + \frac{1}{2} \times \log_2(2^1)$$

$$\Rightarrow \quad H\{s_k\} = \frac{1}{4} \times 2 + \frac{1}{4} \times 2 + \frac{1}{2} \times 1 = \frac{1}{2} + \frac{1}{2} + \frac{1}{2}$$

$$\therefore \quad H\{s_k\} = \frac{3}{2} \text{ bits}$$

(2) Entropy for Second-order Discrete Memoryless Source: An extended discrete memoryless source of second order generates a number of blocks each with two symbols at a time. With the source alphabet $\{s_k\}$ consisting of given three symbols, that is, $\{s_k\} = \{s_0, s_1, s_2\}$, it follows that the second-order source alphabet $\{s_k\}^{(2)}$ of the extended discrete memoryless source has nine symbols.

- These may be expressed as σ_k, where k varies from 0 to 8 (that is, nine symbols as σ_0, σ_1, σ_2, σ_3, σ_4, σ_5, σ_6, σ_7, and σ_8, with corresponding sequence of symbols of source alphabet $\{s_k\}$ as $\{s_{0,}\, s_0;\ s_{0,}\, s_1;\ s_{0,}\, s_2;\ s_{1,}\, s_0;\ s_{1,}\, s_1;\ s_{1,}\, s_2;\ s_{2,}\, s_0;\ s_{2,}\, s_1;\ s_{2,}\, s_2\}$ taking two symbols at a time.
- The respective probabilities $p(\sigma_k)$, where k varies from 0 to 8 can be computed as below:

$$p(\sigma_0) = p(s_0, s_0) = p(s_0) \times p(s_0) = p_0 \times p_0 = \frac{1}{4} \times \frac{1}{4} = \frac{1}{16}$$

$$p(\sigma_1) = p(s_0, s_1) = p(s_0) \times p(s_1) = p_0 \times p_1 = \frac{1}{4} \times \frac{1}{4} = \frac{1}{16}$$

$$p(\sigma_2) = p(s_0, s_2) = p(s_0) \times p(s_2) = p_0 \times p_2 = \frac{1}{4} \times \frac{1}{2} = \frac{1}{8}$$

$$p(\sigma_3) = p(s_1, s_0) = p(s_1) \times p(s_0) = p_1 \times p_0 = \frac{1}{4} \times \frac{1}{4} = \frac{1}{16}$$

$$p(\sigma_4) = p(s_1, s_1) = p(s_1) \times p(s_1) = p_1 \times p_1 = \frac{1}{4} \times \frac{1}{4} = \frac{1}{16}$$

$$p(\sigma_5) = p(s_1, s_2) = p(s_1) \times p(s_2) = p_1 \times p_2 = \frac{1}{4} \times \frac{1}{2} = \frac{1}{8}$$

$$p(\sigma_6) = p(s_2, s_0) = p(s_2) \times p(s_0) = p_2 \times p_0 = \frac{1}{2} \times \frac{1}{4} = \frac{1}{8}$$

$$p(\sigma_7) = p(s_2, s_1) = p(s_2) \times p(s_1) = p_2 \times p_1 = \frac{1}{2} \times \frac{1}{4} = \frac{1}{8}$$

$$p(\sigma_8) = p(s_2, s_2) = p(s_2) \times p(s_2) = p_2 \times p_2 = \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$$

Thus, the entropy of the second-order extended discrete memoryless source is given as

$$H\{s_k{}^2\} = \sum_{k=0}^{8} p(\sigma_k) \log_2\left(\frac{1}{p(\sigma_k)}\right)$$

$$H\{s_k{}^2\} = p(\sigma_0)\log_2\left(\frac{1}{p(\sigma_0)}\right) + p(\sigma_1)\log_2\left(\frac{1}{p(\sigma_1)}\right) + p(\sigma_2)\log_2\left(\frac{1}{p(\sigma_2)}\right) + \cdots + p(\sigma_8)\log_2\left(\frac{1}{p(\sigma_8)}\right)$$

$$H\{s_k{}^2\} = \frac{1}{16}\log_2\left(\frac{1}{1/16}\right) + \frac{1}{16}\log_2\left(\frac{1}{1/16}\right) + \frac{1}{8}\log_2\left(\frac{1}{1/8}\right) + \frac{1}{16}\log_2\left(\frac{1}{1/16}\right) + \frac{1}{16}\log_2\left(\frac{1}{1/16}\right) + \frac{1}{8}\log_2\left(\frac{1}{1/8}\right) + \frac{1}{8}\log_2\left(\frac{1}{1/8}\right) + \frac{1}{8}\log_2\left(\frac{1}{1/8}\right) + \frac{1}{4}\log_2\left(\frac{1}{1/4}\right)$$

$$H\{s_k{}^2\} = \frac{1}{16}\log_2(16) + \frac{1}{16}\log_2(16) + \frac{1}{8}\log_2(8) + \frac{1}{16}\log_2(16) + \frac{1}{16}\log_2(16) + \frac{1}{8}\log_2(8) + \frac{1}{8}\log_2(8) + \frac{1}{8}\log_2(8) + \frac{1}{4}\log_2(4)$$

$$H\{s_k{}^2\} = \frac{1}{16}\log_2(2^4) + \frac{1}{16}\log_2(2^4) + \frac{1}{8}\log_2(2^3) + \frac{1}{16}\log_2(2^4) + \frac{1}{16}\log_2(2^4) + \frac{1}{8}\log_2(2^3) + \frac{1}{8}\log_2(2^3) + \frac{1}{8}\log_2(2^3) + \frac{1}{4}\log_2(2^2)$$

$$\Rightarrow \quad H\{s_k{}^2\} = \frac{1}{16}\times 4 + \frac{1}{16}\times 4 + \frac{1}{8}\times 3 + \frac{1}{16}\times 4 + \frac{1}{16}\times 4 + \frac{1}{8}\times 3 + \frac{1}{8}\times 3 + \frac{1}{8}\times 3 + \frac{1}{4}\times 2$$

$$\Rightarrow \quad H\{s_k{}^2\} = \frac{1}{4} + \frac{1}{4} + \frac{3}{8} + \frac{1}{4} + \frac{1}{4} + \frac{3}{8} + \frac{3}{8} + \frac{3}{8} + \frac{1}{2}$$

$$\Rightarrow \qquad H\{s_k{}^2\} = \frac{2+2+3+2+2+3+3+3+4}{8} = \frac{24}{8} = 3$$

$$\therefore \qquad H\{s_k{}^2\} = 3 \text{ bits}$$

The ratio $\dfrac{H\{s_k{}^2\}}{H\{s_k\}} = \dfrac{3 \text{ bits}}{(3/2) \text{ bits}} = 2$

$$\Rightarrow \qquad H\{s_k^2\} = 2 \times H\{s_k\}$$

Therefore, in general this can be represented as

$$H\{s_k^n\} = n \times H\{s_k\}$$

Hence, Proved.

SOLVED EXAMPLE 4.2.13 Entropy of Extended Source

For a lossless channel $H(X|Y) = 0$ because $P(x_i \mid y_j) = 0$ or 1. Verify it.

Solution We know that

$$H(X|Y) = \sum_{j=1}^{n}\sum_{i=1}^{m} P(x_i, y_j)\log_2 P(x_i|y_j)$$

But $\qquad P(x_i, y_j) = P(y_j)\, P(x_i \mid y_j)$

$$\therefore \qquad H(X|Y) = \sum_{j=1}^{n}\sum_{i=1}^{m} P(y_j)P(x_i|y_j)\log_2 P(x_i|y_j)$$

$$\Rightarrow \qquad H(X|Y) = \sum_{j=1}^{n} P(y_j)\sum_{i=1}^{m} P(x_i|y_j)\log_2 P(x_i|y_j)$$

Given that $P(x_i \mid y_j) = 0$, therefore,

$$H(X|Y) = \sum_{j=1}^{n} P(y_j)\sum_{i=1}^{m} 0 \times \log_2 0$$

$$\Rightarrow \qquad H(X|Y) = 0 \qquad \text{Hence, Verified.}$$

Also, it is given that $P(x_i \mid y_j) = 1$, therefore

$$H(X|Y) = \sum_{j=1}^{n} P(y_j)\sum_{i=1}^{m} 1 \times \log_2 1 = \sum_{j=1}^{n} P(y_j)\sum_{i=1}^{m} 1 \times 0$$

$$\Rightarrow \qquad H(X|Y) = 0 \qquad \text{Hence Verified.}$$

Similarly, it can be shown that for a noiseless channel, $H(Y \mid X) = 0$ and $H(X) = H(Y)$.

SOLVED EXAMPLE 4.2.14 Average Effective Entropy

Consider the binary sequence, X, for which where *apriori* source probabilities are specified as $P(X = 0) = P(X = 1) = \dfrac{1}{2}$. Determine the average effective entropy if the channel produces one error in a binary sequence of 100 bits.

Solution For given *apriori* source probabilities $P(X = 0) = P(X = 1) = \frac{1}{2}$, we know that the entropy of the source is given as $H(X)$=1 bit/symbol.
The conditional entropy can be expressed as

$$H(X|Y) = -\sum_{X,Y} P(X,Y)\log_2 P(X|Y)$$

Probability of error, P_b = 1 in 100 bits, or $P_b = 0.01$ (Given)

$$\Rightarrow \quad H(X|Y) = -[P_b \log_2 P_b + (1 - P_b) \log_2(1 - P_b)]$$

Putting $P_b = 0.01$, we get

$$H(X|Y) = -[0.01 \times \log_2 0.01 + (1 - 0.01) \times \log_2(1 - 0.01)]$$

$$\Rightarrow \quad H(X|Y) = -[0.01 \times \log_2 0.01 + 0.99 \times \log_2 0.99$$

$$\Rightarrow \quad H(X|Y) = 0.08 \text{ bits/symbol}$$

This means that the channel introduces 0.08 bit of uncertainty to each symbol.
As per definition of average effective entropy, $H_{\text{eff}} = H(X) - H(X|Y)$

$$\therefore \quad H_{\text{eff}} = 1 - 0.08 = 0.92 \text{ bit/symbol}$$ **Ans.**

SOLVED EXAMPLE 4.2.15 Rate of Information

In a system, there are 16 outcomes per second. If the entropy of the system is 31/16 bits/message then calculate the rate of information.

Solution If a system generates '*r*' outcomes (messages) per second, the rate of information is defined as the average number of bits of information per second. Since the entropy is defined as the average number of bits of information per message (outcome), the rate of information is simply the multiplication of number of outcomes per second and the entropy of the system.

Rate of information = No. of outcomes per second × Average no. of bits per outcome

For given data, Rate of information = 16 × (31/16) = 31 bits/second **Ans.**

SOLVED EXAMPLE 4.2.16 Rate of Information

An analog information signal is bandlimited to 5 kHz, and sampled at the Nyquist rate. If the entropy of the system is 2.74 bits/message, then what is the rate of information?

Solution For given f_m = 5 kHz, Nyquist rate, $f_s = 2 \times f_m = 2 \times 5 = 10$ kHz

This implies that the message rate = 10,000 messages/second
Entropy of the system = 2.74 bits/message (given)

Rate of information = 10,000 messages/second × 2.74 bits/message = 27.4 kbps **Ans.**

SOLVED EXAMPLE 4.2.17 Information Rate

An event has four possible outcomes with probabilities of occurrence $p_1 = p_2 = 0.125$ and $p_3 = p_4 = 0.375$, respectively. Determine the entropy of the system and obtain the rate of information if there are 8 outcomes per second.

Solution The entropy of the system is given as

$$H = \sum_{k=1}^{4} p_k \log_2\left(\frac{1}{p_k}\right) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right) + p_3 \log_2\left(\frac{1}{p_3}\right) + p_4 \log_2\left(\frac{1}{p_4}\right)$$

Given $p_1 = 0.125$, $p_2 = 0.125$, $p_3 = 0.375$, and $p_4 = 0.375$

$$H = 0.125 \times \log_2\left(\frac{1}{0.125}\right) + 0.125 \times \log_2\left(\frac{1}{0.125}\right) + 0.375 \times \log_2\left(\frac{1}{0.375}\right) + 0.375 \times \log_2\left(\frac{1}{0.375}\right)$$

Hence, $H = 1.8$ bits per level **Ans.**

We know that the rate of information, $R = rH$ bits/second

For given $r = 8$, $R = 8 \times 1.8 = 14.4$ bits/second **Ans.**

SOLVED EXAMPLE 4.2.18 Average Symbol Duration

In a binary memoryless source, the symbol 0 occurs for 0.1 second and the symbol 1 occurs for 0.2 second. If the probabilities for occurrence of symbol 0 and symbol 1 are $p_{s1} = 0.4$ and $p_{s2} = 0.6$, respectively, determine the average symbol duration.

Solution The average symbol duration, $\overline{T_s} = \sum_{i=1}^{M} T_{si} p_{si}$; where T_{si} is the symbol duration and p_{si} is the probability of occurrence of the symbol, and $M = 2$ is the number of symbols produced by the source.

For the given binary source ($M = 2$), $\overline{T} = \sum_{i=1}^{2} T_{si} p_{si} = T_{s1} p_{s1} + T_{s2} p_{s2}$

Substituting $T_{s1} = 0.1$ second, $p_{s1} = 0.4$, $T_{s2} = 0.2$ second, $p_{s2} = 0.6$; we get

$$\overline{T} = 0.1 \times 0.4 + 0.2 \times 0.6 = 0.04 + 0.12 = 0.16 \text{ seconds per symbol}$$ **Ans.**

SOLVED EXAMPLE 4.2.19 Effective Information Bit Rate

Consider the binary sequence, X, for which where *apriori* source probabilities are specified as $P(X = 0) = P(X = 1) = \frac{1}{2}$. If the actual transmitted bit rate is 1000 binary symbols per second, calculate the effective information bit rate. Assume that the channel produces one error in a binary sequence of 100 bits.

Solution Here the actual transmitted bit rate, $R = 1000$ binary symbols per second (Given)

For given *apriori* source probabilities $P(X = 0) = P(X = 1) = \frac{1}{2}$, we know that the entropy of the source is given as $H(X) = 1$ bit/symbol.

The conditional entropy,

$$H(X|Y) = -\sum_{X,Y} P(X,Y) \log_2 P(X|Y)$$

Probability of error, $P_b = 1$ in 100 bits, or $P_b = 0.01$ (Given)

$$\Rightarrow \quad H(X|Y) = -[P_b \log_2 P_b + (1 - P_b) \log_2(1 - P_b)]$$

Putting $P_b = 0.01$, we get

$$H(X|Y) = -[0.01 \times \log_2 0.01 + (1 - 0.01) \times \log_2(1 - 0.01)]$$

$$\Rightarrow \quad H(X|Y) = -[0.01 \times \log_2 0.01 + 0.99 \times \log_2 0.99]$$

$$\Rightarrow \quad H(X|Y) = 0.08 \text{ bits/symbol}$$

As per definition of average effective entropy, $H_{eff} = H(X) - H(X|Y)$

$$\therefore \quad H_{eff} = 1 - 0.08 = 0.92 \text{ bit/symbol}$$

$\therefore$ Effective information bit rate, $R_{eff} = R \times H_{eff}$

$$\Rightarrow \quad R_{eff} = 1000 \times 0.92 = 920 \text{ bits/second}$$

Ans.

SOLVED EXAMPLE 4.2.20 Worst-case Effective Information Bit Rate

In the worst case, the probability of error is given as 0.5 in a binary transmission system. Show that the effective information bit rate is zero for any transmitted information bit rate. Assume that the binary sequence, X, for which where *apriori* source probabilities are specified as $P(X = 0) = P(X = 1) = 1/2$.

Solution Probability of error, $P_b = 0.5$ (Given)

$$H(X|Y) = -[P_b \log_2 P_b + (1 - P_b) \log_2(1 - P_b)]$$

$$\Rightarrow \quad H(X|Y) = -[0.5 \log_2 0.5 + (1 - 0.5 \log_2 0.5]$$

$$\Rightarrow \quad H(X|Y) = -[0.5 \log_2 0.5 + 0.5 \log_2 0.5]$$

$$\Rightarrow \quad H(X|Y) = 1 \text{ bit/symbol}$$

For the binary sequence, X, for which where *apriori* source probabilities are specified as $P(X = 0) = P(X = 1) = 1/2$, the entropy of the source, $H(X)$ =1 bit/symbol.

Using the expression, $R_{eff} = R \times [1 - H(x)]$, we get

Worst-case effective information bit rate, $R_{eff} = R \times [1 - 1] = 0$ **Hence Proved.**

SOLVED EXAMPLE 4.2.21 Entropy and Code Efficiency

The probability of four symbols produced by a discrete memoryless source are 0.5, 0.25, 0.125, and 0.125. Determine the entropy of the source. What is the code efficiency if fixed-length source code is applied to the source symbols?

Solution The entropy of the given source is given as

$$H = \sum_{k=1}^{4} p_k \log_2\left(\frac{1}{p_k}\right) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right) + p_3 \log_2\left(\frac{1}{p_3}\right) + p_4 \log_2\left(\frac{1}{p_4}\right)$$

Given $p_1 = 0.5$; $p_2 = 0.25$; $p_3 = 0.125$; $p_4 = 0.125$

$$H = 0.5 \times \log_2\left(\frac{1}{0.5}\right) + 0.25 \times \log_2\left(\frac{1}{0.25}\right) + 0.125 \times \log_2\left(\frac{1}{0.125}\right) + 0.125 \times \log_2\left(\frac{1}{0.125}\right)$$

$H = 1.75$ bits per symbol **Ans.**

For fixed length source code, 2 bits per symbol are required to encode 4 given symbols.

Therefore, code efficiency = $1.75/2 \approx 0.875$ or 87.5% **Ans.**

MATLAB simulation exercises related to entropy,

Scan the QR code given here OR visit: http://qrcode.flipick.com/index.php/135

Self-Assessment Exercise linked to LO 4.2

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/136

Q4.2.1 The four symbols produced by a discrete memoryless source has probability as 0.5, 0.25, 0.125, and 0.125 respectively. Find the entropy of the source. ○○●

Q4.2.2 Consider a ternary discrete memoryless source generating three different symbols s_0, s_1, and s_2 with probabilities 0.335, 0.335, and 0.33 respectively. Compute the maximum value of the entropy of the source. ○●●

Q4.2.3 Evaluate the entropy of the source, $H\{s_k\}$, for a discrete memoryless source that has an alphabet $\{s_k\} = \{s_1, s_2\}$ with respective probabilities $\{p_k\} = \left\{\frac{1}{4}, \frac{3}{4}\right\}$. ●●●

Q4.2.4 Evaluate and then show that the entropy of the extended source, $H\{s_k{}^3\}$ for a discrete memoryless source that has an alphabet $\{s_k\} = \{s_1, s_2\}$ with respective probabilities $\{p_k\} = \left\{\frac{1}{4}, \frac{3}{4}\right\}$ is 2.433 bits per symbol. ●●●

Q4.2.5 Consider a binary sequence, X, for which where *apriori* source probabilities are specified as $P(X = 0) = P(X = 1) = 1/2$. If the channel produces one error (on the average) in a sequence of 100 bits, compute the conditional entropy $H(X|Y)$. ○●●

Q4.2.6 A signal is sampled at the rate of 500 samples per second. The source entropy has been computed as 2.0967 bits per symbol. Find the rate of information. ○○●

Q4.2.7 An analog signal is bandlimited to B Hz, sampled at the Nyquist rate, and the samples are quantized into four independent levels Q_1, Q_2, Q_3, and Q_4 with probabilities $p_1 = p_2 = 0.125$ and $p_3 = p_4 = 0.375$, respectively. Determine the entropy of the system and obtain the information rate of the source. ○●●

Q4.2.8 In a binary memoryless source, the symbol 0 occurs for 0.2 second and the symbol 1 occurs for 0.4 second. If the probabilities for occurrence of symbol 0 and symbol 1 are $p_{s1} = 0.6$ and $p_{s2} = 0.4$, respectively, then check that the average symbol duration is 0.28 seconds per symbol. ●●●

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 4.2: *Discuss the probabilistic behavior of a source of information, known as entropy of binary- and multilevel discrete memoryless source.*

Mid-Chapter Check

So far you have learnt the following:

- Discrete and Continuous Messages, Message Sources, Amount of Information and its Measure
- Entropy of Binary- and Multilevel Discrete Memoryless Source

Therefore, you are now skilled to complete the following tasks:

MQ4.1 What is meant by discrete memoryless source? When is a discrete source said to have memory?

MQ4.2 How does probability distribution minimize uncertainty?

MQ4.3 Entropy of the source is an important parameter in digital communication systems. Elucidate its main features which help the system designers.

MQ4.4 Differentiate between the terms *Information* and *Entropy*. Mention the significance of entropy of a source in relation to uncertainty.

MQ4.5 A discrete memoryless source produces a binary symbol with a probability of 0.75. Determine the amount of information associated with the symbol in bits, nats and decits.

LO 4.1

MQ4.6 Consider a quaternary source generating symbols s_0, s_1, s_2, and s_3 with probabilities 0.5, 0.25, 0.125, and 0.125 respectively. Compute the entropy of the source.

MQ4.7 If the entropy of the binary source is computed to be 0.97 bit per symbol, and the average symbol duration is 0.28 second per symbol, then determine the entropy rate of the binary source.

MQ4.8 An analog information signal is quantized in 8 levels of a PCM system with their respective probabilities $\frac{1}{4}, \frac{1}{5}, \frac{1}{5}, \frac{1}{10}, \frac{1}{10}, \frac{1}{20}, \frac{1}{20}$, and $\frac{1}{20}$. Calculate the entropy of the system.

MQ4.9 Estimate the rate of information if a source produces messages at the rate of r messages per second.

MQ4.10 Code efficiency is the ratio of specified entropy of the system and the average code word length. The entropy of a discrete memoryless source is specified as 1.75 bits. If the average code word length of a source code applied to this source is 1.875, then how much efficient is the code?

LO 4.2

Sure of what you have learnt so far? For answers, scan the QR code

OR

visit http://qrcode.flipick.com/index.php/40

Be ready for the next sections!

LO 4.3

4.3 CHARACTERISTICS OF A DISCRETE MEMORYLESS CHANNEL

Recap

Let us try to interpret the term *channel* properly. A channel does not mean the transmitting medium only but it also includes the specifications of the type of signals such as binary, *M*-ary, simplex (one-way) or duplex (two-way), orthogonal, and the type of receiver used. It is the receiver which determines the error probability of a digital communication system.

Discrete Memoryless Channel

We know that a communication channel is said to be *discrete channel* when both the input and output alphabets $\{X_k\}_i$ and $\{Y_k\}_o$ have finite sizes. A communication channel is said to be *memoryless channel* when the current output symbol depends only on the current input symbols and not any of the previous input symbols. A *discrete memoryless channel* is a statistical model with an input X and an output Y that is a noisy version of X; both X and Y being random variables. At regular intervals with every unit of time, the communication channel accepts an input symbol X selected from an alphabet $\{X_k\}_i$ and in response, it generates an output symbol Y from an alphabet $\{Y_k\}_o$. Thus, a discrete memoryless channel may be described in terms of an input alphabet, $\{X_k\}_i$ specified as $\{X_k\}_i = \{x_1, x_2, ..., x_k\}$ and an output alphabet, $\{Y_k\}_o$ specified as $\{Y_k\}_o = \{y_1, y_2, ..., y_k\}$, and a set of conditional (transitional) probabilities specified as $p\left(\frac{y_k}{x_j}\right) = P\left(\frac{Y = y_k}{X = x_j}\right)$; for all values of j and k ($k \geq j$ or $k \leq j$, depending upon the channel coding); where $0 \leq p\left(\frac{y_k}{x_j}\right) \leq 1$ for all values of j and k.

Channel Matrix

A convenient way of describing a discrete memoryless channel is to arrange the various conditional probabilities of the communication channel in the form of *channel matrix*, *probability transition matrix*, or simply *transition matrix* as given below:

$$\left[p\left(\frac{Y}{X}\right)\right] = \begin{pmatrix} p\left(\frac{y_1}{x_1}\right) & p\left(\frac{y_2}{x_1}\right) \cdots\cdots & p\left(\frac{y_K}{x_1}\right) \\ p\left(\frac{y_1}{x_2}\right) & p\left(\frac{y_2}{x_2}\right) \cdots\cdots & p\left(\frac{y_K}{x_2}\right) \\ \vdots & \ddots & \vdots \\ p\left(\frac{y_1}{x_J}\right) & p\left(\frac{y_2}{x_J}\right) \cdots\cdots & p\left(\frac{y_K}{x_J}\right) \end{pmatrix}$$

- Each row in this matrix corresponds to a fixed channel input, and each column corresponds to a fixed channel output.
- The channel matrix includes all the specifications of the channel mentioned above and completely specifies the channel.
- For example, if we decide to use 4-ary digits instead of binary digits over the same physical channel, the channel matrix changes to 4 × 4 matrix.
- Similarly, a change in the receiver or the signal power or noise power will change the channel matrix.

Properties of Channel Matrix

The channel matrix has numerous fundamental properties. These are briefly described here.

Property 1: As evident from the channel matrix, each row contains the conditional probabilities of all possible output symbols for a particular input symbol. Thus, the sum of the elements along any row of the channel matrix is always equal to unity. That is,

$$\sum_{k=1}^{K} p\left(\frac{y_k}{x_j}\right) = 1 \text{ ; for all values of } j$$

Property 2: In the event that the channel input $X = x_j$ occurs with probability given as $p(x_j) = P(X = x_j)$; for $j = 1, 2, \ldots, J$, and the channel output $Y = y_k$ occurs with probability given as $p(y_k) = P(Y = y_k)$; for $k = 1, 2, \ldots, K$ then $[P(Y)] = [P(X)]\,[P(Y/X)]$.

Property 3: The *joint probability distribution* of random variables X and Y is given by

$$p(x_j, y_k) = P(X = x_j, Y = y_k)$$

$$\Rightarrow \quad p(x_j, y_k) = P\left(\frac{Y = y_k}{X = x_j}\right) P(X = x_j)$$

$$\Rightarrow \quad p(x_j, y_k) = p\left(\frac{y_k}{x_j}\right) p(x_j)$$

Property 4: The *marginal probability distribution* of the output random variable Y is obtained by averaging out the dependence of $p(x_j, y_k)$ on x_j, as given by

$$p(y_k) = P(Y = y_k)$$

$$\Rightarrow \quad p(y_k) = \sum_{j=1}^{J} P\left(\frac{Y = y_k}{X = x_j}\right) P(X = x_j)$$

$$\Rightarrow \quad p(y_k) = \sum_{j=1}^{J} P\left(\frac{y_k}{x_j}\right) p(x_j) \text{ ; for } k = 1, 2, \ldots, K$$

Property 5: Given the input *apriori* probabilities $p(x_j)$ and the channel matrix, that is, the matrix of transition probabilities $p\left(\frac{y_k}{x_j}\right)$, then the probabilities of the various output symbols, the $p(y_k)$, can be calculated.

Property 6: The probabilities $p(x_j)$ for $j = 1, 2, \ldots, J$ are known as the *apriori* probabilities of the various input symbols.

Property 7: Probability of error in the output is given as

$$p_e(y_k) = \sum_{\substack{k=1,\\ j \neq k}}^{K} p(y_k)$$

$$\Rightarrow \quad p_e(y_k) = \sum_{\substack{k=1,\\ j \neq k}}^{K} \sum_{j=1}^{J} p\left(\frac{y_k}{x_j}\right) p(x_j)$$

$$\Rightarrow \qquad p_e(y_k) = \sum_{\substack{k=1,\\ j\neq k}}^{K} \text{(Marginal probability distribution of output random variable } Y\text{)}$$

$$\text{or,} \qquad p_e(y_k) = \sum_{\substack{k=1,\\ j\neq k}}^{K} \sum_{j=1}^{J} \text{(Joint probability distribution of random variables } X \text{ and } Y\text{)}$$

Property 8: Probability that the received output will be correct (free of any error) is given as $p_c(y_k) = 1 - p_e(y_k)$.

Property 9: If the channel matrix of a discrete memoryless channel contains only one non-zero element in each column, the channel is described as a *lossless channel*. In a lossless channel, no source information is lost during transmission.

Property 10: If the channel matrix of a discrete memoryless channel contains only one non-zero element (i.e., unity) in each row, the channel is described as a *deterministic channel*. This implies that when a symbol is transmitted through this channel, it is known which output symbol will be received.

Property 11: If the channel matrix of a discrete memoryless channel contains only one non-zero (unity) element in each row as well as in each column (i.e., a channel is both lossless and deterministic), the channel is known as *noiseless channel*. As the number of input symbols and output symbols is the same, the channel matrix is a square matrix.

Property 12: If the channel matrix of a discrete memoryless channel contains two inputs ($x_1 = 0$, $x_2 = 1$) and two outputs ($y_1 = 0$, $y_2 = 1$), the channel is symmetric because the probability of misinterpreting a transmitted 0 as 1 is the same as misinterpreting a transmitted 1 as 0. Such a channel is known as *binary symmetric channel*.

Binary Symmetric Channel

A *Binary Symmetric Channel (BSC)* is a binary channel which can transmit only one of two symbols (0 and 1). [6]

A non-binary channel would be capable of transmitting more than two symbols, possibly even an infinite number of choices.

In BSC channel, the transmission is not perfect, and occasionally the receiver gets the wrong bit. It is assumed that the bit (1 or 0) is usually transmitted correctly. But it may be received inverted (1 for 0, or 0 for 1) with a small probability (known as crossover probability).

Binary Erasure Channel

Binary Erasure Channel (BEC) is a binary channel in which a transmitter sends a bit (0 or 1) but the receiver either receives the bit, or it receives a message that the bit was not received at all because the bit would have been erased.

The BEC channel is not perfect and sometimes the bit gets erased, that is, the receiver has no idea about the bit.

Unlike the binary symmetric channel, when the receiver gets a bit, it is 100% certain that the bit is correct. In this sense, the BEC can be said to be error-free. The only confusion arises when the bit is erased.

---

[6] A binary symmetric channel (BSC) is often used in information theory because it is one of the simplest noisy channels to analyze. *Many problems in communication theory can be reduced to a BSC*. If transmission can be made effective over the BSC then solutions can be evolved for more complicated channels too.

Figure 4.3.1 depicts a general model of a binary erasure channel.

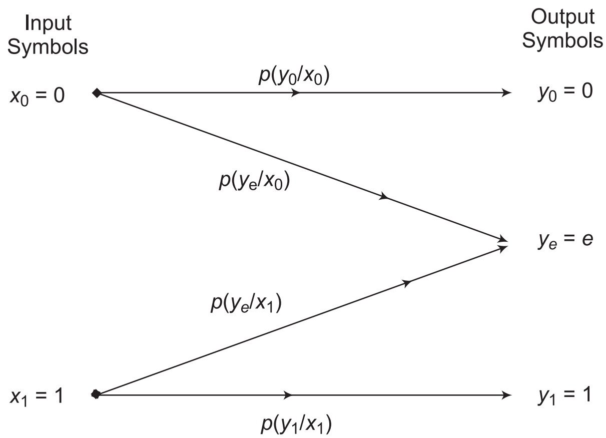


Figure 4.3.1 A General Model of a Binary Erasure Channel

Derivation

Let $\{X\}$ be the transmitted random variable with alphabet $\{0, 1\}$. A binary erasure channel with probability p, is a channel with binary input, and ternary output. Let $\{Y\}$ be the received variable with alphabet $\{0, e, 1\}$, where e is the erasure symbol. Then, the channel is characterized by the conditional probabilities:

$$P(Y = 0 \mid X = 0) = 1 - p$$

$$\Rightarrow \quad P(Y = e \mid X = 0) = p$$

$$\Rightarrow \quad P(Y = 1 \mid X = 0) = P(Y = 0 \mid X = 1) = 0$$

$$\Rightarrow \quad P(Y = e \mid X = 1) = p$$

$$\Rightarrow \quad P(Y = 1 \mid X = 1) = 1 - p$$

Interpretation of Results

- The capacity of a binary erasure channel is $(1 - p)$.
- Intuitively, $(1 - p)$ can be seen to be an upper bound on the channel capacity.
- The source cannot do anything to avoid erasure, but it can fix them when they happen. For example, the source could repeatedly transmit a bit until it gets through.
- There is no need for $\{X\}$ to code, as $\{Y\}$will simply ignore erasures, knowing that the next successfully received bit is the one that $\{X\}$ intended to send.
- Therefore, on an average, the capacity of BEC can be $(1 - p)$.
- This additional information is not available normally and hence $(1 - p)$ is an upper bound.

Discrete-Input Continuous-Output Channels

A channel is called *discrete-input continuous-output channel* if the channel input alphabet $\{X\}$ is discrete and the channel output alphabet $\{Y\}$ is continuous. The channel capacity of discrete-time channel is expressed in bits per channel use. The channel capacity of continuous-time channel is expressed in bits per second.[7]

Waveform Channels

In a *waveform channel*, continuous-time waveforms are transmitted over a channel that adds continuous-time *white Gaussian noise* to signals.

[7] A combination of a digital modulator and a physical channel is an example of a discrete-input continuous-output channel because if the channel receives discrete input signals from the digital modulator, it gives a continuous output signal depending upon the characteristics of the physical channel.

- A continuous memoryless channel is related to transmission of analog data as a discrete memoryless channel to transmission of digital data.
- A continuous-input channel can be considered as a limiting case of a discrete-input channel.

The concepts of information theory applicable to discrete channels can be extended to continuous channels. The concept of different entropies can be defined in case of continuous channels. The channel capacity of continuous-input channels can be expected to be greater than, or equal to the channel capacity of discrete input channels. Analog data is represented by continuous random variables.[8]

The entropy for continuous random variables $\{X\}$ is defined as

$$H(X) = \int_{i=-\infty}^{\infty} p(x_i)\log_2 \frac{1}{p(x_i)} dx_i$$

For a continuous channel,

$$I\{X;\, Y\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(x,y)\log_2 \frac{p(x,y)}{p_1(x)p_2(y)} dxdy$$

We know that

$$I\{X;\, Y\} = H(X) + H(Y) - H(X, Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$$

$$I\{X;\, Y\} = -\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(x,y)\log_2 \frac{p_1(x)}{p(x|y)} dxdy$$

Hence, $I\{X;\, Y\} \geq 0$

This means that the mutual information of continuous channel is non-negative.

4.3.1 Mutual Information

The *mutual information*, $I\{X: Y\}$, of a discrete memoryless channel with the input alphabet, X, the output alphabet, Y, and conditional probabilities $p(y_k/x_j)$, where $j = 1, 2, \ldots, J$, and $k = 1, 2, \ldots, K$ is defined as

$$\boxed{I\{X;Y\} = \sum_{k=1}^{K}\sum_{j=1}^{J} p(x_j, y_k)\log_2\left[\frac{p(y_k/x_j)}{p(y_k)}\right]}$$

where

- $p(x_j, y_k)$ is the *joint probability distribution* of the random variables X and Y, and is defined as $p(x_j, y_k) = p(y_k/x_j)\, p(x_j)$,
- $p(y_k/x_j)$ is the *conditional probability* or *transitional probability*, and is given as $p(y_k/x_j) = p[(Y = y_k)/(X = x_j)]$ for all values of j and k, and
- $p(y_k)$ is the marginal probability distribution of the output random variable Y, and is given as $p(y_k) = \sum_{j=1}^{J} p\left(\frac{y_k}{x_j}\right) p(x_j)$; for $k = 1, 2, \ldots, K$.

[8] A number of analog communication systems such as AM, FM, PM use continuous sources (voice signals) and, thus use the communication channel (wireless) continuously.

In order to calculate the mutual information $I\{X; Y\}$, it is necessary to know the input probability distribution, $p(x_j)$ where $j = 1, 2, \ldots, J$. Since $p(x_j)$ is independent of the channel, the mutual information $I\{X; Y\}$ of the channel with respect to $p(x_j)$ can be maximized. The mutual information of a channel depends not only on the channel but also on the way in which the channel is used.

Recap

From the basic definitions of entropy, we know that it represents uncertainty about the channel input before observing the channel output, and conditional entropy, which represents uncertainty about the channel input after observing the channel output. So it follows that the difference between entropy and conditional entropy must represent uncertainty about the channel input that is resolved by observing the channel output which is nothing but mutual information of the channel.

Mathematical Representation of Mutual Information

The mutual information of the channel can be represented as

$$I(X; Y\} = H(X) - H(X|Y)$$

where $H(X)$ is the entropy of the channel input and $H(X|Y)$ is the conditional entropy of the channel input given the channel output.

Similarly, $I\{Y; X\} = H(Y) - H(Y \mid X)$

where $H(Y)$ is the entropy of the channel output and $H(Y \mid X)$ is the conditional entropy of the channel output given the channel input.

Properties of Mutual Information

Property 1: *Symmetrical Property*
$I\{X; Y\} = I\{Y; X\}$ where $I\{X; Y\}$ is a measure of the uncertainty about the input of the channel that is resolved by observing the output of the channel, and $I\{Y; X\}$ is a measure of the uncertainty about the output of the channel that is resolved by sending the input of the channel.

Property 2: *Non-negative Property*
The information cannot be lost, on the average, by observing the output of a channel. That is, $I\{X; Y\} \geq 0$ *where* $I\{X; Y\}$ *is a measure of the uncertainty* about the input of the channel that is resolved by observing the output of the channel.

Property 3: *Joint Entropy of Input/Output Channel*
The joint entropy of the channel input and the channel output are related with each other. That is,

$$I\{X; Y\} = H(X) + H(Y) - H(X, Y)$$

where $H(X)$ is the entropy of the channel input, $H(Y)$ is the entropy of the channel output, and $H(X, Y)$ is the joint entropy of the channel input and channel output.

Property 4

$$I\{X; Y\} = H(X) - H(X|Y)$$

where $H(X)$ is the differential entropy of X;
$H(X \mid Y)$ is the conditional entropy of X, given Y,
and is defined as

$$H(X|Y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p_{X,Y}(x,y)\log_2\left[\frac{1}{p_X(x|y)}\right] dxdy$$

For a *lossless channel*, $H(X|Y) = 0$, which leads to $I\{X; Y\} = H(X)$. Therefore, the mutual information is equal to the source entropy and no source information is lost during transmission.

As a result, the channel capacity of the lossless channel is given by $C = \{P(x_j)\} \max H(X) = \log_2 M$ bits/message.

Property 5

$$I\{X;Y\} = H(Y) - H(Y|X)$$

where $H(Y)$ is the differential entropy of Y; and $H(Y|X)$ is the conditional entropy of Y, given X, and is defined as

$$H(Y|X) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p_{X,Y}(x,y)\log_2\left[\frac{1}{p_X(y|x)}\right]dxdy$$

The mutual information between a pair of continuous random variables X and Y is given as

$$I\{X;Y\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p_{X,Y}(x,y)\log_2\left[\frac{p_X(x|y)}{p_X(x)}\right]dxdy$$

where $p_{X,Y}(x, y)$ is the joint probability density function of X and Y, and $p_X(x|y)$ is the conditional probability density function of X, given $Y = y$.

ATTENTION In case of a *deterministic channel* (when a particular input symbol is transmitted through it, it is known which output symbol will be received), $H(Y|X) = 0$ for all input distributions of $P(x_j)$. This leads to $I\{X; Y\} = H(Y)$, i.e., **the information transfer is equal to the output entropy**.

SOLVED EXAMPLE 4.3.1 Symmetry of Mutual Information of a Channel

Prove that the mutual information of a channel is symmetric, that is, $I\{X; Y\} = I\{Y; X\}$.

Solution We know that the entropy,

$$H(X) = \sum_{j=1}^{J} p(x_j)\log_2\left(\frac{1}{p(x_j)}\right)$$

The sum of the elements along any row of the channel matrix is always equal to unity. That is, $\sum_{k=1}^{K} p(y_k|x_j) = 1$; for all values of j

$$\therefore \quad H(X) = \sum_{j=1}^{J} p(x_j)\log_2\left(\frac{1}{p(x_j)}\right)\sum_{k=1}^{K} p(y_k|x_j)$$

$$\Rightarrow \quad H(X) = \sum_{j=1}^{J}\sum_{k=1}^{K} p(y_k|x_j)p(x_j)\log_2\left(\frac{1}{p(x_j)}\right)$$

But $\quad p(y_k|x_j)\, p(x_j) = p(x_j, y_k)$

$$\therefore \quad H(X) = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j,y_k)\log_2\left(\frac{1}{p(x_j)}\right)$$

The conditional entropy, $H(X|Y) = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j,y_k)\log_2\left(\frac{1}{p(x_j|y_k)}\right)$

Using the expression, $I\{X;\,Y\} = H(X) - H(X|Y)$, we get

$$I\{X;\,Y\} = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\log_2\left(\frac{1}{p(x_j)}\right) - \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\log_2\left(\frac{1}{p(x_j|y_k)}\right)$$

$$\Rightarrow \qquad I\{X;\,Y\} = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\left[\log_2\left(\frac{1}{p(x_j)}\right) - \log_2\left(\frac{1}{p(x_j|y_k)}\right)\right]$$

$$\Rightarrow \qquad I\{X;\,Y\} = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\left[\log_2\left(\frac{p(x_j|y_k)}{p(x_j)}\right)\right]$$

From the definition of joint probability distribution of the random variable X and Y, we have

$$p(x_j,\, y_k) = p(y_k|x_j)\, p(x_j) = p(y_k)\, p(x_j|y_k)$$

$$\Rightarrow \qquad \frac{p(x_j|y_k)}{p(x_j)} = \frac{p(y_k|x_j)}{p(y_k)}$$

Therefore, $$I\{X;\,Y\} = \sum_{k=1}^{K}\sum_{j=1}^{J} p(x_j, y_k)\left[\log_2\left(\frac{p(y_k|x_j)}{p(y_k)}\right)\right]$$

Similarly, using the expression, $I\{Y;\,X\} = H(Y) - H(Y|X)$, we can get

$$I\{Y;\,X\} = \sum_{k=1}^{K}\sum_{j=1}^{J} p(x_j, y_k)\left[\log_2\left(\frac{p(y_k|x_j)}{p(y_k)}\right)\right]$$

Hence, $I\{X;\,Y\} = I\{Y;\,X\}$ **Hence, Proved.**

This shows that the mutual information of a channel is symmetric.

SOLVED EXAMPLE 4.3.2 Symmetry of Mutual Information of a Channel

Prove that the mutual information of a channel is always non-negative, that is, $I\{X;\,Y\} \geq 0$; and $I\{X;\,Y\} = 0$ **if and only if**, the input and output symbols of the channels are statistically independent, that is $p(x_j,\, y_k) = p(x_j)\, p(y_k)$ for all values of j and k.

Solution We know that $p(x_j,\, y_k) = p(x_j|y_k)\, p(y_k)$

$$\Rightarrow \qquad p(x_j|y_k) = \frac{p(x_j, y_k)}{p(y_k)}$$

Substituting it in the expression $I\{X;Y\} = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\left[\log_2\left(\frac{p(x_j|y_k)}{p(x_j)}\right)\right]$, the mutual information of the channel can be written as

$$I\{X;\,Y\} = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\left[\log_2\left(\frac{p(x_j, y_k)}{p(x_j)p(y_k)}\right)\right]$$

Using the fundamental inequality, $\sum_{k=1}^{K} p_k \log_2\left(\frac{q_k}{p_k}\right) \le 0$, we obtain $I\{X;Y\} \ge 0$ and $I\{X;Y\} = 0$, **if and only if**, $p(x_j, y_k) = p(x_j)\, p(y_k)$ for all values of j and k.

This shows that the mutual information of a channel is always non-negative.

SOLVED EXAMPLE 4.3.3 Mutual Information and Joint Entropy

Prove that the mutual information of a channel is related to the joint entropy of the channel input and channel output by $I\{X;Y\} = H(X) + H(Y) - H(X, Y)$.

Solution We know that the joint entropy of the channel input and channel output is given by

$$H(X, Y) = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\log_2\left[\frac{1}{p(x_j, y_k)}\right]$$

$$\Rightarrow \quad H(X, Y) = \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\log_2\left[\frac{p(x_j)p(y_k)}{p(x_j, y_k)}\right] + \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\log_2\left[\frac{1}{p(x_j)p(y_k)}\right]$$

But the first term $\sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\left[\log_2\left(\frac{p(x_j, y_k)}{p(x_j)p(y_k)}\right)\right] = -I\{X;Y\}$

$$\therefore \quad H(X, Y) = -I\{X;Y\} + \sum_{j=1}^{J}\sum_{k=1}^{K} p(x_j, y_k)\log_2\left[\frac{1}{p(x_j)p(y_k)}\right]$$

$$\Rightarrow \quad H(X, Y) = -I\{X;Y\} + \sum_{j=1}^{J}\log_2\left[\frac{1}{p(x_j)}\right]\sum_{k=1}^{K} p(x_j, y_k) + \sum_{k=1}^{K}\log_2\left[\frac{1}{p(y_k)}\right]\sum_{j=1}^{J} p(x_j, y_k)$$

$$\Rightarrow \quad H(X, Y) = -I\{X;Y\} + \sum_{j=1}^{J} p(x_j)\log_2\left[\frac{1}{p(x_j)}\right] + \sum_{k=1}^{K} p(y_k)\log_2\left[\frac{1}{p(y_k)}\right]$$

$$\Rightarrow \quad H(X, Y) = -I\{X;Y\} + H(X) + H(Y)$$

Hence, $I\{X;Y\} = H(X) + H(Y) - H(X, Y)$ **Hence, Proved**

SOLVED EXAMPLE 4.3.4 A Binary Symmetric Channel Model

Draw a general model of a binary symmetric channel. Show that the mutual information of the binary symmetric channel is given by $I\{X;Y\} \le 1 - H(p)$, where $H(p)$ is the entropy of the channel.

Solution Figure 4.3.1 depicts a general model of a binary symmetric channel.

A binary symmetric channel is a channel with binary input and binary output and probability of error p; that is, if $\{X\}$is the transmitted random variable and $\{Y\}$ the received variable, then the channel is characterized by the conditional probabilities:

$$P(Y = 0 \mid X = 0) = 1 - p$$

$$\Rightarrow \quad P(Y = 1 \mid X = 0) = p$$

$$\Rightarrow \quad P(Y = 1 \mid X = 0) = P(Y = 0 \mid X = 1) = p$$

$$\Rightarrow \qquad P(Y = 0 \mid X = 1) = p$$

$$\Rightarrow \qquad P(Y = 1 \mid X = 1) = 1 - p$$

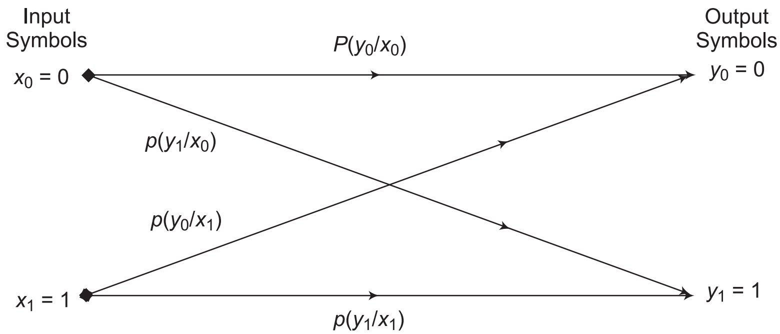


Figure 4.3.2 General Model of a Binary Symmetric Channel

It is assumed that $0 \le p \le 1/2$. If $p > 1/2$ then the receiver can swap the output (interpret 1 when it sees 0, and vice versa) and obtain an equivalent channel with probability $1 - p \le 1/2$.
The mutual information is given by

$$I\{X; Y\} = H(Y) - H(Y|X)$$

$$\Rightarrow \qquad I\{X; Y\} = H(Y) - \sum_x p(x) H(Y \mid X = x)$$

$$\Rightarrow \qquad I\{X; Y\} = H(Y) - \sum_x p(x) H(p)$$

where $\qquad H(p) = -p \log_2 p - (1-p) \log_2 (1-p)$

$$\Rightarrow \qquad I\{X; Y\} = H(Y) - H(p) \qquad \because \quad \sum_x p(x) = 1$$

$$\Rightarrow \qquad I\{X; Y\} \le 1 - H(p)$$

The inequality sign follows because $\{Y\}$ is a binary random variable.

If $p(Y = 1) = p(Y = 0) = 0.5$, **then and only then** $I\{X; Y\} = 1 - H(p)$. This is equivalent to uniform input distribution $p(X = 1) = p(X = 0) = 0.5$.

Self-Assessment Exercise linked to LO 4.3

Q4.3.1 What do you mean by a channel from the engineering perspective?

Q4.3.2 A discrete memoryless channel can be best described in the form of channel matrix. Represent it mathematically. Give an account of various properties of channel matrix.

Q4.3.3 Prove the following properties of mutual information:

$$I\{X; Y\} = I\{Y; X\} \text{ and } I\{X; Y\} \ge 0.$$

Q4.3.4 Define mutual information for a lossless channel and deterministic channel.

Q4.3.5 Compare and contrast between binary symmetric channel and binary erasure channel.

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/140

> ✓ **If you have been able to solve the above exercises then you have successfully mastered**
>
> **LO 4.3:** *Illustrate the properties of discrete memoryless channel and mutual information.*

LO 4.4

4.4 SHANNON'S CHANNEL-CODING THEOREM

Recap

We have seen that the messages of a source with entropy *H* can be encoded by using an average of *H* bits (or, digits) per message. This type of source encoding has zero redundancy. All real-life communications channels are affected by noise. Hence, if we transmit these source-encoded data over a noisy communications channel, there will be some bits received in errors.

Example

In a typical noisy channel, the probability of bit error may be as high as 10^{-2} (that is, on an average, 1 bit out of every 100 bits that are transmitted over this channel will be received in error).

Reliable Data Transmission—A Necessity

Various application areas require high reliability of digital communication, for example, 10^{-6} for voice band data and Internet access, 10^{-7} for video telephony and high speed computing. Such a highly reliable communication over noisy (unreliable) channel can be carried out by introducing channel coding in which the controlled amount of redundant data is added in the transmitted data stream. The channel decoder exploits this redundancy so as to reconstruct the original source data sequence as accurately as possible.

Now the question arises here as how to determine the number of redundant bits to be added for desired level of reliable communication. Before that we must know whether there exists a channel coding scheme which can ensure small arbitrarily probability of error when data is sent over a given noisy channel. The answer is provided by Shannon's channel-coding theorem, also known as Shannon's second theorem.

Shannon's Channel-Coding Theorem

Let a discrete memoryless source (DMS) with an alphabet *X* have entropy $H(X)$ and produce symbols every T_s seconds.

Let a discrete memoryless channel have a capacity *C* and be used once every T_s seconds.

Recall

The communication channel is said to be memoryless channel when the current output symbol depends only on the current input symbols and not any of the previous input symbols.

Then as per the statement of *Shannon's channel-coding theorem*, if

$$\frac{H(X)}{T_s} \le \frac{C}{T_c}$$

there exists a coding scheme for which the source output can be transmitted over the noisy channel and be reconstructed with an arbitrarily small probability of error.

Conversely if

$$\frac{H(X)}{T_s} \le \frac{C}{T_c}$$

then it is not possible to transmit source data over the noisy channel and reconstruct it with an arbitrarily small probability of error.

For the binary symmetric channel, for which entropy $H(X) = 1$ bit, we can say that

$$\frac{1}{T_s} \leq \frac{C}{T_c}$$

In such situation, the probability of error can be made arbitrarily low by the use of an appropriate channel coding scheme.

$$\Rightarrow \qquad \frac{T_c}{T_s} \leq C$$

But the ratio $\frac{T_c}{T_s}$ equals the information rate, R of the channel encoder (sometimes known as code rate), that is,

$$R = \frac{T_c}{T_s}$$

Hence, $$\boxed{R \leq C}$$

Interpretation

Given a source of M equally likely messages, with $M \gg 1$, which is generating information at a rate R, are to be transmitted over a noisy channel with channel capacity C (bits/sec). Then, if $R \leq C$, then there exists a *channel coding* scheme such that the transmitted messages can be received in a receiver with an arbitrarily small probability of error.

Significance of Theorem

Shannon's channel-coding theorem is a very important result in information theory. According to this theorem, one can transfer information through a communication channel of capacity C (bits/sec) with an arbitrarily small probability of error if the source of information rate R is less than or equal to C. Thus, this theorem indicates that even in presence of inevitable channel noise during transmission of digital data, error-free transmission is possible if $R \leq C$.

The $R = C$ puts the fundamental limit on the rate of information at which reliable communication can be carried out over a noisy DMS channel. It is worthwhile to note here that the Shannon's channel-coding theorem tells us about the existence of some channel codes that can achieve reliable communications in a noisy environment. But it does not give us any clue about the construction of these codes. In the next chapter we shall discuss some good channel codes.

Implementation in a Binary Symmetric Channel (BSC)

We consider a digital communication system using binary symmetric channel. We know that a binary symmetric channel (BSC) is a binary channel which can transmit only one of two symbols 0 and 1. Due to channel noise, occasionally the receiver gets the inverted bit (i.e., 1 for 0, or 0 for 1) with a small probability (also known as *crossover probability*).

Figure 4.4.1 shows a functional block schematic of a digital communication system using binary symmetric channel.

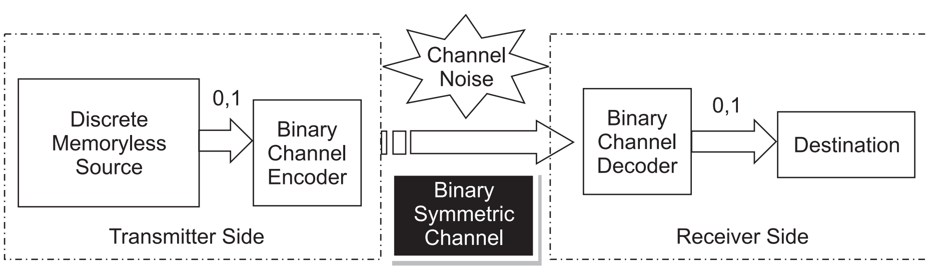


Figure 4.4.1 Binary Symmetric Channel

Entropy of BSC

Consider a discrete memoryless source that generates equally likely binary symbols (0s and 1s) once every T_s seconds where T_s is the symbol period. That means, the symbol 0 occurs with probability p_0 (say), and symbol 1 occurs with probability $p_1 = (1 - p_0)$, such that $p_0 = p_1 = \frac{1}{2}$, as applicable to binary symmetric channel.

The entropy of such a source is given by

$$H(X) = p_0 \log_2\left(\frac{1}{p_0}\right) + p_1 \log_2\left(\frac{1}{p_1}\right)$$

$$\Rightarrow \quad H(X) = \frac{1}{2}\log_2\left(\frac{1}{1/2}\right) + \frac{1}{2}\log_2\left(\frac{1}{1/2}\right)$$

$$\Rightarrow \quad H(X) = \frac{1}{2}\log_2(2^1) + \left(\frac{1}{2}\right)\log_2(2^1)$$

$$\Rightarrow \quad H(X) = \frac{1}{2}\times 1 + \frac{1}{2}\times 1 = \frac{1}{2} + \frac{1}{2}$$

$$\Rightarrow \quad H(X) = 1 \text{ bit/source symbol}$$

Thus, the entropy $H(X)$ attains its maximum value of 1 bit/source symbol in case of binary symmetric channel.

Channel Capacity of BSC

Let the information data sequence from the discrete memoryless source be applied to a binary channel encoder which produces a symbol once every T_c seconds.

Hence, the encoded symbol transmission rate is $1/T_c$ symbols/second.

The channel encoder occupies the use of a binary symmetric channel once every T_c seconds.

Hence, the channel capacity per unit time = C/T_c bits/second where C is the channel capacity.

We know that the channel capacity C of binary symmetric channel is related to the entropy $H(X)$ of binary memoryless source and is given as

$$C = 1 - H(X)$$

But

$$H(X) = p_0 \log_2\left(\frac{1}{p_0}\right) + p_1 \log_2\left(\frac{1}{p_1}\right)$$

Substituting $p_0 = p$, and $p_1 = (1 - p)$, we get

$$C = 1 - \left\{ p\log_2\left(\frac{1}{p}\right) + (1-p)\log_2\left(\frac{1}{(1-p)}\right)\right\}$$

Application of this result is illustrated in the following solved example.

SOLVED EXAMPLE 4.4.1 Application of Shannon's Channel-Coding Theorem

Consider a binary symmetric channel with transition probability, $p = 0.01$. Show that a suitable channel code exists for information rate, $R \leq 0.9192$.

Solution A binary symmetric channel with transition probability, p is shown in Fig. 4.4.2.

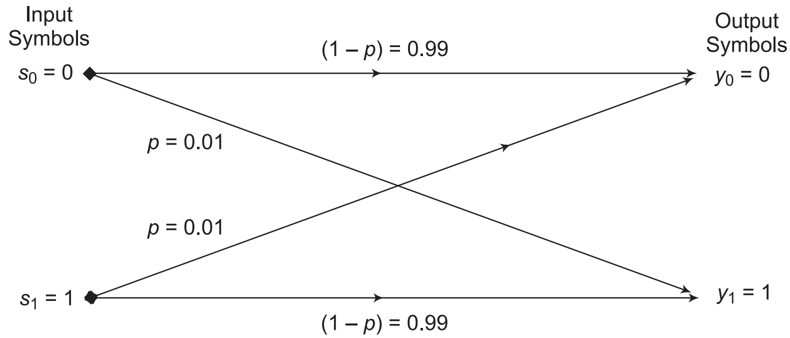


Figure 4.4.2 Binary Symmetric Channel with $p = 0.01$

We know that

$$C = 1 - \left\{ p \log_2\left(\frac{1}{p}\right) + (1-p)\log_2\left(\frac{1}{(1-p)}\right) \right\}$$

Since value of $p = 0.01$, we have

$$\Rightarrow \qquad C = 1 - \left\{ 0.01 \times \log_2\left(\frac{1}{0.01}\right) + (1-0.01) \times \log_2\left(\frac{1}{(1-0.01)}\right) \right\}$$

$$\Rightarrow \qquad C = 0.9192$$

According to Shannon's theorem, for any arbitrary small probability of error, that is, $\varepsilon > 0$, we have

$$R \leq C$$

$$\Rightarrow \qquad R \leq 0.9192$$

This clearly shows that there exists a code of large enough length, n and information rate, R and an appropriate decoding algorithm, such that when the code is used on the given channel, the average probability of decoding error is less than ε.

4.4.1 Channel Capacity

What is a Gaussian Channel?

A channel that possesses Gaussian noise characteristics is known as a *Gaussian channel*. The study of the characteristics of the Gaussian channel is very much essential because these channels are often encountered in practice. The results obtained with Gaussian channel generally provide a lower bound on the performance of a system with non-Gaussian channel.

Recall

We know that when the thermal noise power has a uniform power spectral density, it is referred to as *white noise* (the adjective 'white' is used in the same sense as it is used with white light, which contains equal amount of the whole range of frequencies within the visible band of electromagnetic transmission). In other words, the power spectral density of white noise is flat for all frequencies (from $-\infty$ to $+\infty$).

AWGN Channel

Generally, the probability of occurrence of the white noise level is represented by Gaussian distribution function, and it is known as *white Gaussian noise*. If the band-limited white

Gaussian noise is linearly added with input during transmission through a channel, then it is called *additive white Gaussian noise* (AWGN), and the channel is called *AWGN channel*.

We have seen that the theoretical channel capacity C is the maximum error-free information rate achievable on an optimum digital communication system. This is true without any restrictions except for channel bandwidth B, transmitted signal power P, and white Gaussian channel noise power spectral density $N_0/2$ ($N_0/2$ is two-sided power spectral density of the noise in Watts/Hz).

Shannon Channel Capacity Theorem

Shannon channel capacity theorem establishes a relationship between the *channel capacity* of a continuous channel of bandwidth B Hz, influenced by additive white Gaussian noise of power spectral density $N_0/2$ and limited in bandwidth to B Hz.

Consider a zero-mean stationary random process $X(t)$ that is band-limited to B Hz. Let X_k (where $k = 1, 2, \ldots, n$) denote the continuous random variables obtained by uniform sampling of the process $X(t)$ at the rate of $2B$ samples per second, as per the Nyquist criterion. We refer to X_k as a sample of the transmitted signal. These samples are then transmitted in T seconds over a noisy channel, also band-limited to B Hz. Hence, the number of samples, n is given by

$$n = 2B \times T$$

The output of the channel is influenced by additive white Gaussian noise (AWGN) of zero-mean and two-sided power spectral density $N_0/2$. The noise is also band-limited to B Hz.

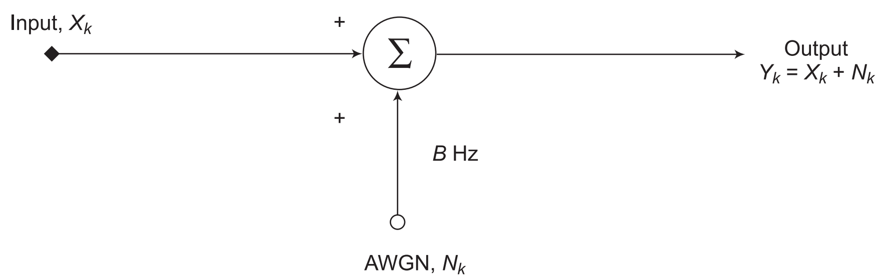


Figure 4.4.3 Simple Model of a Discrete-Time Memoryless Gaussian Channel

Mathematical Analysis

Let the continuous random variable $Y_k (k = 1, 2, \ldots, n)$ denote samples of the statistically independent received signal. Then

$$Y_k = X_k + N_k \qquad (k = 1, 2, \ldots, n)$$

The noise sample, N_k is Gaussian with zero mean and variance given by

$$\sigma^2 = N_0 B$$

The *information capacity*, or *channel capacity* is defined as the maximum of the mutual information between the channel input X_k and the channel output Y_k over all distributions on the input X_k that satisfy the power constraint as $E[X_k^2] = P$, where P is the *average transmitted power* because the transmitter is *power limited*. It is, therefore, reasonable to define the channel capacity of the channel as

$$C = \max\{I(X_k; Y_k) : E[X_k^2] = P\}$$

where $I(X_k;\ Y_k)$ denote the average mutual information between a sample of the transmitted signal, X_k, and the corresponding sample of the received signal, Y_k, and the maximization is performed with respect to the probability density function of X_k.

We know that $I(X_k;\ Y_k) = H(Y_k) - H(Y_k | X_k)$

where $H(Y_k)$ is the differential entropy of sample Y_k of the received signal and $H(Y_k | X_k)$ is the conditional differential entropy of Y_k, given X_k.

Since X_k and N_k are independent random variables, and $Y_k = X_k + N_k$, therefore,

$$H(Y_k | X_k) = H(N_k)$$

Hence, $$I(X_k;\ Y_k) = H(Y_k) - H(N_k)$$

Since $H(N_k)$ is independent of the distribution of X_k, maximizing $I(X_k;\ Y_k)$ requires maximizing $H(Y_k)$. For $H(Y_k)$ to be the maximum, Y_k has to be the Gaussian random variable. Since it is assumed that N_k is Gaussian, the sample X_k of the transmitted signal must be Gaussian too. Thus, channel capacity is maximum when the samples of the transmitted signal are also Gaussian in nature. That means

$$C = I(X_k;\ Y_k):\ X_k \text{ Gaussian}$$

$$\Rightarrow \quad C = H(Y_k) - H(N_k):\ X_k \text{ Gaussian}$$

It may be noted here that the *power-limited Gaussian channel* models are not only theoretical. There are many practical communication channels including radio and satellite links.

Example

This principle is used in spread spectrum communication systems such as code division multiple access (CDMA) based 2G and 3G mobile communication systems.

The variance of sample Y_k of the received signal equals $P + \sigma^2$. Therefore, the differential entropy of Y_k is given as

$$H(Y_k) = \frac{1}{2}\log_2\left[2\pi e(P+\sigma^2)\right]$$

where P is the average power of power-limited input signal, that is, $E[X_k^2] = P$ and σ^2 is the variance of noise sample, N_k.

The differential entropy of N_k, $H(N_k) = \frac{1}{2}\log_2(2\pi e\sigma^2)$. Thus,

$$C = \frac{1}{2}\log_2\left[2\pi e(P+\sigma^2)\right] - \frac{1}{2}\log_2(2\pi e\sigma^2)$$

$$\Rightarrow \quad C = \frac{1}{2}\log_2\left[\frac{2\pi e(P+\sigma^2)}{2\pi e\sigma^2}\right]$$

$$\Rightarrow \quad C = \frac{1}{2}\log_2\left(1+\frac{P}{\sigma^2}\right) \text{ bits per channel use}$$

For transmission of n samples of the process $X(t)$ in T seconds, the channel is used n times. The channel capacity per unit time is given by

$$C = \left(\frac{n}{T}\right)\times\frac{1}{2}\log_2\left(1+\frac{P}{\sigma^2}\right) \text{ bits/second}$$

Using $n = 2\,BT$, and $\sigma^2 = N_0 B$, we have

$$\boxed{C = B\log_2\left(1+\frac{P}{N_0 B}\right)} \text{ bits/second}$$

IMPORTANT! This is the expression which relates the channel capacity or information capacity of a channel having B Hz, influenced by additive white Gaussian noise (AWGN) of power spectral density $N_0/2$ and band-limited to B Hz, where P is the average transmitted power.

Shannon-Hartley Theorem

Channel capacity theorem, also known as the *information capacity theorem,* is one of the most remarkable results of information theory. It highlights the trade-off among three key system parameters: channel bandwidth (B Hz), average transmitted power (P Watts) or equivalently average transmitted signal power S Watts, and one-sided noise power spectral density N_0 at the channel output, or $N = 2B \times (N_0/2) = N_0 B$ is the average noise power in Watts. Therefore, we can re-write the above expression as

$$\boxed{C = B\log_2\left(1+\frac{S}{N}\right)}$$

where C represents the channel capacity of a white-noise band-limited Gaussian channel (expressed in bits per seconds), B is the channel bandwidth in Hz, S is the average power of the transmitted signal within the bandwidth B of the channel (expressed in Watts), and N is the average power of the white Gaussian noise within the bandwidth B of the channel (expressed in Watts).

This expression is also known as *Shannon–Hartley theorem*, or sometimes called *Shannon-Hartley's law*. It may be noted that while the channel capacity is linearly related to the bandwidth B, it is logarithmically related to the signal-to-noise power ratio (S/N). This means that it is easier to increase the capacity of a given communication channel by increasing its bandwidth, rather than by increasing the transmitted power.

IMPORTANT! The Shannon expression provides the relationship between the theoretical ability of a communication channel to transmit information without errors for a given signal-to-noise ratio and a given bandwidth of the channel. Shannon modeled the communication channel at baseband. However, this expression is also applicable to a RF channel based on AWGN model. It is assumed that the intermediate frequency (IF) filter has an ideal (flat) bandpass frequency response over at least $2B$ bandwidth.

Shannon's Third Theorem

Combination of Shannon's channel coding theorem and Shannon-Hartley theorem is known as *Shannon's third theorem*. we find that for given average signal power S and channel bandwidth B, we can transmit information over the Gaussian channel at the rate of C bits per second as given by $C = B\log_2(1 + S/N)$ with arbitrarily small probability of error (by employing sufficiently complex encoding scheme) and it is not possible to transmit at a rate greater than C.

Significance

Clearly, this theorem sets a fundamental limit on the data rate (bits per seconds) at which error-free transmission can be achieved for a power-limited, band-limited Gaussian channel. This limit can be approached only if the transmitted signal too has statistical properties approximating those of white Gaussian noise.

Shannon's Bound

Assume Gaussian channel that is limited in both power and bandwidth. We know that the channel capacity or information capacity C of a channel having its bandwidth B Hz, influenced by additive white Gaussian noise (AWGN) of two-sided power spectral density $N_0/2$ and bandlimited to B Hz is given by

$$C = B\log_2\left(1+\frac{P}{N_0B}\right)$$

where P is the average transmitted power, which can be expressed in terms of transmitted energy per bit E_b as

$$P = E_bC$$

Therefore, $$C = B\log_2\left(1+\frac{E_bC}{N_0B}\right)$$

$\Rightarrow$ $$\frac{C}{B} = \log_2\left(1+\frac{E_bC}{N_0B}\right)$$

Taking antilog on both sides, we have

$\Rightarrow$ $$2^{C/B} = 1+\frac{E_bC}{N_0B}$$

Re-arranging the terms, we get

$\Rightarrow$ $$\frac{E_bC}{N_0B} = 2^{C/B} - 1$$

$\Rightarrow$ $$\boxed{\frac{E_b}{N_0} = \frac{2^{C/B}-1}{C/B}}$$

This expression defines the signal-per-bit to noise power spectral density ratio E_b/N_0 in terms of the bandwidth efficiency C/B (bps/Hz) for an ideal system, which is defined as the system that transmits data at a bit rate R equal to the channel capacity C.

That is, for an ideal system, $C = R$

Hence, $$\boxed{\frac{E_b}{N_0} = \frac{2^{R/B}-1}{R/B}}$$

For an infinite bandwidth, the ratio E_b/N_0 approaches the limiting value which can be determined as

$$\left(\frac{E_b}{N_0}\right)_\infty = \lim_{B\to\infty}\left(\frac{E_b}{N_0}\right) = \log 2 = 0.693, \text{ or } -1.6 \text{ dB}$$

Thus, the minimum value for E_b/N_0 is –1.6 dB and is called the *Shannon limit* for an AWGN channel. This means that if ideal (unknown) channel coding and decoding is employed at the respective transmitter and receiver of a digital communication system, then error-free data will be recovered at the receiver output, provided that the E_b/N_0 at the receiver input is more than –1.6 dB. Any practical system will perform worse than this ideal system described by Shannon's limit. Therefore, the goal of system designer is to find practical channel codes that approach the performance of Shannon's ideal (unknown) code.

The corresponding limiting value of the channel capacity will be

$$C_\infty = \lim_{B\to\infty} C = \lim_{B\to\infty}\left\{B\log_2\left(1+\frac{P}{N_0B}\right)\right\} = \frac{P}{N_0}\log_2 e$$

It also implies that there may be certain system parameters which may ensure error-free transmission ($R < C$), and $R = C$ is the *critical bit rate*. Moreover, there exists a potential trade-off among three key system performance parameters, namely E_b/N_0, R/B, and the probability of error P_e.

Thus, the Shannon–Hartley theorem relates to the power-bandwidth trade-off and results in the Shannon limit of –1.6 dB of theoretical minimum value of E_b/N_0, that is necessary to achieve an arbitrary low error probability over an AWGN channel. In fact, channel capacity imposes a limit on error-free transmission. The upper limit for reliable information transmission rate in a bandlimited AWGN channel is given by

$$R \leq B \log_2 (1 + S/N) \text{ bits per second}$$

Theoretically, the maximum transmission rate possible is the channel capacity in accordance with the Nyquist rate samples.

A Real-Life Practical Case

Practically, Shannon–Hartley's Law can be applied for transmission of a sound or video signal whose bandwidth exceeds the channel capacity by taking more time than normal. However, this is not allowed in real-time communication as in a telephone conversation. For example,

- The transmission speed achieved by the ITU V.34 standard QAM modem approaches the Shannon-limit for a dial-up telephone line.
- Depending on the S/N ratio, the line may not accommodate the maximum data rate; in this case, the modem automatically slows down to accommodate the line.
- To communicate at data rates near the Shannon limit, modems must apply equalization to the line to compensate for its uneven frequency and phase response.
- In fact, the ITU V.90 standard modems operate at a maximum data rate of 33.6 kbps and do not violate the Shannon limit.

Information Capacity

In general, *information* is referred as the knowledge or intelligence that is communicated between two or more points. As mentioned earlier, the study of *information theory* is highly theoretical about the efficient use of channel bandwidth to transmit information from source to destination through electronic communications systems. Information theory can be used to determine the information capacity of a digital communication system. *Information capacity* is a measure of the amount of information data that can be transmitted through a communication system. In fact, it represents the number of independent digital symbols (expressed in bits) that can be carried through a system in a given unit of time.

Hartley's Law

R Hartley of Bell Telephone Labs developed a useful relationship among information capacity, bandwidth, and transmission time. As per *Hartley's Law*, the information capacity in bits per second is given as

$$I \propto B \times t$$

where B is the channel bandwidth in Hz, and t is transmission time in seconds. Thus, information capacity is a linear function of channel bandwidth and transmission time. If either the channel bandwidth or transmission time changes, the information capacity too changes in the same proportion. It is often convenient to express the information capacity of a digital communication system as a *bit rate*.

As discussed earlier, if the channel bandwidth is doubled, the information capacity of a communications channel is also doubled.

As per Hartley's law, there is no theoretical limit to the amount of information that can be transmitted in a given channel bandwidth provided there is sufficient time to send it. That is, a large amount of information can be transmitted in a small bandwidth by using more time. Thus, we can say that the information data rate and the channel bandwidth are theoretically interchangeable.

Channel Capacity for Multi-level Signaling

So far we have discussed digital transmission using binary (two-level i.e. 1 and 0) encoding. It is often advantageous to encode at a level higher than binary, known as *M-ary*, where M simply represents a digit that corresponds to the number of levels possible for a given number of binary variables. For example,

- A digital signal with four possible levels or conditions (in terms of voltage levels, frequencies or phases) is an M-ary system where $M = 4$.
- Similarly, if there are eight possible levels, $M = 8$ and in case of 16 possible levels, $M = 16$ and so on.

In general, we can express the number of bits necessary to produce a given number of levels as

$$n = \log_2 M$$

This expression can be rearranged as $2^n = M$.

For example, with two number of bits, $2^2 = 4$ levels are possible, with three bits, $2^3 = 8$ levels are possible, and so on. Binary is a case of $n = 1$, $2^1 = 2$ levels are possible.

In a *noise-free communication channel*, with *multilevel signaling* the *channel capacity, C* (expressed in bits per seconds) as per *Nyquist criterion* is given as

$$C = 2B \log_2 M; \text{ for } M > 2$$

where B is the channel bandwidth in Hz, and M is the number of discrete levels.

Trade-Off between Bandwidth and S/N

It is quite obvious that channel bandwidth and average transmitted power are two primary communication resources in any communication systems. Recall that each type of communication channel has its own characteristics including frequency or range of frequencies, called its *bandwidth*. We know that the frequency spectrum of a transmitted signal fully describes the signal in the frequency domain, containing all its components. It is desirable that a communication channel must pass every frequency component of the transmitted signal, while preserving its amplitude and phase. But no communication channel or transmission medium is perfect! In fact, each transmitting medium passes some frequencies, weakens others, and may block still others, thereby producing signal distortions.

Similarly, the *transmitted power* determines the signal-to-noise power ratio (*S/N*) at the receiver input. This, in turn, determines the allowable distance (operating range) between the transmitter and receiver for an acceptable signal quality. The *S/N* at the receiver determines the noise performance of the receiver which must exceed a specified level for satisfactory communication over the channel.

So, we must utilize these two communication resources – *channel bandwidth* and *average transmitted power* efficiently.

Transmitted signal is distorted in propagating through a channel because of noise, interference, distortion due to nonlinearities and imperfections in the frequency response of the channel. However, it is desirable that the received signal should be the exact replica of the transmitted signal.

Transmitted Signal Spectrum Different from Channel Bandwidth

Communication channels may be classified as band-limited or power-limited channel. For example, the telephone communication link is a band-limited channel whereas a satellite communication link is typically power-limited channel. If the channel bandwidth does not match with the frequency spectrum of the transmitted signal, some of the frequency components of that signals may be highly attenuated and not received.

Example

What happens when the spectrum of the transmitted signal is different from that of communication channel?

To understand it, let us consider an example. A transmitting medium has a bandwidth of 1 kHz and can pass frequencies from 3 kHz to 4 kHz. Can a signal having bandwidth of 1 kHz in a spectrum with frequencies from between 1 kHz and 2 kHz be passed through this medium?

The answer is **No!**

The signal will be totally attenuated and lost. Although the bandwidth of the signal matches with that of the transmitting medium but its spectrum does not match.

Therefore, it is essential that the bandwidth as well as the frequency spectrum of the communication channel must match with that of the signal to be transmitted. Note the following:

- Analog transmission can use a bandpass channel but digital transmission needs a low-pass channel (theoretically between zero and infinity).
- A bandpass channel is generally available as compared to a low-pass channel because the bandwidth of a transmitting medium can be divided into several bandpass channels to carry multiple transmissions.

Constraint on Transmission Data Rate

In a *noise-free communication channel* (ideal channel), the limitation on data rate of the transmitted signal is mainly constrained by the available bandwidth of the communication channel. That means the bandwidth of the communication channel must be large enough to pass all the significant frequencies of the information data and transmitted signal. We can say that a communication channel cannot propagate a signal that contains a frequency which is changing at a rate greater than the bandwidth of the channel. Thus, the usable bandwidth of the channel is one of the important parameters which determine the channel capacity and signal-to-noise ratio.

Recall that the *channel capacity* (also known as information-carrying capacity) refers to the maximum data transmission rate at which the digital transmission is theoretically possible without any errors. For a given channel bandwidth, it is desirable to get as high data rate as possible at a specified limit of error rate.

- For binary information, the data rate that can be supported by channel bandwidth of B Hz is $2B$ bits per second (bps).
- Alternatively, the highest data rate that can be supported is $2B$ bps in a given bandwidth of B Hz.

So, the greater the available bandwidth, the higher is the channel capacity.

A Practical Case

In a practical communication system, the channel is not noise free. We have seen that according to Claude E. Shannon, the *channel capacity* (C) of a communication channel is related to channel bandwidth (B Hz) and signal-to-noise power ratio (S/N) by the expression

$$C = B \log_2 (1 + S/N)$$

From the Shannon's limit for channel capacity, it can be interpreted that for a given level of noise, the data rate or the channel capacity can be increased by increasing either bandwidth or the signal strength. However, we have to remember that in case the bandwidth B is increased, more noise is introduced to the system since the noise is assumed to be white noise (i.e. $N = kTB$). Thus, as B increases, S/N decreases. As the signal strength (S) is increased, the noise also increases due to non-linearity effects of the system.

Bandwidth versus S/N

The Shannon–Hartley theorem indicates that we may trade-off bandwidth for S/N ratio and vice versa. For example, for $B = 4$ kHz and $S/N = 7$ (assume), we have

$$C = 4 \text{ kHz} \times \log_2 (1 + 7) = 4 \text{ kHz} \times \log_2 2^3 = 12 \text{ kbps}$$

Now, if bandwidth is reduced by 25% (i.e., $B = 3$ kHz) and S/N is increased by more than 100% (i.e., $S/N = 15$, say), then

$$C = 3 \text{ kHz} \times \log_2 (1 + 15) = 3 \text{ kHz} \times \log_2 2^4 = 12 \text{ kbps} \qquad \text{(same as before)}$$

It may be noted that when B decreases from 4 kHz to 3 kHz, the noise power also decreases by 3/4 times. Thus, the signal power will have to be increased by the factor (3/4 × 15/7 =) 1.6, i.e. 60%. This implies that the 25% reduction in channel bandwidth requires a 60% increase in signal power for the same channel capacity. So there is a trade-off between bandwidth and S/N ratio and is not limited by a lower limit on channel bandwidth.

Interpretation of Shannon–Hartley Theorem

As per *Shannon–Hartley theorem*, the channel capacity of a bandlimited additive white Gaussian noise channel is given by

$$C = B \log_2 (1 + S/N) \text{ bits per second}$$

where B is the channel bandwidth in Hz, S is the signal power, and N is the total noise power within the channel bandwidth, i.e., $N = N_0 B$, with $N_0/2$ as two-sided noise power spectral density.

The following points may be observed:

- As the channel bandwidth B increases, the channel capacity C does increase.
- But the channel capacity does not become infinite as the bandwidth becomes infinite because with an increase in bandwidth, the noise power N also increases.
- Thus, for a fixed signal power S and in the presence of white Gaussian noise, the channel capacity approaches an upper limit with increasing bandwidth.
- Using $N = N_0 B$, we can write

$$C = B \log_2 \left(1 + \frac{S}{N_0 B}\right)$$

Multiplying and dividing the right-hand term with S/N_0, we have

$$\Rightarrow \qquad C = \frac{S}{N_0} \times \frac{N_0}{S} \times B \log_2 \left(1 + \frac{S}{N_0 B}\right)$$

$$\Rightarrow \qquad C = \frac{S}{N_0} \times \log_2 \left(1 + \frac{S}{N_0 B}\right)^{\frac{N_0 B}{S}}$$

Let us denote $\frac{S}{N_0 B} = x$, or $\frac{N_0 B}{S} = \frac{1}{x}$. And as $B \to \infty$, $x \to \infty$. Then,

$$\Rightarrow \qquad C = \frac{S}{N_0} \times \log_2 (1 + x)^{1/x}$$

We know that lim $x \to \infty$, $(1 + x)^{1/x} = e$ (the Naperian base). Let lim $B \to \infty$, $C = C_\infty$

$$\therefore \qquad C_\infty = \frac{S}{N_0} \times \log_2 e = \frac{S}{N_0} \times 3.32 \log_{10} e$$

Hence, $$\boxed{C_\infty = 1.44 \frac{S}{N_0}}$$

Error-free Transmission using Orthogonal Signals

The Shannon–Hartley theorem also indicates that a noise-free or noiseless ideal Gaussian channel (i.e., $N = 0$ means $S/N = \infty$) has an infinite channel capacity C. It implies that even when the channel bandwidth may be restricted to low, it is always possible to receive a signal without any error in a noise-free channel, as predicted by the Shannon limit.

- In the presence of Gaussian noise, there is a possibility to attain the performance predicted by Shannon–Hartley theorem.
- In such case a set of orthogonal signals has to be considered over the interval 0 to T such that

$$\int_0^T s_i(t) s_j(t) dt = 0 \qquad \text{for } i \neq j$$

- We know that for M-ary signals, the rate of information R for all M signals having equal probability of occurrence, the probability of error P_e decreases as S/N_0 increases.
- Also, we note that as S/N_0 approaches to infinity, P_e approaches to zero.
- As M approaches to infinity,

$$P_e = \begin{cases} 0 & \text{provided } \dfrac{S}{N_0 R} \geq \ln 2 \\ 1 & \text{otherwise} \end{cases}$$

- Thus, we observe that for fixed values of M and R, P_e decreases as the signal power S goes on increasing, or the noise power spectral density N_0 goes on decreasing.
- Similarly, for fixed values of S, N_0 and M, the P_e decreases as we allow more time (i.e., decreasing R) for the transmission of a single message.
- The required bandwidth is now infinite. From this condition, we may compute the maximum allowable error-free transmission rate R_{max}, which in fact, is the channel capacity C. Therefore,

$$R_{max} = C = \frac{S}{N_0} \frac{1}{\ln 2} = \frac{S}{N_0} \log_2 e = \frac{S}{N_0} 3.32 \log_{10} e$$

Hence, $$\boxed{C = 1.44 \frac{S}{N_0}}$$

This result is similar to the result obtained earlier which we deduced from the Shannon-Hartley theorem for a channel of limited bandwidth.

When M approaches to infinity, to reduce P_e to zero, we find that we must wait for an infinite time before we receive any information!

A Practical Example

When we calculate the performance limits of channel coding using the Shannon's channel capacity formula, the coding is a part of the optimum modulation process. For example, for a CDMA based direct-sequence (DS) cellular system each user's data rate is much less than the total spread spectrum bandwidth. Theoretically, the multiple users from a single cell have an error-free reception in an AWGN environment, as long as the total transmission data rate (which is summed over the individual rates of all users) does not exceed Shannon capacity. The Shannon capacity is based on a given summed power of all users and at an interference level within the bandwidth. The Shannon channel capacity can be treated as an upper limit for the mobile radio channels including the Rayleigh fading channel. However, the channel capacity in a Rayleigh fading environment is always lower than in a Gaussian noise environment.

Conclusion

Hence, we have discussed one possible way of achieving error-free transmission through the use of orthogonal signals (and matched filters) over a channel of unlimited bandwidth. We have also computed the channel capacity and arrived at a result which is consistent with the Shannon–Hartley theorem.[9]

SOLVED EXAMPLE 4.4.2 Maximum Channel Capacity

Determine the channel capacity for a communications channel with a bandwidth of a 50 kHz and a signal-to-noise ratio of 40 dB.

Solution Given $S/N = 40$ dB, or 10,000 $[\because S/N|_{dB} = 10\log_{10} S/N|_{ratio}]$

The channel capacity for specified bandwidth B is given as

$$C = B\log_2(1 + S/N) = 3.32B\log_{10}(1 + S/N)$$

$$C = 3.32 \times 50 \text{ kHz} \times \log_{10}(1 + 10000) = 664 \text{ kbps}$$ **Ans.**

SOLVED EXAMPLE 4.4.3 Transmission Bandwidth

A modulation scheme requires 10 kHz bandwidth for the telephone voice transmission of frequencies up to about 3.4 kHz. How much bandwidth will be needed to transmit video broadcast signals of frequencies up to 5 MHz using the same modulation scheme?

Solution

Baseband audio signal bandwidth = 3.4 kHz (Given)

Baseband video signal bandwidth = 5 MHz (Given)

Transmission audio signal bandwidth = 10 kHz (Given)

To calculate transmission video signal bandwidth:

Using Hartley's law, which states that information rate (which is equivalent to baseband bandwidth) is directly proportional to transmission bandwidth or vice versa, we get

Transmission video signal bandwidth = (10 kHz/3.4 kHz) $\times$ 5 MHz = 14.7 MHz **Ans.**

SOLVED EXAMPLE 4.4.4 Nyquist Channel Capacity

Consider the case of a noise-free communication channel. Compute the maximum channel capacity of the channel having bandwidth of 3100 Hz for (a) binary rate; (b) each signal element representing 2 bit data; and (c) each signal element representing 3 bit data. Also interpret the results.

Solution The channel bandwidth, $B = 3100$ Hz (Given)

[9] Remarkably, Shannon showed that the channel capacity could be achieved with bit error probability approaching zero by a completely random channel codes (i.e., a randomly chosen mapping set of code words) only when the block length approaches infinity. However, random channel codes are not practically feasible. Recently, turbo codes have become popular because they can perform near Shannon's limit, yet they also can have reasonable decoding complexity.

We know that in case of noise-free communication channel, Nyquist criterion states that the maximum channel capacity is given by

$$C = 2B \log_2 M$$

where M is the number of discrete signal elements or levels.

(a) For binary data ($n = 1$), $M = 2^1 = 2$ [Using $M = 2^n$]
Therefore, $C = 2 \times 3100$ Hz $\times \log_2 (2)$ bps = **6200 bps** **Ans.**

(b) For each signal element representing 2-bit data, number of possible levels of signaling, $M = 2^2 = 4$
Therefore, $C = 2 \times 3100$ Hz $\times \log_2(4)$ bps = **12,400 bps** **Ans.**

(c) For each signal element representing 3 bit data, number of possible levels of signaling, $M = 2^3 = 8$
Therefore, $C = 2 \times 3100$ Hz $\times \log_2(8)$ bps = **18,600 bps** **Ans.**

Interpretation of the Results: From the above results, it is observed that for a given bandwidth, the channel capacity (transmission data rate) can be increased by increasing the number of different signal elements. However, interference and noise present in the communication channel will limit the practical limit of M. Moreover, the design of the receiver becomes complex as it has now to distinguish one of M possible signal elements.

SOLVED EXAMPLE 4.4.5 Determination of Signaling Levels

It is desired that the information data rate of 30.9 kbps need to be propagated through a 3.1 kHz band communications channel. Determine the optimum number of signaling elements per bit to be transmitted to achieve it.

Solution Information data rate, $C = 30.9$ kbps (Given)
Channel bandwidth, $B = 3.1$ kHz (Given)
In case of noise-free communication channel, as per Nyquist criterion, the information data rate (channel capacity) is given by

$$C = 2B \log_2 M$$

where M is the number of discrete signal elements.

$$30.9 \text{ kbps} = 2 \times 3.1 \text{ kHz} \times \log_2 M$$

$$30.9 \text{ kbps} = 2 \times 3.1 \text{ kHz} \times 3.32 \times \log_{10} M \quad (\text{Using } \log_2 M = 3.32 \times \log_{10} M)$$

or, $M \approx 32$

Hence, the optimum number of signaling elements per bit, M = **32** **Ans.**

SOLVED EXAMPLE 4.4.6 Shannon Limit for Information Capacity

Compute the Shannon limit for information capacity for a standard telephone channel having 2.7 kHz bandwidth and signal-to-noise power ratio of 30 dB. Comment on the result obtained.

Solution We know that Shannon's limit for information capacity is given as

$$C = B \log_2 \left(1 + \frac{S}{N}\right)$$

Converting given S/N of 30 dB to ratio, we have

$$30 \text{ dB} = 10 \log(S/N); \Rightarrow S/N = 10^{30/10} = 1000$$

Therefore, $C = (2.7 \text{ kHz}) \times \log_2(1 + 1000)$

$$\Rightarrow \quad C = (2.7 \text{ kHz}) \times 3.32 \times \log_{10}(1001) = 26.9 \text{ kpbs} \qquad \textbf{Ans.}$$

Comment on the Result: This result indicates that information can be transmitted through a 2.7 kHz communications channel at a data rate of 26.9 kbps, as per Shannon's limit for information capacity. This is not possible with a binary system. **Quite often Shannon's law is misunderstood**. The above results can be achieved by transmitting each symbol with more than one bit.

SOLVED EXAMPLE 4.4.7 Maximum Channel Capacity

A typical voice communications channel has a bandwidth of 3.1 kHz (300 Hz – 3400 Hz) and $S/N = 30$ dB. Calculate the maximum channel capacity.

Solution Given channel bandwidth, $B = 3.1$ kHz

Given $S/N = 30$ dB, or S/N in ratio = 1000

We know that the maximum channel capacity is given by Shannon's channel capacity, that is,

$$C = B \log_2 (1 + S/N), \text{ or } C = 3.32 \times B \log_{10}(1 + S/N)$$

or,

$$C = 3.32 \times 3.1 \text{ kHz} \times \log_{10}(1 + 1000) = \mathbf{30.9\ kbps} \qquad \textbf{Ans.}$$

It implies that the maximum channel capacity or information data rate of 30.9 kbps can be propagated through a 3.1 kHz communications channel. But this can be achieved only when each transmitted symbol contains more than one bit. Thus, it is possible to design signaling schemes that exchange bandwidth for S/N, as noise is inevitable in any electronic communication system.

SOLVED EXAMPLE 4.4.8 Channel Bandwidth versus Channel Capacity

A standard 4 kHz telephone channel has $S/N = 25$ dB at the receiver input. Calculate its channel capacity. If channel bandwidth is doubled, how much the channel capacity increases, assuming that the transmitted signal power remains constant.

Solution We know that channel capacity, $C = 3.32 \times B \log_{10}(1 + S/N)$

For given $B = 4$ kHz, $S/N = 25$ dB;

$$S/N = \text{antilog}\left(\frac{S/N}{10}\text{dB}\right) = \text{antilog}\left(\frac{25}{10}\right) = 316$$

[10]It may be recalled that S/N of 316 means that when signal power is 316 mW (say), the noise power is 1 mW.

Therefore,

$$C = 3.32 \times 4 \text{ kHz} \times \log_{10}(1 + 316) = 33.2 \text{ kbps}$$

Now when the channel bandwidth is doubled, i.e., $B = 8$ kHz. We know that with constant signal power, the noise power is also doubled due to bandwidth being doubled.

$$\therefore \quad C = 3.32 \times 8 \text{ kHz} \times \log_{10}\left(1 + \frac{316}{2}\right) = 58.5 \text{ kbps}$$

[10]Recently, one of the important breakthroughs in wireless digital communications is the advent of multiple-input-multiple-output (MIMO) technologies. In fact, the development of MIMO originates from the fundamentals of information theory. The key advantage of MIMO wireless communication systems lies in their ability to significantly increase wireless channel capacity without either requiring additional bandwidth or substantially increasing the signal power at the transmitter. Both the Wi-Fi (IEEE 802.11n) and the WiMAX (IEEE 802.16e) standards have incorporated MIMO transmitters and receivers.

Discussion: Indeed, the channel capacity has increased from 33.2 kbps to 58.5 kbps. It may be observed that channel capacity does not increase by two times with doubling of bandwidth because noise also increases with increase in channel bandwidth (although signal power remains constant). In other words, for doubling the channel capacity with doubling of bandwidth, signal power also has to be increased by two times.

Self-Assessment Exercise linked to LO 4.4

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/137

Q4.4.1 Find signal-to-noise ratio in dB required to achieve an intended capacity of 20 Mbps through a communications channel of 3 MHz bandwidth. ○○●

Q4.4.2 If a signal element encodes a 8-bit word, then determine the minimum required channel bandwidth used for a digital signaling system operating at 9600 bps. ○●●

Q4.4.3 The frequency spectrum of a communications system is allocated as 24 MHz – 25 MHz. If the channel data rate is specified as 8000 kbps, then how many signaling levels are required to achieve this data rate? ●●●

Q4.4.4 Determine the number of signaling levels and number of bits per symbol needed to achieve theoretical maximum channel data rate of 26.9 kbps for given channel bandwidth of 2.7 kHz. ○●●

Q4.4.5 The channel data rate for GSM cellular system is specified as 270.833 kbps. The carrier channel bandwidth is 200 kHz. If the S/N of a wireless communications channel is 10 dB, show that the maximum achievable channel data rate is adequate for GSM specification. ●●●

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 4.4: *Analyze the intrinsic ability of the communication channel to convey information reliably, known as channel capacity, focus on selected topics such as Shannon's channel coding theorem, Shannon–Hartley theorem, and trade-off between bandwidth and S/N.*

Key Concepts

- alphabets
- average effective entropy
- average information
- AWGN
- binary erasure channel
- binary source
- binary symmetric channel
- channel capacity
- channel coding
- conditional entropy
- channel matrix
- code efficiency
- differential entropy
- discrete memoryless source
- entropy
- Hartley theorem
- information rate
- joint entropy
- Markov source
- *M*-ary source
- mutual information
- nat
- Shannon's theorem
- symbol probability

Learning Outcomes

- Information theory is a branch of probability theory, which can be applied to the study of communication systems.
- Information theory deals with the measure for an amount of information and the means to apply it to improve the communication of information.
- The probability of an event and the amount of information associated with it are inversely related to each other.

LO 4.1

- The source may be described in terms of average amount of information per individual message, known as entropy of the source.
- The entropy of a source is a function of the probabilities of the source symbols that constitute the alphabet of the source.
- Since the entropy is a measure of uncertainty, the entropy is maximum when the associated probability distribution generates maximum uncertainty.
- It is true that average information is maximum when all the messages are equally likely.

LO 4.2

- A binary symmetric channel is the simplest form of a discrete memoryless channel.
- A binary symmetric channel is symmetric because the probability of receiving a binary logic 1 if a 0 is sent is exactly the same as the probability of receiving a binary logic 0 if a 1 is sent.

LO 4.3

- The channel capacity represents the maximum rate at which data can be transferred between transmitter and receiver, with an arbitrarily small probability of error.
- When the system operates at a rate greater than *C*, it is liable to incur high probability of error.

LO 4.4

Objective-Type Questions

For Interactive Quiz with answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/82

4.1 Identify a discrete source that generates statistically independent symbols.
(a) Memoryless
(b) With memory
(c) Maximally matched
(d) Optimally matched

4.2 A discrete source with memory is known to be
(a) consisting of short sequence of symbols
(b) consisting of long sequence of symbols
(c) Markov source
(d) stochastic source

4.3 The maximum entropy of a binary source is computed to be ________ per symbol.
(a) 0.5 bit
(b) 1 bit
(c) 1.5 bit
(d) 2 bits

4.4 Check that the maximum entropy of a binary source occurs when one of the following condition is satisfied.

(a) $p(0) = p(1) = 0$
(b) $p(0) = p(1) = 0.5$
(c) $p(0) = p(1) = 1$
(d) $p(0) = 0.4$ and $p(1) = 0.6$

4.5 *Statement I:* An M-ary discrete memoryless source produces all the symbols with equal probabilities. The maximum entropy of the source is given by $\log_2 M$ bits per symbol.
Statement II: It is not possible to find any uniquely decodable code whose average length is less than the entropy of the source expressed in bits.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

4.6 Exemplify that a given discrete memoryless source will have maximum entropy provided the messages generated are
(a) statistically independent
(b) statistically dependent
(c) equiprobable
(d) binary

4.7 The ________________ decides the maximum permissible rate at which an error-free transmission of information is possible through the communication channel.
(a) source entropy
(b) channel capacity
(c) source coding
(d) channel coding

4.8 *Statement I:* The rate of transmission of information over a channel is maximum if the symbol probabilities are unequal.
Statement II: Channel capacity is the property of a particular physical channel over which the information is transmitted.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

4.9 The information rate for an analog signal band-limited to B Hz and having entropy of 1.8 bits/message is
(a) 0.9B bps
(b) 1.8B bps
(c) 3.6B bps
(d) B bps

4.10 A discrete source generates one of the five possible messages during each message interval with respective probabilities of these messages as $p_1 = 0.5$, $p_2 = 0.0625$, $p_3 = 0.125$, $p_4 = 0.25$, and $p_5 = 0.0625$. The information content in third message is calculated to be
(a) 1 bit
(b) 2 bits
(c) 3 bits
(d) 4 bits

Short-Answer-Type Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/47

4.1 Define entropy. State the condition for the entropy of the source to attain the maximum value.

4.2 Bring out clearly similarities and dissimilarities between the terms Information and Entropy.

4.3 How can the amount of information be measured which has been gained after the occurrence of the event? Under what conditions is the information gained zero?

4.4 Give an account of the various properties of information.

4.5 The entropy of a source is a function of the message probabilities. Can we say that entropy is maximum when all the messages are equiprobable?

4.6 Outline the upper and lower bounds on entropy.

4.7 Discriminate between conditional entropy and joint entropy.

4.8 Determine the entropy of a binary memoryless source when the symbols 0 and 1 occur with equal probability.

4.9 Why does downloading an audio or video stored file from the Internet sometimes take much longer time than it requires playing? Use Shannon–Hartley theorem to justify your answer.

4.10 What is the practical difficulty in achieving transmission rate equal to theoretical capacity of the channel?

4.11 Specify source information rate for equiprobable symbols. For non-equiprobable symbols, how is the source information rate related to that of equiprobable symbols?

4.12 Why is white Gaussian noise considered the worst possible noise in terms of interference associated with signal transmission?

4.13 Why do we need channel coding? State Shannon's channel-coding theorem.

4.14 How does increase in data rate affect the quality of a received signal?

4.15 The Shannon–Hartley theorem shows that theoretically information rate and bandwidth are interchangeable. Give an example of the application of the Shannon–Hartley theorem.

Discussion Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/139

4.1 The digital pulse waveforms are propagated between source and destination through various levels of coding as required by different system applications. Elucidate the functional block schematic of a typical digital communication system for reliable transmission of analog information. **[LO 4.1]**

4.2 Consider a binary source for which the symbol 0 occurs with probability p_0 and the symbol 1 occurs with probability $p_1 = (1 - p_0)$. Show that the entropy is zero when symbol 0 occurs with either probability $p_0 = 0$, or $p_0 = 1$. Under what conditions the entropy would be unity? **[LO 4.2]**

4.3 "The entropy of the extended discrete memoryless source is equal to n times H, the entropy of the original discrete memoryless source". Justify this statement with the help of appropriate data. **[LO 4.2]**

4.4 A communication channel is said to be a memoryless channel when the current output symbol depends only on the current input symbols and not any of the previous input symbols. How can a discrete memoryless channel be described conveniently? Describe its fundamental properties. **[LO 4.3]**

4.5 The study of the characteristics of the Gaussian channel is very much essential because these channels are often encountered in practice. Draw a simplified model of a discrete-time memoryless Gaussian channel and derive an expression for channel capacity in terms of bandwidth, transmitted power, and additive white Gaussian noise power spectral density. **[LO 4.4]**

4.6 Assuming that the Gaussian channel is limited in both transmitted power and bandwidth, show that the Shannon upper limit for an AWGN channel is –1.6 dB for an ideal system in which the transmitted bit rate is equal to the channel capacity. **[LO 4.4]**

Problems

For answer keys, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/141

4.1 Consider a binary source with probability $P(0) = 0.6$ and $P(1) = 0.4$. Determine the entropy of the binary source.

4.2 Calculate the entropy rate of the source if the entropy is given as 1.75 bits per symbol and the source produces 100 symbols per second.

4.3 Clarify that the entropy of the system in which an event has six possible outcomes with their respective probabilities $p_1 = 0.5$, $p_2 = 0.25$, $p_3 = 0.125$, $p_4 = 0.0625$, $p_5 = 0.03125$, and $p_6 = 0.03125$ is 1.9375 bits per message.

4.4 Consider a signal which is sampled at the rate of 500 samples per second, and then quantized with the probability distribution of the source alphabet as 0.02275, 0.13595, 0.3413, 0.3413, 0.13595, and 0.02275. Estimate the source entropy and the rate of information.

4.5 An analog signal, bandlimited to 4 kHz and sampled at Nyquist rate, is quantized into 4 independent levels which occur with probabilities $p_1 = p_4 = 0.125$ and $p_2 = p_3 = 0.375$. Determine the average information per level and the information rate of the source.

4.6 Consider the binary source having the state transition probabilities $P(0/1)$ and $P(1/0)$ as 0.45 and 0.05 respectively. Hypothesize the *apriori* probability of each state and the entropy of the binary source.

4.7 A telephone line has a bandwidth of 3.1 kHz and requires a signal-to-noise power ratio of 35 dB for transmitting a digital data using a four-level code. What is the maximum theoretical data rate?

4.8 The redundancy in a simple source encoding with two binits is specified as 0.25 bits per symbol. If a source produces 10^6 symbols per second, calculate the number of digits that are redundant every second.

4.9 Compute the minimum required bandwidth of the communications channel for a digital transmission system required to operate at 9600 bps. Assume that each signal element encodes a 4-bit word.

4.10 Determine the Shannon limit for channel capacity for a standard telephone circuit with a specified bandwidth of 2.7 kHz and a signal-to-noise power ratio of 30 dB.

Critical Thinking Questions

4.1 Consider the event which describes the generation of symbol s_k by the discrete memoryless source with probability p_k. It is required to measure the extent of uncertainty in the information so as to know the amount of information contents contained in the signal. State the three distinct conditions that may exist and analyze how the amount of information is related to the probability of occurrence. **[LO 4.1]**

4.2 The entropy function of a binary memoryless source varies with symbol probability. Illustrate the variation of entropy function when during the value of p_0 from 0 to 1/2, and then from 1/2 to 1 and show that it has maximum value of unity only when binary symbols are equally probable. **[LO 4.2]**

4.3 The difference between entropy and conditional entropy must represent uncertainty about the channel input that is resolved by observing the channel output which is nothing but mutual information of the channel. State symmetrical and non-negative property of mutual information and discuss their corresponding significance. **[LO 4.3]**

4.4 Shannon's channel-coding theorem is fundamental to the information theory of communications. For a binary symmetric channel, show that for a channel code to exist, $R \leq C$ where R is the information rate, and C is the channel capacity, which is capable of achieving an arbitrary low probability of error. **[LO 4.4]**

References for Further Reading

[Ash07] Robert, Ash B; *Information Theory.* Dover Special Priced Titles, 2007.
[SI12] Singal, TL; *Analog and Digital Communications.* Tata McGraw-Hill, 2012.
[Wel08] Wells Richard B; *Applied Coding and Information Theory for Engineers*. Dorling Kindersley India, 2008.

5 Coding

Learning Objectives

To master coding, you must cover these milestones:

- **LO 5.1** Understand the basics of source encoding, its classifications and source-coding theorem
- **LO 5.2** Discuss source coding techniques including Shannon–Fano, Huffman, and Lempel–Ziv
- **LO 5.3** Implement error-control channel–coding techniques such as Hamming codes, cyclic codes, BCH code, Hadamard code, and LDPC code
- **LO 5.4** Illustrate the concept of convolution codes, Viterbi and sequential decoding algorithms
- **LO 5.5** Outline burst-error correction codes such as Interleaving, RS codes, and turbo codes

Essence of Source and Channel Coding

When digital data is transmitted or stored, we cannot ignore coding or encoding. The field of mathematics that deals with sending data (a digital bit stream) over a noisy channel is called coding theory. For error-free transmission over a noisy communication channel, both source and channel coding are essential in modern digital communication systems. The source output is transformed into a sequence of binary digits called the information sequence. Source encoding deals with efficient representation of symbols which leads to compression of data such that the average bit rate is minimized by reducing the redundancy of the source information. The source output can be reconstructed from the information sequence without any ambiguity. Source coding, in general, increases the data rate at which information may be transmitted over a communications channel while maintaining the acceptable information bit error rate. Thus, a digital communication system is designed to achieve desired information data rate as well as acceptable error rate within available channel bandwidth. To achieve reasonably low error rate during transmission through a noisy channel, error-control coding (channel coding) is applied by adding some redundancies in the source encoded signals. Channel coding enhances the capability of transmitted digital data to detect and correct errors at the receiver.

INTRODUCTION

Recap

A digital communication system requires transmission of digital data in communications channels. The data could be true digital data as in a computer network or analog information which has been converted to a digital format as in analog-to-digital conversion such as PCM. A typical transmission model of digital data involves several components as depicted in Figure 5.0.1.

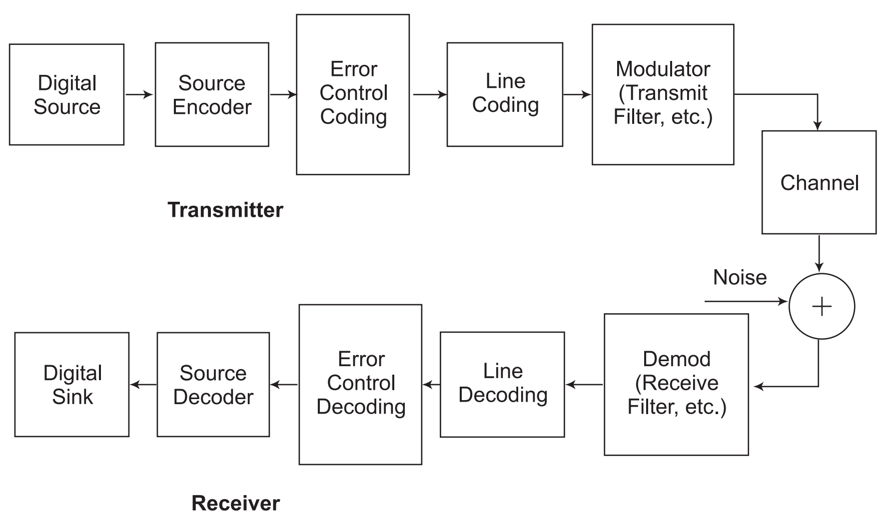


Figure 5.0.1 A Typical Transmission Model of Digital Data

Assume that the input digital source delivers output in the form of symbols, e.g., bytes from a PC, or 8-bit analog samples.

Source Encoder Transforms the source symbols into information sequence by means of data compression (reducing the redundancy of the information source).

Error Control Coding Adds extra bits (redundancy) to allow error checking and correction.

Line Coding Coding of the bit stream to make its spectrum suitable for the channel response. Also to ensure the presence of frequency components to permit bit timing extraction at the receiver.

Digital Modulator Generation of analog signals for transmission by the channel.

Channel Will affect the shape of the received pulses. Noise is also present at the receiver input, e.g., thermal noise, electrical interference, etc.

The receiver functions are just opposite to that of transmitter functions.

Note... *We have discussed digital source such as PCM in Chapter 1, Line Coding in Chapter 2 and Digital Modulator in Chapter 3. In this chapter, we shall discuss Source Encoder and Error Control Coding.*

A finite *discrete source* is the one which can be defined by the list of source symbols, referred as the *alphabet*, and the probabilities assigned to these symbols. The source is assumed to be *short-term stationary*, that is, the probability assigned is fixed over the observation interval. Shannon showed in his pioneer work that the performance of communications systems in a noisy environment was limited. However, by adding controlled redundancy to the information sequence, errors induced by a channel can be reduced to any desired level without sacrificing the rate of information transmission provided the rate of information is less than the channel capacity. Conversely, he showed that if redundancy was not used in a noise-controlled environment, error-free performance cannot be achieved. Consequently, many efficient error-control coding and decoding methods have been devised.

Recall

Refer Chapter 4: Information Theory

A Source Encoding Model

A typical source-encoding model depicts a discrete memoryless source X having finite entropy $H(X)$ and a finite set of source symbols $\{x_1, x_2, \ldots, x_k\}$, known as the *source alphabets*, with corresponding probabilities of occurrence p_k, where $k = 1, 2, \ldots, K$. Let the symbol s_k is converted to a binary sequence b_k by the distortion-less source encoder.

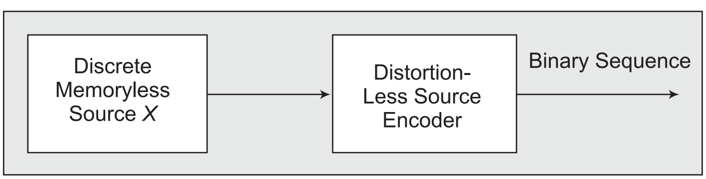


Figure 5.0.2 A Typical Source Encoding Model

Thus, the input to the source encoder is a sequence of symbols occurring at a rate of R symbols per second. The source encoder converts the symbol sequence into a binary sequence of 0s and 1s by assigning codewords to each symbol in the input sequence.

Source Encoding

Coding is an important consideration in the design of highly reliable modern digital communications systems. The *source encoder* transforms the source output into *information sequence* by means of data compression (reducing the redundancy of the information source).

Channel Encoding

Next to the source encoder in the transmitter section of a digital communication system is the channel encoder. The *channel encoder* converts the information sequence into a discrete *encoded sequence* (also called *codeword*) to combat the errors introduced in the transmitted signal by noisy channel.

Error-control Coding

In typical applications, *error-control coding* is used in environments in which the transmitter power is limited because increasing the transmitter power almost always improves system performance in a noise-limited environment (but not in a cellular interference-limited environment).

- The error control bits added in the *block codes* will correct a limited number of bit errors in each codeword.
- *Convolution coding* is an alternative to block coding of digital messages. It can offer higher coding gain for both hard and soft kind of decoding.

However, there are certain situations when the average bit error rate is small, yet the error-correcting codes such as systematic or non-systematic block codes are not effective in correcting the errors. Because the errors are clustered, we can say that the errors have occurred in bursts. For example, in radio transmission due to lightning, burst errors occur as long as it lasts; fading wireless channels in which the levels of the received signal power varies a lot and the received signal power is quite low. This requires another dimension in channel coding which is known as *interleaving*.

IMPORTANT!

The goal of error-control coding is to approach the limits imposed by information theory but constrained by practical considerations.

A PRIMER

We need a Reliable and Efficient Digital Communication System!

The signals generated by the physical source in their natural form contain a significant amount of redundant information. For example, the speech contains lot of pauses in between the spoken words. This type of redundant information, when transmitted along with useful information, occupies the channel bandwidth.

For efficient utilization of channel bandwidth, the redundant information should be removed from the signal prior to transmission in such a way so that the original data can be reconstructed at the receiver without any loss of information. The resultant source code provides a representation of the original information which is quite efficient in terms of the average number of coded bits per symbol. The extent of removal of redundant information from the source data is limited by the entropy of the source. Generally, the source information has some output data which occurs more frequently and other output data which occurs less frequently. So data compaction necessitates the assignment of short codes to the most frequently occurring symbols, and longer codes to the less frequently occurring symbols.

Source Encoding—A Key to Efficient Digital Communication System

Source encoding is the key aspect in the design of an efficient digital communication system which can, in principle, obtain the theoretical Shannon limit for channel capacity. Source encoding is the process by which data generated by a discrete memoryless source is represented efficiently. The redundant data bits in the source symbols are reduced by applying the concepts of information theory in the source encoder—a device that performs source encoding. The efficiency of a source encoder is determined precisely by the extent to which the Shannon limit is achieved.

The main objectives of source encoding are to form efficient descriptions of the source information for a given available data rate as well as to allow low data rates to obtain an acceptable efficient description of the source information. It may be noted that source encoding may be needed for a continuous source as well as discrete source. For continuous sources, however, waveform coding (analog-to-digital conversion, as discussed in Chapter 1) is required prior to the source encoding. For discrete memoryless source, the source encoding is directly related to the information content and the statistical correlation among the source symbols.

Objectives of Source Encoding

A source generates information in the form of messages, intended for onward transmission to one or more destinations. The source information may not be always in the form suitable for the transmission channel, which also does not exhibit ideal characteristics. It is usually bandlimited and introduces noise, and, therefore, efficient utilization of the channel needs proper source encoding.

IMPORTANT!

As long as channel noise exists, any type of communication cannot be free of errors. It is possible to improve the accuracy of the received signals by reducing the error probability but at the cost of reducing the transmission rate. However, Shannon showed that for a given channel, as long as the rate of information digits per second to be transmitted is maintained within a certain limit (determined by the physical channel), it is possible to achieve error-free communication. Because the presence of random disturbance in a channel does not, by itself, define any limit on transmission accuracy except that it does impose a limit on the information rate for which an arbitrarily small error probability can be achieved.

Need of Error-free Communication

Nyquist formulation on channel capacity assumes noise-free communication channel whereas Shannon formulation includes signal-to-noise power ratio applicable to the type of transmission media.

IMPORTANT!

In this chapter...

- We begin with definitions of important parameters used in source encoding of binary data, followed by classification of source codes and source-coding theorem. **LO 5.1**
- Then we describe various source encoding techniques such as Shannon–Fano, Huffman, and Lempel–Ziv which are widely employed in modern digital communication systems. **LO 5.2**
- To combat channel error during digital transmission, we introduce error-control codes in source-encoded binary data. We discuss implementation of various error-control channel coding techniques such as Hamming codes, cyclic codes, BCH code, Hadamard code, and LDPC code. **LO 5.3**
- Next, we illustrate the concept and application of convolution codes, Viterbi and sequential decoding algorithms that have been recently used in wireless digital communications. **LO 5.4**
- Finally, we briefly describe burst-error correction coding techniques such as interleaving, RS codes, and turbo codes. **LO 5.5**

LO 5.1

5.1 BASICS OF SOURCE ENCODING

What is Meant by a Message of a Source?

A source may be analog or digital. A digital source produces a number of discrete symbols or digits which either individually or in a group may constituent a message. For example, the output from the keyboard of a computer. A binary source produces two different symbols or digits usually represented by zero (0) or one (1). An M-ary source produces M different symbols where M is equal to 2^n, where n is any integer more than 1. For example, 4-ary, 8-ary, 16-ary, and so on. In general, the source symbols are not equally likely, that is, apriori probabilities of occurrence of the symbols are not the same.

Why Source Encoding?

For transmission through a binary channel, the output of the source symbols, either it is in binary form or in M-ary form, is encoded into a group of binits of the same or different duration. This is called *source encoding*. The binary source encoder converts incoming source symbols from a discrete memoryless source generating symbols with different probability at a variable symbol rate to binary codewords at a fixed binary symbol rate or bit rate.

The system *advantages* of source coding include reduction in the need for the system resources of bandwidth and energy per bit required to deliver a description of the source.

Its *disadvantages* are sophisticated computation and additional memory requirement.

Important Definitions

Let us now define some coding parameters applicable to source encoding:

Average Codeword Length In physical terms, the average codeword length, L_{avg} represents the average number of bits per source symbol used in the source encoding process. If the length of the binary codeword assigned by the source encoder to symbol s_k is l_k bits then the average codeword length L_{avg} per source symbol is defined as

$$\boxed{L_{avg} = \sum_{k=1}^{K} p_k l_k} \text{ bits per source symbol}$$

It is obvious that the average codeword length L_{avg} should be minimum for efficient transmission.

Code Efficiency The code efficiency, η_{code} of the source encoder is defined as the ratio of minimum possible value of average codeword length L_{min} to the average codeword length L_{avg} of the symbols used in the source encoding process. That is,

$$\boxed{\eta_{code} = \frac{L_{min}}{L_{avg}}}$$

Generally, $L_{avg} \geq L_{min}$. Therefore, $\eta_{code} \leq 1$. Obviously, the code is said to be the most efficient when $\eta_{code} = 1$ and the source encoder is said to be an efficient one. The code efficiency is usually expressed in percent as

$$\eta_{code}(\%) = \frac{L_{min}}{L_{avg}} \times 100$$

Code Redundancy The redundancy of a code, γ is defined as

$$\boxed{\gamma = 1 - \eta_{\text{code}}}$$

Basic Requirements of an Efficient Source Encoder

- The codewords generated by the source encoder should be in binary form.
- The concept of variable-length code for each source symbol should be applied, that is, if some source symbols are likely to occur more often than others, then short codewords can be assigned to them, and if some source symbols are likely to occur rarely, then longer codewords can be assigned to them.
- The source code should be uniquely decodable so that the original source sequence can be reconstructed perfectly from the encoded binary sequence.[1]

5.1.1 Classification of Source Codes

Fixed-Length Source Codes

The simplest source encoder can assign a fixed length binary codeword to each symbol in the input sequence. But fixed length source coding of individual symbols of a discrete source output is efficient only if all symbols occur with equal probabilities in a statistically independent sequence. A fixed-length code assigns fixed number of bits to the source symbols, irrespective of their statistics of occurrence.

For a discrete memoryless source having source alphabet $\{s_1, s_2, ..., s_k\}$. If k is a power of 2, the number of bits (codeword length) required for unique coding is $\log_2 k$. If k is not a power of 2, the number of bits (codeword length) required for unique coding will be $[(\log_2 k) + 1]$. Table 5.1.1 shows an example of fixed-length ($l_k = 2$) binary source code.

Table 5.1.1 Fixed-Length Source Code

Source Symbol, s_k	Codeword, b_k	Length of Code, l_k
s_1	00	2
s_2	01	2
s_3	10	2
s_4	11	2

A typical example of fixed-length source code is the ASCII code for which all source symbols (*A* to *Z*, *a* to *z*, 0 to 9, punctuation marks, etc.) have a 7-bit codeword.

Variable-Length Source Codes

For a variable-length source code, the number of bits (length) assigned to a codeword is not fixed. When the source symbols are not equiprobable, a variable-length source code can be more efficient than a fixed-length source code.

- If some source symbols are likely to occur more frequently than other symbols then we assign short codewords (less number of bits per symbol).
- If some source symbols are likely to occur less frequently then we assign longer codewords (more number of bits per symbol) can be assigned to them.

Table 5.1.2 shows an example of variable-length binary source code.

[1] The important parameters of a source encoder include symbol size, codeword length, average length of the codeword, and the code efficiency. Due to constraints imposed by practical systems, the actual output information rate of source encoders will be greater than the source information rate.

Table 5.1.2 Variable-Length Binary Source Code

Source Symbol, s_k	Codeword, b_k	Length of Code, l_k
s_1	0	1
s_2	10	2
s_3	110	3
s_4	111	3

As an example, we can consider the English alphabet consisting of 26 letters (a to z). Some letters such as a, e, o, etc., are used more frequently in a word or in a sentence as compared to some other letters such as q, x, z, etc. By assigning variable-length codes, we might require overall fewer number of bits to encode an entire given text than to encode the same with a fixed-length source code.

Distinct Codes

A code is called *distinct* if each codeword is unique and clearly distinguishable from the other. Let us consider an example of a source alphabet $\{s_1, s_2, s_3, s_4\}$ having their respective codewords as {00, 01, 10, 11}.

All codewords are unique and can be easily distinguished from each other. Table 5.1.3 shows an example of distinct binary source code.

Table 5.1.3 Distinct Binary Source Code

Source Symbol, s_k	Code 1	Code 2	Code 3	Code 4
s_1	00	0	0	1
s_2	01	1	10	01
s_3	10	00	110	001
s_4	11	11	111	0001

Uniquely Decodable Codes

A distinct code is said to be uniquely decodable if the original source symbol sequence can be reconstructed perfectly from the received encoded binary sequence. It may be noted that a uniquely decodable code can be both fixed-length code and variable-length code.

Let us consider four source symbols A, B, C, and D encoded with two different coding techniques—Code 1 with fixed-length binary code, and Code 2 with variable-length binary code, as shown in Table 5.1.4.

Table 5.1.4 Binary Source Codes

Source Symbol	Code 1 (fixed-length)	Code 2 (variable-length)
A	00	0
B	01	1
C	10	00
D	11	01

Let us determine whether these codes are uniquely decodable codes. The encoded format for a message 'A BAD CAB' will be:

Using Code 1 (fixed-length): {00}{010011}{100001} (Total 14 bits)

Using Code 2 (variable-length): {0}{1001}{0001} (Total 9 bits)

Although Code 2 requires fewer number of bits for the same message, yet it is not a uniquely decodable code. *Why?* The encoded message sequence 0 1001 0001 can be regrouped as {0}{1}{00}{1} {00}{01} which means A BCB CD. It can also be regrouped as {0} {1}{00}{1}{0}{0}{01} which stands for A BCB AAD.

On the other hand, although Code 1 requires more number of bits but it is uniquely decodable code as each group must include 2 bits together.

Prefix Source Code

A prefix source code is defined as a code in which no codeword is the prefix of any other code assigned to the symbol of the given source alphabet.

- A prefix source code is used to evolve uniquely decodable source code representing the output of this source.
- Any code sequence made up of the initial part of the codeword is called a prefix of the codeword.
- The end of a codeword is always recognizable in prefix codes.
- Prefix codes are distinguished from other uniquely decodable codes.

Table 5.1.5 shows the prefix code for a source symbol having four unique symbols.

Table 5.1.5 An Example of a Prefix Code

Source Symbol	Prefix Code (variable-length)
A	0
B	10
C	110
D	1110

The encoded format for a message 'A BAD CAB' will be {0} {10 110 1110} {110 0 10} - Total 16 bits. For example,

- If 0 is received then it is the complete codeword for symbol 'A' and it is not the prefix of any other symbol.
- If 10 is received then it is the complete codeword for symbol 'B' and it is not the prefix of any other symbol.
- If 110 is received then it is the complete codeword for symbol 'C' and it is not the prefix of any other symbol.
- If 1110 is received then it is the complete codeword for symbol 'D' and it is not the prefix of any other symbol.

So no codeword forms the prefix of any other codeword. This is known as *prefix condition* or *prefix-free property*. Thus, the code illustrated in Table 5.1.5 is the prefix code.

Condition for Constructing Prefix Code Consider a discrete memoryless source of alphabet $[s_1, s_2, ..., s_k]$ having respective probabilities $[p_1, p_2, ..., p_k]$.

The average codeword length of a prefix code is given by $L_{\text{avg}} = \sum_{k=1}^{K} \frac{l_k}{2^{l_k}}$; where l_k is the codeword length of the source symbol s_k.

If the symbol s_k is generated by the source with probability $p_k = \frac{1}{2^{l_k}}$ then $L_{avg} = \sum_{k=1}^{K} p_k l_k$.

The corresponding entropy of the source using prefix code is given by

$$H(X) = \sum_{k=1}^{K} \left(\frac{1}{2^{l_k}} \right) \log_2 (2^{l_k}) = \sum_{k=1}^{K} \left(\frac{l_k}{2^{l_k}} \right) = \sum_{k=1}^{K} p_k l_k$$

It is interesting to note that if in a given discrete memoryless source of entropy $H(X)$, the average codeword length L_{avg} is same as entropy $H(X)$, then a prefix code can be constructed.

Note... *For illustration, see Solved Example 5.1.5.*

Instantaneous Source Codes

A uniquely decodable source code is said to be an *instantaneous source code* if the end of any codeword is recognizable without checking any subsequent codewords. An example of such code is prefix source code. Prefix codes are also referred to as *instantaneous source codes* because the decoding of a prefix code can be accomplished only after receiving the complete binary sequence representing a source symbol. Table 5.1.6 illustrates examples of instantaneous source codes for a source having four symbols.

Table 5.1.6 Instantaneous Source Code

Source Symbol, s_k	Code 1	Code 2	Code 3
s_1	00	0	1
s_2	01	10	01
s_3	10	110	001
s_4	11	111	0001

Optimal Source Codes

A code is said to be an *optimal source code* if it is instantaneous and has minimum average length of its codeword for a given source with a particular assignment for the source symbols. Table 5.1.7 shows an example of optimal source code.

Table 5.1.7 Optimal Source Code

Source Symbol, s_k	Codeword, b_k
s_1	00
s_2	01
s_3	10
s_4	11

It can be seen that the average length of the code is also 2 as the length of each codeword is 2.

Entropy Coding

Entropy coding is a source-coding technique in which the design of a variable-length code ensures that the average codeword length approaches the entropy of discrete memoryless source. There are two main types of entropy-coding techniques: Shannon–Fano source coding and Huffman source coding. These source-coding techniques are discussed with illustrations in Section 5.2.

LET'S RECONFIRM OUR UNDERSTANDING!!

- Define Average Codeword Length.
- Which is better—Fixed-length or Variable-length Code?
- Does the code {0, 01, 011, 0111} satisfy the prefix-free property?

5.1.2 Kraft–McMillan Inequality

Define

The *Kraft–McMillan Inequality* specifies a condition on the codeword length of the prefix source code.

It is defined as $\boxed{\sum_{k=1}^{K} 2^{-l_k} \leq 1}$; where l_k, $k = 1, 2, ..., K$ is the length of the codeword for symbol s_k.

Its Significance

- If a code satisfies the Kraft–McMillan inequality condition then it may or may not be a prefix code.
- If a code violates the Kraft–McMillan inequality condition then it is certain that it cannot be a prefix code.

As an example, let us consider that the codewords [0, 10, 110, 111] are assigned to four source symbols [s_1, s_2, s_3, s_4] with their respective probability as $\left[\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\right]$. It is observed that the specified codewords is a prefix code which satisfies the Kraft–McMillan inequality condition as well as it is uniquely decodable. Using the Kraft–McMillan inequality, a prefix code can be constructed with an average codeword length L_{avg} such that $H(X) \leq L_{avg} < H(X) + 1$.

IMPORTANT!

The Kraft–McMillan inequality condition neither provides any information about the codewords nor confirms about the validity of a prefix code which has to be uniquely decodable code.

Note... *See Solved Example 5.1.8 on application of Kraft Inequality.*

5.1.3 Source-Coding Theorem

Statement

The source-coding theorem states that for a given discrete memoryless source X, the average codeword length L_{avg} per symbol for any distortion-less source encoding scheme is bounded by the source entropy $H(X)$ as $\boxed{L_{avg} \geq H(X)}$

Its Significance

According to this theorem,

- The entropy $H(X)$ represents a fundamental limit on the average number of bits (codeword) per source symbol necessary to represent a discrete memoryless source, and
- The average codeword length L_{avg} cannot be made smaller than the entropy $H(X)$.

Thus, the source-coding theorem describes the lower bound of the source-coding theorem for any decodable code. It means that L_{avg} can be made as close to $H(X)$ as desired for a suitably chosen code.

When $L_{min} = H(X)$, the code efficiency of a binary source encoder can be written in terms of the entropy $H(X)$ as

$$\boxed{\eta_{code} = \frac{H(X)}{L_{avg}}}$$

where the $H(X)$ represents the entropy of a discrete memoryless source with source alphabet X, and is given as

$$\boxed{H(X) = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)}$$

Similarly, for M-ary source, the code efficiency of the source encoder is given by

$$\boxed{\eta_{code}(\%) = \frac{H(X)}{L_{avg} \times \log_2 M} \times 100}$$

where M is the number of levels. $M = 2$ in case of binary source.

Condition for Optimum Source Code

Thus, we see that the condition $L_{min} = H(X)$ gives an *optimum source code*, and $L_{min} > H(X)$ specifies the sub-optimum source code.

The average length of the codeword of an optimum source code is the entropy of the source itself. But to obtain this codeword length, an infinite sequence of messages has to be encoded at a time which is next to impossible. If each message is encoded directly without using longer sequences, then the average length of the codeword per message will be generally greater than the source entropy.[2]

SOLVED EXAMPLE 5.1.1 Code Efficiency

Consider there are two symbols generated by a discrete memoryless source s_1 and s_2. The respective probabilities of occurrence and code assigned are shown in Table 5.1.8. Assuming noiseless channel, compute the code efficiency and code redundancy.

Table 5.1.8 Binary-Code Representation of Given Symbols

Message Symbol, s_k	Probability, p_k	Binary Code, b_k
s_1	0.8	0
s_2	0.2	1

Solution

We know that $\eta_{code} = \dfrac{H(X)}{L_{avg}} = \dfrac{\sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)}{\sum_{k=1}^{K} p_k l_k}$

The entropy, $H(X) = \sum_{k=1}^{2} p_k \log_2\left(\frac{1}{p_k}\right) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right)$

[2] In practice, it is not desirable to use long sequences of the message at a time as they cause transmission delay and increases hardware complexity. Hence, it is preferable to encode messages directly even if the codeword length is slightly longer.

Using the given data, we have

$$\Rightarrow \qquad H(X) = 0.8 \times \log_2\left(\frac{1}{0.8}\right) + 0.2 \times \log_2\left(\frac{1}{0.2}\right) = 0.722 \text{ bits/symbol}$$

The average length of codeword, $L_{\text{avg}} = \sum_{k=1}^{2} p_k l_k = p_1 l_1 + p_2 l_2$

From the given data, we deduce that $l_1 = 1$ and $l_2 = 1$.

$$\Rightarrow \qquad L_{\text{avg}} = 0.8 \times 1 + 0.2 \times 1 = 1 \text{ bit}$$

Hence, the code efficiency, $\eta_{\text{code}} = \dfrac{0.722 \text{ bit}}{1 \text{ bit}} = 0.722; \text{ or } 72.2\%$ **Ans.**

We know that the code redundancy, $\gamma = 1 - \eta_{\text{code}}$

Hence, the code redundancy, $\gamma = 1 - 0.722 = 0.728; \text{ or } 27.8\%$ **Ans.**

SOLVED EXAMPLE 5.1.2 Code Efficiency of Binary Code

Consider there are four messages generated by a source having their respective probabilities of occurrence as 1/2, 1/4, 1/8, 1/8. Assuming noiseless channel, compute the code efficiency if a binary code is applied for coding the messages.

Solution

We know that $\eta_{\text{code}} = \dfrac{H(X)}{L_{\text{avg}}} = \dfrac{\sum_{k=1}^{K} p_k \log_2\left(\dfrac{1}{p_k}\right)}{\sum_{k=1}^{K} p_k l_k}$

Since there are 4 messages, 2 bits are required to represent each one of them. This is shown in Table 5.1.9.

Table 5.1.9 Binary-code Representation of Given Messages

Message symbol, s_k	Probability, p_k	Binary code, b_k	Length of code, l_k
s_1	1/2	00	2
s_2	1/4	01	2
s_3	1/8	10	2
s_4	1/8	11	2

Using the given data, we can write

$$\eta_{\text{code}} = \frac{\left(\frac{1}{2}\right)\log_2\left(\frac{1}{1/2}\right) + \left(\frac{1}{4}\right)\log_2\left(\frac{1}{1/4}\right) + \left(\frac{1}{8}\right)\log_2\left(\frac{1}{1/8}\right) + \left(\frac{1}{8}\right)\log_2\left(\frac{1}{1/8}\right)}{\frac{1}{2}\times 2 + \frac{1}{4}\times 2 + \frac{1}{8}\times 2 + \frac{1}{8}\times 2}$$

$$\Rightarrow \qquad \eta_{\text{code}} = \frac{\frac{1}{2}\times 1 + \frac{1}{4}\times 2 + \frac{1}{8}\times 3 + \frac{1}{8}\times 3}{1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{4}} = \frac{\frac{7}{4}}{2} = \frac{7}{8}$$

Hence, $\eta_{\text{code}}(\%) = \frac{7}{8} \times 100 = 87.5\%$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 5.1.3 AConsider an alphabet of a discrete memoryless source having eight source symbols with their respective probabilities as given below:

$$\{s_k\} = \{s_1 \quad s_2 \quad s_3 \quad s_4 \quad s_5 \quad s_6 \quad s_7 \quad s_8\}$$
$$\{p_k\} = \{0.48 \quad 0.15 \quad 0.10 \quad 0.10 \quad 0.07 \quad 0.05 \quad 0.03 \quad 0.02\}$$

Determine the entropy of the specified discrete memoryless source. If the simple binary encoding is used, what is the maximum and average length of the codewords?

SOLVED EXAMPLE 5.1.4 Fixed versus Variable-length Source Codes

Consider there are four symbols generated by a discrete memoryless source. Their respective probabilities of occurrence and two different binary codes assigned to these symbols are shown in Table 5.1.10. Assuming noiseless channel, compare their code efficiency.

Table 5.1.10 Comparison of Two Binary Codes

Message Symbol, s_k	Probability, p_k	Binary Code 1	Binary Code 2
s_1	1/2	00	0
s_2	1/4	01	10
s_3	1/8	10	110
s_4	1/8	11	111

Solution

We know that code efficiency, $\eta_{\text{code}} = \dfrac{H(X)}{L_{\text{avg}}} = \dfrac{\sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)}{\sum_{k=1}^{K} p_k l_k}$

For *Code 1*, the entropy is

$$H(X) = \sum_{k=1}^{4} p_k \log_2\left(\frac{1}{p_k}\right) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right) + p_3 \log_2\left(\frac{1}{p_3}\right) + p_4 \log_2\left(\frac{1}{p_4}\right)$$

Using the given data, we have

$$H(X) = \frac{1}{2} \times \log_2\left(\frac{1}{1/2}\right) + \frac{1}{4} \times \log_2\left(\frac{1}{1/4}\right) + \frac{1}{8} \times \log_2\left(\frac{1}{1/8}\right) + \frac{1}{8} \times \log_2\left(\frac{1}{1/8}\right) = 1.75 \text{ bits/symbol}$$

We observe that Code 1 is a fixed-length code having codeword length as 2.

The average length of codeword per symbol,

$$L_{avg} = \sum_{k=1}^{4} p_k l_k = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4$$

$$\Rightarrow \quad L_{avg(1)} = (1/2) \times 2 + (1/4) \times 2 + (1/8) \times 2 + (1/8) \times 2 = 2 \text{ bits}$$

Hence, the code efficiency for Code 1, $\eta_{code(1)} = \dfrac{1.75 \text{ bits}}{2 \text{ bits}} = 0.875;$ or 87.5% **Ans.**

For *Code 2*, the entropy will be same as calculated for Code 1 because the probability corresponding to each symbol is the same. That is, $H(X) = 1.75$ bits/symbol

We observe that Code 2 is a variable-length code. The average length of codeword per symbol for the given data will be

$$L_{avg} = \sum_{k=1}^{4} p_k l_k = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4$$

$$\Rightarrow \quad L_{avg(2)} = (1/2) \times 1 + (1/4) \times 2 + (1/8) \times 3 + (1/8) \times 3 = 1.75 \text{ bits}$$

Hence, the code efficiency for Code 2, $\eta_{code(2)} = \dfrac{1.75 \text{ bits}}{1.75 \text{ bits}} = 1;$ or 100% **Ans.**

Thus, variable-length Code 2 is better than fixed-length Code 1.

SOLVED EXAMPLE 5.1.5 Code Efficiency of Prefix Source Code

Consider there are four symbols generated by a source having their respective probabilities of occurrence as 1/2, 1/4, 1/8, 1/8. Assuming noiseless channel, compute the code efficiency if a prefix source code is applied for coding the symbols.

Solution

We know that
$$\eta_{code} = \frac{\sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)}{\sum_{k=1}^{K} p_k l_k}$$

The prefix code is shown in Table 5.1.11.

Table 5.1.11 Prefix-code Representation of Given Messages

Message symbol, s_k	Probability, p_k	Binary code, b_k	Length of code, l_k
s_1	1/2	0	1
s_2	1/4	10	2
s_3	1/8	110	3
s_4	1/8	111	3

Using the given data, we can write

$$\eta_{code} = \frac{\left(\frac{1}{2}\right)\log_2\left(\frac{1}{1/2}\right) + \left(\frac{1}{4}\right)\log_2\left(\frac{1}{1/4}\right) + \left(\frac{1}{8}\right)\log_2\left(\frac{1}{1/8}\right) + \left(\frac{1}{8}\right)\log_2\left(\frac{1}{1/8}\right)}{\frac{1}{2}\times 1 + \frac{1}{4}\times 2 + \frac{1}{8}\times 3 + \frac{1}{8}\times 3}$$

$$\eta_{\text{code}} = \frac{\frac{1}{2}\times 1+\frac{1}{4}\times 2+\frac{1}{8}\times 3+\frac{1}{8}\times 3}{\frac{1}{2}+\frac{1}{2}+\frac{3}{8}+\frac{3}{8}} = \frac{\frac{7}{4}}{\frac{7}{4}} = 1 \text{, or } 100\%$$ **Ans.**

SOLVED EXAMPLE 5.1.6 Probability Calculations for Prefix Code

Show that the probability of 0 and 1 are equal in case of prefix coding, and, therefore, the coding efficiency is 100%. Assume noiseless channel.

Solution Let us consider the example of four messages having their respective probabilities as 1/2, 1/4, 1/8, 1/8. The probability of 0 is given by

$$P(0) = \frac{\sum_{k=1}^{4} p_k C_{k0}}{\sum_{k=1}^{4} p_k l_k}$$; where C_{k0} denotes the number of 0s in the k^{th} coded message.

Table 5.1.12 gives the prefix-code representation along with number of 0s and 1s in each code.

Table 5.1.12 Prefix-code Representation for $P(0)$ and $P(1)$ Calculations

Message Symbol	Probability	Prefix Code	Length of Code	Number of 0s	Number of 1s
s_1	1/2	0	1	1	0
s_2	1/4	10	2	1	1
s_3	1/8	110	3	1	2
s_4	1/8	111	3	0	3

Using the given data, we can write

$$P(0) = \frac{\left(\frac{1}{2}\times 1\right)+\left(\frac{1}{4}\times 1\right)+\left(\frac{1}{8}\times 1\right)+\left(\frac{1}{8}\times 0\right)}{\frac{1}{2}\times 1+\frac{1}{4}\times 2+\frac{1}{8}\times 3+\frac{1}{8}\times 3} = \frac{\frac{7}{8}}{\frac{7}{4}} = \frac{1}{2}$$

Similarly,

$$P(1) = \frac{\left(\frac{1}{2}\times 0\right)+\left(\frac{1}{4}\times 1\right)+\left(\frac{1}{8}\times 2\right)+\left(\frac{1}{8}\times 3\right)}{\frac{1}{2}\times 1+\frac{1}{4}\times 2+\frac{1}{8}\times 3+\frac{1}{8}\times 3} = \frac{\frac{7}{8}}{\frac{7}{4}} = \frac{1}{2}$$

Therefore, we see that $P(0) = P(1) = \frac{1}{2}$,

and as calculated in the previous example, $\eta_{\text{coding}} = 100\%$. **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 5.1.7 The four symbols produced by a discrete memoryless source have their respective probabilities as 0.5, 0.25, 0.125, and 0.125, respectively. The codewords for each of these symbols after applying a particular source encoding technique are 0, 10, 110, and 111, respectively. Determine the entropy of the source, the average codeword length, and the code efficiency.

SOLVED EXAMPLE 5.1.8 **Application of Kraft Inequality**

Consider there are four source symbols generated by a discrete memoryless source. These are encoded with four different binary codes along with their respective codeword lengths and shown in Table 5.1.13. Find out which code(s) satisfy the Kraft inequality criterion.

Table 5.1.13 Different Binary Codes

Message Symbol, s_k	Binary Code 1 (l_k)	Binary Code 2 (l_k)	Binary Code 3 (l_k)	Binary Code 4 (l_k)
s_1	00 (2)	0 (1)	0 (1)	0 (1)
s_2	01 (2)	10 (2)	11 (2)	100 (3)
s_3	10 (2)	11 (2)	100 (3)	110 (3)
s_4	11 (2)	110 (3)	110 (3)	111 (3)

Solution

We know that Kraft inequality states that $\sum_{k=1}^{K} 2^{-l_k} \le 1$

For given four symbols, we have $\sum_{k=1}^{4} 2^{-l_k} \le 1; \Rightarrow (2^{-l_1} + 2^{-l_2} + 2^{-l_3} + 2^{-l_4}) \le 1$

For Code 1, $l_1 = 2, l_2 = 2, l_3 = 2, l_4 = 2$ (Given)

$(2^{-2} + 2^{-2} + 2^{-2} + 2^{-2}) = (1/4 + 1/4 + 1/4 + 1/4) = 1; (= 1)$

Hence, the Kraft inequality criterion for Code 1 *is satisfied.*

For Code 2, $l_1 = 1, l_2 = 2, l_3 = 2, l_4 = 3$ (Given)

$(2^{-1} + 2^{-2} + 2^{-2} + 2^{-3}) = (1/2 + 1/4 + 1/4 + 1/8) = 9/8; (> 1)$

Hence, the Kraft inequality criterion for Code 2 *is not satisfied.*

For Code 3, $l_1 = 1, l_2 = 2, l_3 = 3, l_4 = 3$ (Given)

$(2^{-1} + 2^{-2} + 2^{-3} + 2^{-3}) = (1/2 + 1/4 + 1/8 + 1/8) = 8/8; (= 1)$

Hence, the Kraft inequality criterion for Code 3 *is satisfied.*

For Code 4, $l_1 = 1, l_2 = 3, l_3 = 3, l_4 = 3$ (Given)

$(2^{-1} + 2^{-3} + 2^{-3} + 2^{-3}) = (1/2 + 1/8 + 1/8 + 1/8) = 7/8; (< 1)$

Hence, the Kraft inequality criterion for Code 4 *is satisfied.*

Hence, all given binary codes are prefix codes except Code 2.

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/119

Self-Assessment Exercise linked to LO 5.1

Q5.1.1 Mention the factors which need to be considered for applying a particular source-encoding technique to a digital message source. Is the knowledge about the source characteristics essential in designing an efficient source encoder?

Q5.1.2 Define the code efficiency of a source encoder.

Q5.1.3 State the source coding theorem. What is its main significance?

Q5.1.4 Hypothesize the prefix condition in a variable-length source code.

Q5.1.5 What does the optimum source-encoding technique mean?

Q5.1.6 Consider an alphabet of a discrete memoryless source having eight source symbols with their respective probabilities as given below:

$$\{s_k\} = \{s_1 \quad s_2 \quad s_3 \quad s_4 \quad s_5 \quad s_6 \quad s_7 \quad s_8\}$$
$$\{p_k\} = \{0.02 \quad 0.03 \quad 0.05 \quad 0.07 \quad 0.10 \quad 0.10 \quad 0.15 \quad 0.48\}$$

Determine the code efficiency and code redundancy of simple binary encoding technique used for the given data.

Q5.1.7 The codewords for four symbols produced by a discrete memoryless source having probability 0.5, 0.25, 0.125, and 0.125 respectively, after applying a particular source coding technique are 0, 1, 10, and 11. Show that the average codeword length is less than the source entropy.

Q5.1.8 The codewords for four symbols produced by a discrete memoryless source have probabilities as 0.5, 0.25, 0.125, and 0.125 respectively. The average codeword length is 1.25 bits per symbol, corresponding to a particular source code 0, 1, 10, and 11. The Kraft inequality for this code is specified as 1.5. Will this code have any deciphering problem?

Note
- Level 1 and Level 2 Category
- Level 3 and Level 4 Category
- Level 5 and Level 6 Category

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 5.1: *Understand the basics of source encoding, its classifications and source-coding theorem.*

LO 5.2

5.2 SOURCE-CODING TECHNIQUES

Recall

When the symbols generated by the discrete memoryless source are not equiprobable and every symbol gets encoded by identical codeword length in the encoding process, a sub-optimum code is obtained. PCM encoding, DPCM encoding, Grey coding, or character codes like ASCII are fixed length codes. Source coding must be used along with any of these fixed-length codes in order to improve their efficiency in accordance with the source-coding theorem.

When source coding is applied to fixed-length codes used for a non-equiprobable source then the coding becomes variable-length because the codeword length for a symbol is chosen in proportion to its information content (probability of occurrence). The usual practice of assigning codewords in any source-coding algorithm is that the symbols that occur quite frequently are assigned shorter codewords and those occurring infrequently are assigned longer codewords, as in the Morse code.

What We Discuss Here ...

- The rate of transmission of information (binary data, say) over a communication channel is maximum if the symbol probabilities are all equal. So, it is desirable that the transformed code digits have equal probability of occurrence. This technique of source coding is called entropy coding and the codes are called instantaneous codes. Shannon–Fano source coding and Huffman source coding are examples of entropy coding. These are discussed in details with adequate illustrations and examples.
- Based on fixed-length code, Lempel–Ziv (LZ) code represent a variable number of source symbols and does not require knowledge of a probabilistic model of the source. A brief account of LZ source code is given here.

5.2.1 Shannon–Fano Source Code

What is Shannon–Fano Source Coding Technique?

Shannon–Fano source coding, named after *Claude Shannon* and *Robert Fano*, is a sub-optimal source coding technique for constructing a prefix code with fairly efficient variable-length codewords based on a set of discrete source symbols and their probabilities. It is suboptimal because it does not achieve the lowest possible expected codeword length.[3]

The algorithm for the Shannon–Fano source encoding is illustrated in Table 5.2.1.

Table 5.2.1 Algorithm for the Shannon–Fano Source Encoding

Step #	Action to be Taken
I	Arrange the given source symbols in order of their specified decreasing probability.
II	Group them in two groups (for a 2-level encoder) or three groups (for a 3-level encoder) in such a way so that the sum of individual probability of source symbols in each group is nearly equal.
III	In case of 2-level encoding, assign a binary digit 0 to source symbols contained in first group, and a binary digit 1 to source symbols contained in second group. In case of 3-level encoding, assign a level -1 to source symbols contained in first group, a level 0 to source symbols contained in second group, and a level 1 to source symbols contained in third group.
IV	If any of the divided groups contains more than one symbol, divide them again in two or three groups, as the case may be, in such a way so that the sum of individual probability of source symbols in each group is nearly equal.
V	The assignment of a binary level is done as described in Step III above,
VI	Repeat the procedure specified in steps from IV and V any number of times until a final set of two or three groups containing one source symbol each is obtained. Assign a binary level to final two or three source symbols also as obtained in Step V.

[3] Shannon–Fano source-encoding algorithm produces fairly efficient variable-length encoding. However, it does not always produce optimal prefix codes. Hence, is has limited application in the IMPLODE compression method, which is a part of the ZIP file format, where a simple algorithm with high performance and the minimum requirements for programming is desired.

Note... *The application of the Shannon–Fano source-coding algorithm specified above is explained with the help of the following example.*

SOLVED EXAMPLE 5.2.1 Creation of Shannon–Fano Codewords

Consider an alphabet of a discrete memoryless binary source having eight source symbols with their respective probabilities as given below:

$$\begin{Bmatrix} s_k \\ p_k \end{Bmatrix} = \begin{Bmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 & s_8 \\ 0.48 & 0.15 & 0.10 & 0.10 & 0.07 & 0.05 & 0.03 & 0.02 \end{Bmatrix}$$

Create a Shannon–Fano source codeword for each symbol.

Solution Table 5.2.2 shows the step-by-step procedure of creating Shannon–Fano codewords for the given source symbols.

Table 5.2.2 Shannon–Fano Source-Encoding Technique

s_k	p_k	1st Group	2nd Group	3rd Group	4th Group	5th Group	Codeword
s_1	0.48	**0**					**0**
s_2	0.15	**1**	**0**	**0**			**1 0 0**
s_3	0.10	**1**	**0**	**1**			**1 0 1**
s_4	0.10	**1**	**1**	**0**	**0**		**1 1 0 0**
s_5	0.07	**1**	**1**	**0**	**1**		**1 1 0 1**
s_6	0.05	**1**	**1**	1	0		**1 1 1 0**
s_7	0.03	**1**	**1**	**1**	**1**	**0**	**1 1 1 1 0**
s_8	0.02	**1**	**1**	**1**	**1**	**1**	**1 1 1 1 1**

Thus, the computed codeword for each given symbol is 0, 100, 101, 1100, 1101, 1110, 11110, and 11111 respectively. **Ans.**

SOLVED EXAMPLE 5.2.2 Length of Shannon–Fano Codewords

Calculate the respective length of the codewords created for each of the given source symbols in Solved Example 5.2.1.

Solution We know that the length of the codeword corresponding to a particular symbol of the message is simply the number of digits in the codeword assigned to that symbol.

As calculated in the previously solved example,

The codeword for each symbol is 0, 100, 101, 1100, 1101, 1110, 11110,and 11111.

Therefore, the respective length of the codeword is 1, 3, 3, 4, 4, 4, 5, and 5. **Ans.**

SOLVED EXAMPLE 5.2.3 Average Codeword Length

Determine the average codeword length for codewords created for each of the given source symbols in Example 5.2.1 and the respective lengths of the codewords computed in Example 5.2.2 above.

Solution We know that the average codeword length, L_{avg}, of the source encoder is given as

$$L_{avg} = \sum_{k=1}^{K} p_k l_k = \sum_{k=1}^{8} p_k l_k$$

Therefore, $$L_{avg} = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4 + p_5 l_5 + p_6 l_6 + p_7 l_7 + p_8 l_8$$

$$L_{avg} = (0.48 \times 1) + (0.15 \times 3) + (0.1 \times 3) + (0.1 \times 4) + (0.07 \times 4) + (0.05 \times 4) + (0.03 \times 5) + (0.02 \times 5)$$

$\Rightarrow$ $$L_{avg} = 0.48 + 0.45 + 0.3 + 0.4 + 0.28 + 0.2 + 0.15 + 0.1$$

Hence, the average codeword length, $L_{avg} = 2.36$ bits **Ans.**

SOLVED EXAMPLE 5.2.4 Code Efficiency

Determine the code efficiency of the given source data in Example 5.2.1 for Shannon–Fano source encoding.

Solution

We know that the coding efficiency, $\eta_{code}(\%) = \dfrac{H(X)}{L_{avg}} \times 100$

The entropy of the discrete memoryless source is given as

$$H(X) = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$$

For given eight source symbols, k varies from 1 to 8. Therefore,

$$H(X) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right) + p_3 \log_2\left(\frac{1}{p_3}\right) + p_4 \log_2\left(\frac{1}{p_4}\right) + p_5 \log_2\left(\frac{1}{p_5}\right) + p_6 \log_2\left(\frac{1}{p_6}\right) + p_7 \log_2\left(\frac{1}{p_7}\right) + p_8 \log_2\left(\frac{1}{p_8}\right)$$

$$H(X) = 0.48\log_2\left(\frac{1}{0.48}\right) + 0.15\log_2\left(\frac{1}{0.15}\right) + 0.1\log_2\left(\frac{1}{0.1}\right) + 0.1\log_2\left(\frac{1}{0.1}\right) + 0.07\log_2\left(\frac{1}{0.07}\right) + 0.05\log_2\left(\frac{1}{0.05}\right) + 0.03\log_2\left(\frac{1}{0.03}\right) + 0.02\log_2\left(\frac{1}{0.02}\right)$$

Hence, $H(X) = 2.33$ bits

The average codeword length, $L_{avg} = \sum_{k=1}^{K} p_k l_k = \sum_{k=1}^{8} p_k l_k$

Therefore, $L_{avg} = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4 + p_5 l_5 + p_6 l_6 + p_7 l_7 + p_8 l_8$

$$L_{avg} = (0.48 \times 1) + (0.15 \times 3) + (0.1 \times 3) + (0.1 \times 4) + (0.07 \times 4) + (0.05 \times 4) + (0.03 \times 5) + (0.02 \times 5)$$

$\Rightarrow$ $$L_{avg} = 0.48 + 0.45 + 0.3 + 0.4 + 0.28 + 0.2 + 0.15 + 0.1$$

Hence, the average codeword length, L_{avg} = 2.36 bits

We know that code efficiency, $\eta_{code}(\%) = \dfrac{H(X)}{L_{avg}} \times 100$

$$\Rightarrow \qquad \eta_{coding} = \frac{2.33}{2.36} \times 100 = 98.7\% \qquad \textbf{Ans.}$$

SOLVED EXAMPLE 5.2.5 Shannon–Fano Source Encoding (3-level)

Consider an alphabet of a discrete memoryless source having seven source symbols with their respective probabilities as given below:

$$\begin{Bmatrix} s_k \\ p_k \end{Bmatrix} = \begin{Bmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 \\ 0.40 & 0.20 & 0.12 & 0.08 & 0.08 & 0.08 & 0.04 \end{Bmatrix}$$

Suppose there are three symbols in an encoding alphabet.

(a) Create a Shannon–Fano source codeword for each symbol. Compute the respective length of the codewords for each of the given source symbols.
(b) Determine the average codeword length.
(c) Determine the entropy of the specified discrete memoryless source.
(d) Determine the code efficiency.

Solution

(a) To create Shannon–Fano source codewords

Table 5.2.3 shows the step-by-step procedure of generating Shannon–Fano codewords for the given source symbols.

Table 5.2.3 Shannon–Fano's Technique of Source Encoding

$[s_k]$	$[p_k]$	1st Level	2nd Level	3rd Level	Codeword	Code-length, l_k
s_1	0.40	**–1**			–1	**1**
s_2	0.20	**0**	**–1**		0 –1	2
s_3	0.12	**0**	**0**		0 0	**2**
s_4	0.08	**1**	**–1**		1 –1	**2**
s_5	0.08	**1**	**0**		1 0	**2**
s_6	0.08	**1**	**1**	**–1**	1 1 –1	**3**
s_7	0.04	**1**	**1**	**0**	1 1 0	3

The maximum codeword length for any symbol is **three**.

(b) To find the average codeword length.

We know that the average codeword length, L_{avg} of the source encoder is given as $L_{avg} = \sum_{k=1}^{K} p_k l_k$; where K = 7 (given)

For given seven source symbols, k varies from 1 to 7. Therefore,

$$L_{avg} = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4 + p_5 l_5 + p_6 l_6 + p_7 l_7$$

$$L_{avg} = (0.4 \times 1) + (0.2 \times 2) + (0.12 \times 2) + (0.08 \times 2) + (0.08 \times 2) + (0.08 \times 2) + (0.08 \times 3) + (0.04 \times 3)$$

$$\Rightarrow \quad L_{avg} = 0.4 + 0.4 + 0.24 + 0.16 + 0.16 + 0.16 + 0.24 + 0.12$$

Hence, the average codeword length, $L_{avg} = 1.72$ bits **Ans.**

(c) To determine the entropy of the specified discrete memoryless source.

We know that the entropy of the given discrete memoryless source is given as

$$H(X) = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$$

For the given seven source symbols, k varies from 1 to 7. Therefore,

$$H(X) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right) + p_3 \log_2\left(\frac{1}{p_3}\right) + p_4 \log_2\left(\frac{1}{p_4}\right) + p_5 \log_2\left(\frac{1}{p_5}\right) + p_6 \log_2\left(\frac{1}{p_6}\right) + p_7 \log_2\left(\frac{1}{p_7}\right)$$

$$H(X) = 0.4\log_2\left(\frac{1}{0.4}\right) + 0.2\log_2\left(\frac{1}{0.2}\right) + 0.12\log_2\left(\frac{1}{0.12}\right) + 0.08\log_2\left(\frac{1}{0.08}\right) + 0.08\log_2\left(\frac{1}{0.08}\right) + 0.08\log_2\left(\frac{1}{0.08}\right) + 0.04\log_2\left(\frac{1}{0.04}\right)$$

***H*(*X*) = 2.42** bits **Ans.**

(d) To determine the code efficiency

We know that the code efficiency is given by

$$\eta_{code}(\%) = \frac{H(X)}{L_{avg} \times \log_2 M} \times 100$$

$$\Rightarrow \quad \eta_{code}(\%) = \frac{2.42}{1.72 \times \log_2 3} \times 100 = 88.7\%$$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 5.2.6 Consider an alphabet of a discrete memoryless binary source having seven source symbols with their respective probabilities as given below:

$$\begin{Bmatrix} s_k \\ p_k \end{Bmatrix} = \begin{Bmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 \\ 0.4 & 0.2 & 0.12 & 0.08 & 0.08 & 0.06 & 0.06 \end{Bmatrix}$$

Apply the Shannon–Fano source encoding algorithm and find the codeword for each symbol. Also compute the efficiency of the code.

Data compaction, or lossless data compression, is the process of removing the redundant information (source coding) from the signals generated by physical sources, and is usually performed on a signal in digital form. **ATTENTION**

5.2.2 Huffman Source Code

What is Huffman Source-Coding Technique?

The Huffman code is a source code in which the average length of the codewords approaches the fundamental limit set by the entropy of a discrete memoryless source. Thus, the Huffman source-encoding technique produces source codes that always achieve the lowest possible average codeword length. It implies that a codeword assigned to a symbol is approximately equal in length to the amount of information conveyed by it. So, it is an optimal code which has the highest code efficiency or the lowest code redundancy. Hence, it is also known as *optimum code* or the *minimum redundancy code*.

The algorithm for the Huffman source coding is illustrated in Table 5.2.4.

Table 5.2.4 Algorithm for Huffman Source Coding

Step #	Action to be Taken
I	List the given source symbols in order of their specified decreasing probability. The symbols with equal probabilities can be arranged in any arbitrary order.
II	Assign a binary logic 0 and a binary logic 1 to the two source symbols of lowest probability in the list obtained in step I above. This is called splitting stage.
III	Add the probability of these two source symbols and regard it as one new source symbol. This results into reduction of the size of the list of source symbols by one. This process is called reduction level 1.
IV	Place the assigned probability of the new symbol high or low in the list with the rest of the symbols with their given probabilities. In case the assigned probability of the new symbol is equal to another probability in the reduced list, it may either may be placed higher (preferably) or lower than the original probability. However, it is presumed that whatever may be the placement (high or low), it is consistently adhered to throughout the encoding procedure.
V	Repeat the procedure specified in steps from II to IV any number of times until a final set of two source symbols (one original and the other new) is obtained. Each process results into reduction of the size of the list of source symbols by one. The final process is called the last reduction level.
VI	Assign a binary logic 0 and a binary logic 1 to the final two source symbols also as obtained in Step V above. This is known as *Huffman Tree*.
VII	Determine the codeword for each original source symbol by working backward and tracing the sequence of binary logic values 0s and 1s assigned to that symbol as well as its successors.

The application of the above algorithm specified for Huffman source-encoding technique is illustrated with the help of the following examples.

SOLVED EXAMPLE 5.2.7 **Creation of Huffman Tree**

Consider an alphabet of a discrete memoryless source having five different source symbols with their respective probabilities as 0.1, 0.2, 0.4, 0.1, and 0.2. Create a Huffman tree for the Huffman source-encoding technique.

Solution

Steps I and II: The given source symbols are listed in order of their specified decreasing probability and the two source symbols of lowest probability in the list are assigned a binary logic 0 and a binary logic 1 respectively, as shown in Table 5.2.5.

Table 5.2.5 Steps I and II for creating a Huffman Tree

Symbol, s_k	s_1	s_2	s_3	s_4	s_5
Probability, p_k	0.4	0.2	0.2	0.1	0.1
Binary Logic Level	--	--	--	**0**	**1**

Step III: The probability of the last two source symbols (i.e., s_3 and s_4) is combined into one new source symbol with probability equal to the sum of the two original probabilities (0.1 and 0.1), that is, 0.1 + 0.1 = 0.2, as shown in Figure 5.2.1.

Step I	*Step I*	*Step II*	*Step III*
Symbol, s_k	*Probability, p_k*	*Binary Logic*	*Combined Probability*
s_1	0.4	...	...
s_2	0.2	...	...
s_3	0.2	...	...
s_4	0.1	0	0.2
s_5	0.1	1	

Figure 5.2.1 Step III for Creating a Huffman Tree

Step IV: The assigned probability of the new symbol is placed in the list in accordance with the rest of the symbols with their given probabilities. Let the new symbol be placed HIGH in the list, as shown in Figure 5.2.2.

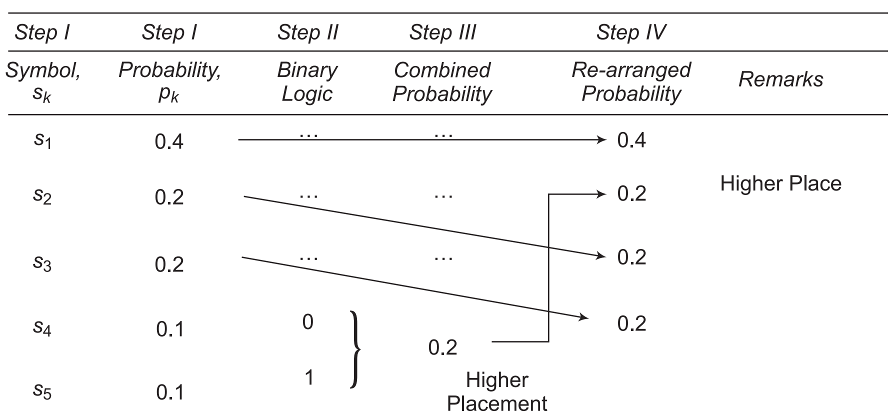


Figure 5.2.2 Step IV for Creating a Huffman Tree

Steps V and VI: The procedure specified in steps from II to IV is repeated until a final set of two source symbols (one original and the other new) is obtained. A binary logic 0 and a binary logic 1 is assigned to the final two source symbols. This is known as a Huffman tree, as illustrated in Figure 5.2.3.

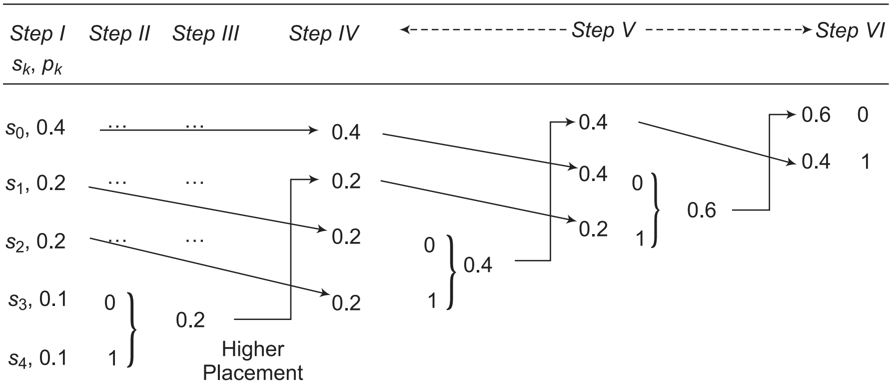


Figure 5.2.3 Huffman Tree

SOLVED EXAMPLE 5.2.8 Codeword of Huffman Source Encoding

For the Huffman tree (Refer Figure 5.2.3) created for the given source data in Example 5.2.7, compute the codeword for each of the given source symbols.

Solution The codeword for each original source symbol can be determined by working backward and tracing the sequence of binary logic values 0s and 1s assigned to that symbol as well as its successors from the Huffman tree. This is shown in Table 5.2.6.

Table 5.2.6 Codeword for Source Symbols

Symbol, s_k	s_1	s_2	s_3	s_4	s_5
Probability, p_k	0.4	0.2	0.2	0.1	0.1
Codeword	**0 0**	**1 0**	**1 1**	**0 1 0**	**0 1 1**

SOLVED EXAMPLE 5.2.9 Length of Codewords of Huffman Source Encoding

For the codewords computed in Example 5.2.8 from the Huffman tree (Figure 5.2.3) for the given source data in Example 5.2.7, find the length of the codewords for each source symbol.

Solution The length of the codewords for a source symbol is the number of binary digits in that source symbol, as computed from the Huffman tree, and is shown in Table 5.2.7.

Table 5.2.7 Codeword Length for Source Symbols

Symbol, s_k	Given Probability, p_k	Codeword	Codeword Length, l_k
s_1	$p_1 = 0.4$	0 0	$l_1 = \mathbf{2}$
s_2	$p_2 = 0.2$	1 0	$l_2 = \mathbf{2}$
s_3	$p_3 = 0.2$	1 1	$l_3 = \mathbf{2}$
s_4	$p_4 = 0.1$	0 1 0	$l_4 = \mathbf{3}$
s_5	$p_5 = 0.1$	0 1 1	$l_5 = \mathbf{3}$

SOLVED EXAMPLE 5.2.10 Average Codeword Length of Huffman Encoding

For the codewords and their codeword lengths calculated in Example 5.2.9, determine the average codeword length.

Solution We know that the average codeword length, L_{avg} of the source encoder is given as

$$L_{avg} = \sum_{k=1}^{K} p_k l_k$$

For the given five messages, k varies from 1 to 5. Therefore,

$$L_{avg} = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4 + p_5 l_5$$

$$\Rightarrow \quad L_{avg} = 0.4 \times 2 + 0.2 \times 2 + 0.2 \times 2 + 0.1 \times 3 + 0.1 \times 3$$

$$\Rightarrow \quad L_{avg} = 0.8 + 0.4 + 0.4 + 0.3 + 0.3$$

Hence, the average codeword length, $L_{avg} = 2.2$ bits **Ans.**

SOLVED EXAMPLE 5.2.11 Code Efficiency of Huffman Source Encoding

For the source data given in Example 5.2.7, show that the average codeword length, $L_{avg} = 2.2$ bits satisfies the source-coding theorem which states that $L_{avg} \geq H(X)$. Also calculate the percent increase in L_{avg}.

Solution We know that the entropy of the given discrete memoryless source is given as

$$H(X) = \sum_{k=1}^{K} p_k \log_2\left(\frac{1}{p_k}\right)$$

For the given five messages, k varies from 1 to 5. Therefore,

$$H(X) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right) + p_3 \log_2\left(\frac{1}{p_3}\right) + p_4 \log_2\left(\frac{1}{p_4}\right) + p_5 \log_2\left(\frac{1}{p_5}\right)$$

$$H(X) = 0.4 \log_2\left(\frac{1}{0.4}\right) + 0.2 \log_2\left(\frac{1}{0.2}\right) + 0.2 \log_2\left(\frac{1}{0.2}\right) + 0.1 \log_2\left(\frac{1}{0.1}\right) + 0.1 \log_2\left(\frac{1}{0.1}\right)$$

$$H(X) = 2.122 \text{ bits}$$

Hence, the average codeword length, $L_{avg} = 2.2$ exceeds the entropy $H(X) = 2.122$**.**

This satisfies the source-coding theorem which states that $L_{avg} \geq H(X)$.

Percent increase in L_{avg} is given by $\frac{\left[L_{avg} - H(X)\right]}{H(X)} \times 100\%$

For $L_{avg} = 2.2$ and $H(X) = 2.122$ bits, we get

$$\frac{\left[L_{avg} - H(X)\right]}{H(X)} \times 100 = \frac{[2.2 - 2.122]}{2.122} \times 100 = 3.67\% \qquad \textbf{Ans.}$$

SOLVED EXAMPLE 5.2.12 Huffman Tree (Alternate Method)

Consider an alphabet of a discrete memoryless source having five different source symbols with their respective probabilities as 0.1, 0.2, 0.4, 0.1, and 0.2.

(a) Create a Huffman tree by placing the combined probability lower than that of other similar probability in the reduced list.
(b) Tabulate the codeword and length of the codewords for each source symbol.
(c) Determine the average codeword length of the specified discrete memoryless source.
(d) Comment on the results obtained.

Solution

(a) The Huffman tree is created by placing the combined probability lower than that of other similar probability in the reduced list is shown in Figure 5.2.4.

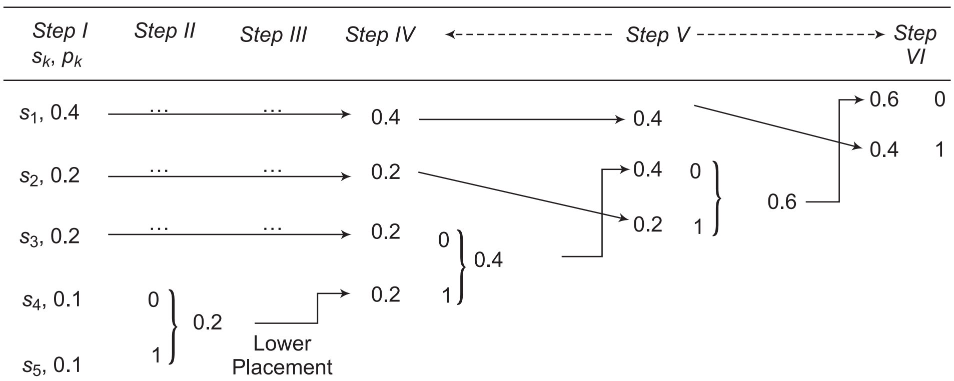


Figure 5.2.4 Huffman Tree (Alternate Method)

(b) The codewords and lengths of the codewords for each source symbol are shown in Table 5.2.8.

Table 5.2.8 Codeword and Codeword Length for Source Symbols

Symbol, s_k	Original Probability, p_k	Codeword	Codeword Length, l_k
s_1	$p_1 = 0.4$	**1**	$l_1 = \mathbf{1}$
s_2	$p_2 = 0.2$	**0 1**	$l_2 = \mathbf{2}$
s_3	$p_3 = 0.2$	**0 0 0**	$l_3 = \mathbf{3}$
s_4	$p_4 = 0.1$	**0 0 1 0**	$l_4 = \mathbf{4}$
s_5	$p_5 = 0.1$	**0 0 1 1**	$l_5 = \mathbf{4}$

It is observed that there is significant difference in the codewords as well as the length of codewords for each source symbol as compared to the earlier method of generating a Huffman tree when the combined probability is placed higher than that of other similar probability in the reduced list.

(c) We know that the average codeword length, L_{avg} of the source encoder is given as

$$L_{avg} = \sum_{k=1}^{K} p_k l_k$$

For the given five messages, k varies from 1 to 5. Therefore,

$$L_{avg} = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4 + p_5 l_5$$

$$\Rightarrow \quad L_{avg} = 0.4 \times 1 + 0.2 \times 2 + 0.2 \times 3 + 0.1 \times 4 + 0.1 \times 4$$

$$\Rightarrow \quad L_{avg} = 0.4 + 0.4 + 0.6 + 0.4 + 0.4$$

Hence, the average codeword length, $L_{avg} = 2.2$ bits **Ans.**

(d) *Comments on the Results Obtained:* Although there is significant difference in the codewords as well as the length of codewords for each source symbol as compared to the earlier method of generating a Huffman tree (higher placement of combined probability in the reduced list), yet the average codeword length remains the same.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 5.2.13 Consider an alphabet of a discrete memoryless source having six source symbols with their respective probabilities as 0.08, 0.10, 0.12, 0.15, 0.25, and 0.30. Create a binary Huffman tree and determine the codeword for each symbol.

Ex 5.2.14 For the Huffman codeword determined in Example 5.2.13, determine the minimum possible average codeword length attainable by coding an infinitely long sequence of symbols.

Ex 5.2.15 Compute the average length of the Huffman source code determined in Example 5.2.13. Also show that the code efficiency is 98.7%.

Application

Prefix and Huffman source encoding are the most commonly used source-encoding techniques which provide data compaction. Huffman codes are used in CCITT, JBIG2, H.261, H.262, H.263, H.264, etc. Modified Huffman coding is widely used in the JPEG and MPEG-1/2/4 standards.

Huffman Source Coding for *M*-ary Source

So far, the Huffman source encoding has been explained for a binary source having two levels 1 and 0. However, it can be applied to an M-ary source as well. The algorithm to create a Huffman tree is similar to what has been described for binary source. That is, the following hold:

- The symbols are arranged in the order of descending probability.
- The last M symbols are assigned one of the 0, 1, …., $(M - 1)$ levels as the first digit (most significant digit) in their codeword.
- The last M symbols are combined into one symbol having probability equal to the sum of their individual probabilities.
- The remaining symbols along with the reduced symbol (as obtained in the previous step), are arranged in the order of descending probability.
- The last M symbols are again assigned one of the 0, 1, …., $(M - 1)$ levels as the second digit in their codeword.

- This is repeated till only M symbols are left, which are also assigned one of the 0, 1, …., $(M - 1)$ levels as the last digit (least significant digit) in their codeword.

Desirable Conditions for *M*-ary Huffman Source Code

For an M-ary Huffman source code, it is desirable that there should be exactly M symbols left in the last reduced set. This can happen only if the total number of source symbols is equal to $M + k(M - 1)$ where k is the number of reduction levels possible. For example,

- For 3-ary Huffman source code, the total number of source symbols required is 5, 7, 9, 11, …. so on.
- For 4-ary Huffman source code, the total number of source symbols required is 7, 10, 13, …. so on.

This is so because each reduction level reduces the number of symbols by $(M - 1)$. If the size of the source alphabet is not equal to $M + k(M - 1)$, then the requisite number of dummy symbols with zero probability must be added.

Variations in Huffman Source-Encoding Process

It may be noted that Huffman source-encoding process, also called the Huffman tree, is not unique. There are at least two variations in the process such as

- At each splitting stage, the assignment of binary logic 0 and 1 can be arbitrarily done to the last two source symbols.
- In case the combined probability of two symbols happens to be exactly equal to any other probability in the list, it may be placed high or low consistently throughout the coding process.

This results in codes for various source symbols having different code lengths, although the average codeword length remains the same. The question here is which method should be preferred and why? The answer can be found by measuring the variability in codeword lengths of a source code with the help of variance of the average codeword length.

The *variance of the average codeword length* can be defined as

$$\boxed{\sigma^2 = \sum_{k=1}^{K} p_K (l_K - L_{\text{avg}})^2}$$

where $p_1, p_2, p_3, ..., p_K$ are the source probabilities; $l_1, l_2, l_3, ..., l_K$ are the length of the codewords assigned to source symbols s_1, s_2, s_3, ..., s_K respectively, L_{avg} is the average length of the codeword.

The application of above-mentioned algorithm specified for Huffman source-encoding technique for M-ary source is illustrated with the help of the following example:

SOLVED EXAMPLE 5.2.16 Huffman Source Encoding for *M*-ary Source

Consider an alphabet of a discrete memoryless source having the following source symbols with their respective probabilities as

$$\begin{Bmatrix} s_k \\ p_k \end{Bmatrix} = \begin{Bmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 \\ 0.40 & 0.20 & 0.12 & 0.08 & 0.08 & 0.08 & 0.04 \end{Bmatrix}$$

Suppose there are 3 number of symbols in an encoding alphabet.

(a) Create a Huffman tree following the standard algorithm for the Huffman encoding, and compute the codeword and the respective length of the codewords for each of the given source symbols.

(b) Determine the average codeword length.
(c) Determine the entropy of the specified discrete memoryless source.
(d) Determine the code efficiency.

Solution

(a) To create a Huffman Tree

Figure 5.2.5 shows the Huffman tree following the standard algorithm for the Huffman source encoding.

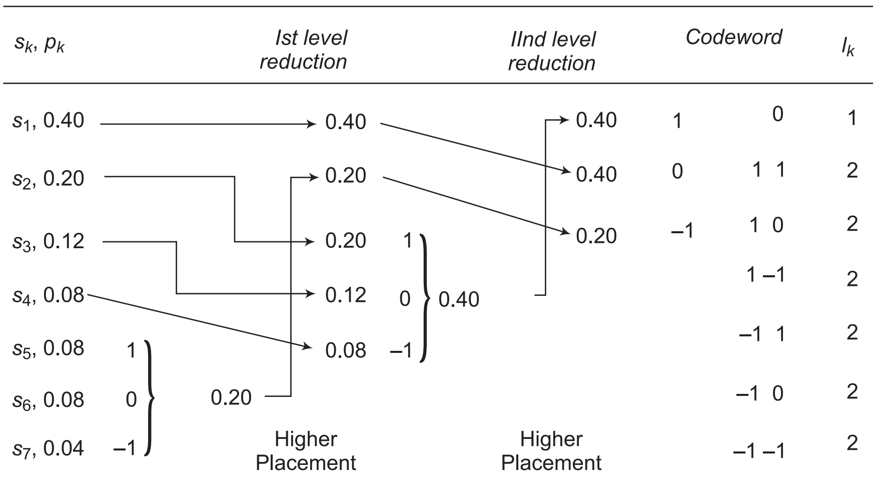


Figure 5.2.5 Huffman Source-Encoding Tree

The maximum codeword length for any symbol is two only.

(b) To determine the average codeword length.

We know that the average codeword length, L_{avg} of the source encoder is given as

$$L_{avg} = \sum_{k=1}^{K} p_k l_k$$

For given seven messages, k varies from 1 to 7, we have

$$L_{avg} = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4 + p_5 l_5 + p_6 l_6 + p_7 l_7$$

$$\Rightarrow \quad L_{avg} = (0.4 \times 1) + (0.2 \times 2) + (0.12 \times 2) + (0.08 \times 2) + (0.08 \times 2) + (0.08 \times 2) + (0.04 \times 2)$$

$$\Rightarrow \quad L_{avg} = 0.4 + 0.4 + 0.24 + 0.16 + 0.16 + 0.16 + 0.08$$

Hence, the average codeword length, L_{avg} = 1.6 3-ary units/message **Ans.**

(c) To determine the entropy of the specified memoryless source.

We know that the entropy of the given discrete memoryless source is given as

$$H(X) = \sum_{k=1}^{K} p_k \log_2 \left(\frac{1}{p_k} \right)$$

For the given seven messages, k varies from 1 to 7. Therefore,

$$H(X) = p_1 \log_2\left(\frac{1}{p_1}\right) + p_2 \log_2\left(\frac{1}{p_2}\right) + p_3 \log_2\left(\frac{1}{p_3}\right) + p_4 \log_2\left(\frac{1}{p_4}\right)$$
$$+ p_5 \log_2\left(\frac{1}{p_5}\right) + p_6 \log_2\left(\frac{1}{p_6}\right) + p_7 \log_2\left(\frac{1}{p_7}\right)$$

$$H(X) = 0.4 \log_2\left(\frac{1}{0.4}\right) + 0.2 \log_2\left(\frac{1}{0.2}\right) + 0.12 \log_2\left(\frac{1}{0.12}\right) + 0.08 \log_2\left(\frac{1}{0.08}\right)$$
$$+ 0.08 \log_2\left(\frac{1}{0.08}\right) + 0.08 \log_2\left(\frac{1}{0.08}\right) + 0.04 \log_2\left(\frac{1}{0.04}\right)$$

$\boldsymbol{H(X) = 2.42}$ 3-ary units/message **Ans.**

(d) To determine the code efficiency.

We know that the code efficiency is given by

$$\eta_{\text{code}}(\%) = \frac{H(X)}{L_{\text{avg}} \times \log_2 M} \times 100$$

$$\Rightarrow \qquad \eta_{\text{code}}(\%) = \frac{2.42}{1.6 \times \log_2 3} \times 100 = \frac{2.42}{2.54} \times 100$$

Hence, code efficiency, $\eta_{\text{code}} = \mathbf{95.3\%}$ **Ans.**

SOLVED EXAMPLE 5.2.17 Variance Comparison of Huffman Tree(s)

Consider an alphabet of a discrete memoryless source having five different source symbols. Table 5.2.9 gives their respective probabilities, codewords and codeword lengths as computed from Huffman Tree from two different methods (higher and lower placement of combined probability in the reduced list).

Table 5.2.9 Codeword and Codeword Length for Source Symbols

Given s_k, p_k	First Method		Second Method	
	Codeword	Codeword Length, l_k	Codeword	Codeword Length, l_k
s_1, 0.4	0 0	$l_1 = 2$	1	$l_1 = 1$
s_2, 0.2	1 0	$l_2 = 2$	0 1	$l_2 = 2$
s_3, 0.2	1 1	$l_3 = 2$	0 0 0	$l_3 = 3$
s_4, 0.1	0 1 0	$l_4 = 3$	0 0 1 0	$l_4 = 4$
s_5, 0.1	0 1 1	$l_5 = 3$	0 0 1 1	$l_5 = 4$

(a) Compute the average codeword length(s) of the source encoder from the two methods.
(b) Determine the respective variance of the average codeword length(s).
(c) Comment on the result and suggest which method should be preferred.

Solution

(a) To compute the average codeword length(s) of the source encoder.

We know that the average codeword length, L_{avg} of the source encoder for a discrete memoryless source having five different source symbols is given as

$$L_{avg} = \sum_{k=1}^{K} p_k l_k = p_1 l_1 + p_2 l_2 + p_3 l_3 + p_4 l_4 + p_5 l_5$$

Using the given data by using <u>first</u> method of generating a Huffman tree, the average codeword length, $L_{avg(1)}$ can be computed as below:

$$\Rightarrow \quad L_{avg(1)} = 0.4 \times 2 + 0.2 \times 2 + 0.2 \times 2 + 0.1 \times 3 + 0.1 \times 3$$

$$\Rightarrow \quad L_{avg(1)} = 0.8 + 0.4 + 0.4 + 0.3 + 0.3$$

Hence, the average codeword length, $L_{avg(1)} = 2.2$ bits

Using the given data by using the <u>second</u> method of generating Huffman Tree, the average codeword length, $L_{avg(2)}$ can be computed as below:

$$\Rightarrow \quad L_{avg(2)} = 0.4 \times 1 + 0.2 \times 2 + 0.2 \times 3 + 0.1 \times 4 + 0.1 \times 4$$

$$\Rightarrow \quad L_{avg(2)} = 0.4 + 0.4 + 0.6 + 0.4 + 0.4$$

Hence, the average codeword length, $L_{avg(2)} = 2.2$ bits

Therefore, $L_{avg(1)} = L_{avg(2)} = 2.2$ bits **Ans.**

(b) To determine the variance of the average codeword length(s).

We know that the variance of the average codeword length of the source encoder is given as

$$\sigma^2 = \sum_{k=1}^{K} p_k (l_k - L_{avg})^2$$

For a discrete memoryless source having five different source symbols, the variance of the average codeword length of the source encoder is given as

$$\sigma^2 = p_1(l_1 - L_{avg})^2 + p_2(l_2 - L_{avg})^2 + p_3(l_3 - L_{avg})^2 + p_4(l_4 - L_{avg})^2 + p_5(l_5 - L_{avg})^2$$

Using the given data by using the <u>first</u> method of generating a Huffman tree, the variance of the average codeword length of the source encoder, σ_I^2 can be computed as below:

$$\sigma_I^2 = 0.4(2-2)^2 + 0.2(2-2.2)^2 + 0.2(2-2.2)^2 + 0.1(3-2.2)^2 + 0.1(3-2.2)^2$$

$$\Rightarrow \quad \sigma_I^2 = 0.4 \times 0.04 + 0.2 \times 0.04 + 0.2 \times 0.04 + 0.1 \times 0.64 + 0.1 \times 0.64$$

$$\Rightarrow \quad \sigma_I^2 = 0.016 + 0.008 + 0.08 + 0.064 + 0.064$$

$$\Rightarrow \quad \boldsymbol{\sigma_I^2 = 0.160}$$ **Ans.**

Using the given data by using the <u>second</u> method of generating a Huffman tree, the variance of the average codeword length of the source encoder, σ_{II}^2 can be computed as below:

$$\sigma_{II}^2 = 0.4(1-2.2)^2 + 0.2(2-2.2)^2 + 0.2(3-2.2)^2 + 0.1(4-2.2)^2 + 0.1(4-2.2)^2$$

$$\Rightarrow \quad \sigma_{II}^2 = 0.4 \times 1.44 + 0.2 \times 1.44 + 0.2 \times 0.64 + 0.1 \times 3.24 + 0.1 \times 3.24$$

$$\Rightarrow \quad \sigma_{II}^2 = 0.576 + 0.576 + 0.128 + 0.324 + 0.324$$

$$\Rightarrow \quad \sigma_{II}^2 = 1.928$$ **Ans.**

(c) To comment on the result and suggest which method should be preferred.

From the above results, it is seen that

$$\sigma_{\mathrm{I}}^2 < \sigma_{\mathrm{II}}^2$$

It means that when a combined probability of the two lowermost symbols is placed as high as possible, the resulting Huffman source code has a significantly smaller value of variance as compared to when it is placed as low as possible.

On the basis of this, it is recommended to prefer the first method over the second method of generating the Huffman source code.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 5.2.18 For a third-order extension, the binary Huffman source code for a discrete memoryless source generates two symbols with probability 0.8 and 0.2 respectively. How many possible messages are available? Find the probabilities of all new message symbols.

5.2.3 Lempel–Ziv Code

Recall

A Huffman encoder takes a block of input characters with fixed length and produces a block of output bits of variable length. It is a fixed-to-variable length code. The design of the Huffman code is optimal (for a fixed block length) assuming that the source statistics are known *apriori*. The basic idea in Huffman coding is to assign short codewords to those input blocks with high probabilities and long codewords to those with low probabilities.

What is Lempel–Ziv Code?

The *Lempel–Ziv code* (LZ code) uses fixed-length codes to represent a variable number of source symbols. It is sometimes known as the Lempel–Ziv algorithm. It does not require knowledge of a probabilistic model of the source. The Lempel–Ziv coding technique is independent of the source statistics in contrast to the Huffman source coding technique. The Lempel–Ziv coding algorithm generates a fixed-length code, whereas the Huffman code is a variable length code.

In other words, we can say that Lempel-Ziv is a variable-to-fixed length code. The Lempel-Ziv code is not designed for any particular source but for a large class of sources. Surprisingly, for any fixed stationary and ergodic source, the Lempel–Ziv algorithm performs just as well as if it was designed for that source. Mainly for this reason, the Lempel–Ziv code is the most widely used technique for lossless file compression.

Algorithm for LZ Code

The Lempel–Ziv algorithm adaptively builds a codebook from the fixed-length source data stream. Encoding is accomplished by parsing the source data stream into segments that are the shortest strings not encountered previously. The LZ coding is done by writing the location of the prefix in binary followed by new last digit. If the parsing gives N strings for a given n-long sequence then $\log_2 N$ (rounded to the next integer) will be the number of binits required for describing prefix. One binit will be required to describe a new symbol, also called the *innovation symbol*.

Decoding of LZ Code

The decoding process is done in a reverse manner. The prefix is used as pointer to the root string and then the innovation symbol is appended to it. The coding efficiency in LZ coding begins to appear at a later part of the string when longer data sequence is taken and the data shows some part of redundancy within it in the form of repetition. LZ coding is useful where *apriori* probabilities are not known and/or where data shows sort of repetition or redundancy in the digit string.

The Lempel–Ziv coding is simpler to implement than Huffman source coding and is intrinsically adaptive. It is suitable for synchronous transmission, and is now the standard algorithm for data compaction and file compression. Lempel–Ziv coding, when applied to ordinary English text, achieves a data compaction efficiency of approximately 55%, as compared to 43% only achieved with Huffman source coding. **Application**

The application of above-mentioned algorithm specified for LZ source-encoding technique is illustrated with the help of the following example:

SOLVED EXAMPLE 5.2.19 **Illustration of LZ Coding Technique**

Consider a data stream of 101010101010101010101010…… (10 repeated 18 times) which is required to be coded using the Lempel–Ziv coding algorithm. Illustrate the process of the LZ encoding algorithm.

Solution Assume that the binary symbols 1 and 0 are already stored in that order in the code book and their numerical position is 1 and 2 respectively in the code book.

The process of parsing begins from the left.

Step 1: With symbols 1 and 0 already stored, the first shortest string of the data stream encountered for the first time and not seen before is 10. That means,
Strings stored so far: 1, 0, 10
Remaining data stream yet to be parsed: 1010101010101010101010….

Step 2: The next shortest string of the data stream encountered for the second time and not seen before is 101. That means,
Strings stored so far: 1, 0, 10, 101
Remaining data stream yet to be parsed: 010101010101010…..

Step 3: The next shortest string of the data stream encountered for the third time and not seen before is 01. That means,
Strings stored so far: 1, 0, 10, 101, 01
Remaining data stream yet to be parsed: 0101010101010…..

Step 4: The next shortest string of the data stream encountered for the fourth time and not seen before is 010. That means,
Strings stored so far: 1, 0, 10, 101, 01, 010
Remaining data stream yet to be parsed: 1010101010…..

Step 5: The next shortest string of the data stream encountered for the fifth time and not seen before is 1010. That means,
Strings stored so far: 1, 0, 10, 101, 01, 010, 1010
Remaining data stream yet to be parsed: 101010…..

Step 6: The next shortest string of the data stream encountered for the sixth time and not seen before is 10101. That means,
Strings stored so far: 1, 0, 10, 101, 01, 010, 1010, 10101
Remaining data stream yet to be parsed: 0…..

And so on ….. until the given data stream has been completely parsed.

Thus, the code book of the binary stream will be 1, 0, 10, 101, 01, 010, 1010, 10101, …..

Self-Assessment Exercise linked to LO 5.2

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/115

Q5.2.1 Consider an alphabet of a discrete memoryless source generating 8 source symbols with probabilities as 0.02, 0.03, 0.05, 0.07, 0.10, 0.10, 0.15, 0.48. Design Shannon–Fano source codeword for each symbol.

Q5.2.2 A discrete memoryless source generates five source symbols with their respective probabilities as 0.15, 0.1, 0.18, 0.17, 0.4. Create Shannon–Fano source codeword for each symbol and calculate the efficiency of the code.

Q5.2.3 Mention the salient features and key advantages of Huffman source-coding technique.

Q5.2.4 A discrete memoryless source generates two symbols with probability 0.8 and 0.2 respectively. Design a binary Huffman source code. Determine the average codeword length, and the code efficiency.

Q5.2.5 For a second-order extension, the binary Huffman source code for a discrete memoryless source generating two symbols with probability 0.2 and 0.8 respectively, compute the probabilities of new message symbols.

Q5.2.6 In general, Huffman coding results in equal or better code efficiency compared to Shannon–Fano coding. Evaluate it for five source messages having probabilities as 0.4, 0.15, 0.15, 0.15, 0.15.

Q5.2.7 Compare and contrast the advantages of Shannon–Fano codes and Huffman source codes for the similar message source.

Q5.2.8 Show how parsing is done in Lempel–Ziv coding for a data stream which is 18 repeating '10' starting with 1. If the number of prefixes used is 8, show how this data is encoded.

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 5.2: *Discuss source-coding techniques including Shannon–Fano, Huffman, and Lempel–Ziv.*

Mid-Chapter Check

So far you have learnt the following:

- Basics of Source Encoding and its Parameters
- Classification of Source Codes and Source-Coding Theorem
- Shannon–Fano, Huffman, and Lempel–Ziv Source Codes

Therefore, you are now skilled to complete the following tasks:

MQ5.1 Classify source codes based on algorithms adopted and highlight the features of each one of them.

MQ5.2 Describe the desirable properties of instantaneous codes generated by entropy coding.

MQ5.3 How is the average codeword length related with the source entropy, as stipulated by the source-coding theorem? Specify the condition to obtain optimum source code.

MQ5.4 Consider an alphabet of a discrete memoryless source having eight source symbols with their respective probabilities as given below:

$$\{s_k\} = \{s_1 \quad s_2 \quad s_3 \quad s_4 \quad s_5 \quad s_6 \quad s_7 \quad s_8\}$$

$$\{p_k\} = \{0.02 \quad 0.03 \quad 0.05 \quad 0.07 \quad 0.10 \quad 0.10 \quad 0.15 \quad 0.48\}$$

Evaluate the code efficiency of a simple binary-encoding technique used for the given data and show that it is only 77%.

LO 5.1

MQ5.5 A source generates 8 source symbols with probabilities as 0.02, 0.03, 0.05, 0.07, 0.10, 0.10, 0.15, 0.48. The lengths of codewords designed using Shannon–Fano source-coding techniques are 1, 3, 3, 4, 4, 4, 5, 5 respectively. Determine the average codeword length. If the pairs of the symbols are encoded rather than the individual symbol then estimate the new coding efficiency.

MQ5.6 Are there any drawbacks of the Huffman source code? If yes, list them.

MQ5.7 Determine the transmission bit rate of a fixed-length code and Huffman code for a symbol rate of 1000 symbols per second, if the original source produces six symbols per second with unequal probability and the average length of the Huffman source code is 2.45 bits per symbol.

MQ5.8 For a second-order extension, the binary Huffman source code for a discrete memoryless source generating two symbols with probabilities 0.2 and 0.8 respectively, find the probabilities of new message symbols.

MQ5.9 Design a binary Huffman source code for a discrete memoryless source of three statistically independent symbols with probabilities 0.02, 0.08, and 0.90 respectively. Determine the average codeword length and the efficiency of the first-order binary Huffman source code.

MQ5.10 The binary n^{th}-order extension source code can significantly improve the coding efficiency of Huffman source-encoding technique. Where does it find applications?

LO 5.2

Sure of what you have learnt so far? For answers, scan the QR code

OR

visit http://qrcode.flipick.com/index.php/116

Be ready for the next sections!

LO 5.3

5.3 ERROR-CONTROL CHANNEL CODING

History of Error Control Channel Coding

Pioneers of coding theory are Claude Shannon (1916–2001) and Richard Hamming (1915–1998) who were colleagues at Bell Labs. Hocquenghem in 1959 and independently Bose and Ray-Chaudhuri in 1960 were responsible for a class of linear block codes known as BCH codes. Irving Reed and Gustave Solomon followed with a set of cyclic BCH codes with an efficient decoding algorithm are well-suited for detecting and correcting burst errors and erasures in widespread applications. Recall Shannon's channel coding theorem that states that if the code transmission rate is less than the maximum channel capacity, it is possible to design an error-control code with almost error-free information transmission. Hamming codes are used to add redundancy to data which aid the detection and correction of errors.

Essence of Channel Coding

The capacity of a binary channel is increased by adding extra bits to information data. This improves the quality of digital data. The process of adding redundant bits is known as *channel coding*. In digital communications, errors can occur due to a lot of reasons such as noisy channel, scratches/dust/fingerprints on CD or DVD, power surge in electronic circuits, etc. It is often desirable to detect and correct those errors. If no additional information is added to the original bit pattern, errors can turn it into another valid bit pattern. Therefore, redundancy is used in error-correcting schemes. By adding redundancy, a lot of bit patterns will become invalid. A good channel coding scheme will make an invalid bit pattern caused by errors to be closer to one of the valid bit patterns than others. In general, *channel coding,* also known as e*rror control channel coding*, is the process of encoding discrete information in a form suitable for transmission, with an objective of enhanced reliability of digital communication.

It is essential to develop and implement appropriate error control and coding procedures in data communications in order to achieve low *Bit Error Rate* (BER) after transmission over a noisy bandlimited channel. We know that data transmission errors occur due to electrical interference from natural sources as well as from human-made sources. Rather, data transmission errors do occur in any type of transmitting medium. The occurrence of errors in data transmission is almost inevitable.

In practice, for a relatively noisy channel, the *probability of error* may have a value as high as 10^{-2}. It means that on the average, at least one bit out of every one hundred bits transmitted are received in error. For most of the applications using digital communications, such a low level of reliable communication is not acceptable. In fact, probability of error on the order of 10^{-6} or even lower is quite often a necessary requirement. We can achieve such a high level of performance with effective error-control coding techniques. It is effective in combating independent random transmission errors over a noisy channel.

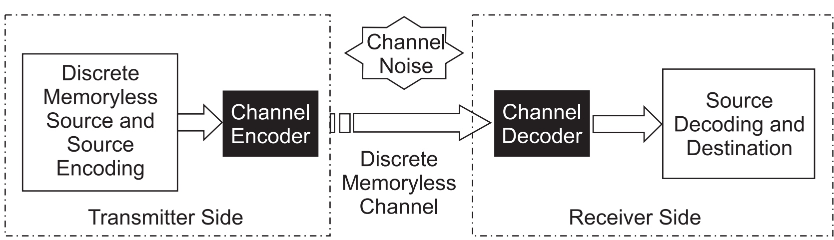


Figure 5.3.1 A Digital Communication System with Channel Encoder and Decoder

- The *channel encoder* introduces extra error-control bits (in the prescribed manner) in the source-encoded input data sequence. The receiver receives the encoded data with transmission errors.
- The *channel decoder* exploits the extra bits to detect/correct any transmission errors and reconstruct the original source data sequence.

Channel Encoder and Decoder

The integrity of received data is a critical consideration in the design of digital communication systems. Many applications require the absolute validity of the received message, allowing no room for errors encountered during transmission. Although the extra error-control bits themselves carry no information, they make it possible for the receiver to detect and/or correct some of the errors caused by the communication channel in the transmitted data bits. Thus, channel coding provides excellent BER performance at low signal-to-noise ratio values. When the encoded data-transmission rate remains the same as in the uncoded system, the transmission accuracy for the coded system is higher. This results in the *coding gain* which translates to higher transmission accuracy, higher power and spectral efficiency.[4]

Significance of Channel Coding

Coding gain is obtained at some price because the introduction of extra bits in channel coding process causes channel coding to consume additional frequency bandwidth during transmission.

IMPORTANT!

- We begin with a brief introduction of types of errors and types of error control codes.
- In this section, we discuss linear block codes including Hamming codes, cyclic codes, BCH codes, Hadamard codes, and LDPC codes.
- Hamming code is the first class of (n, k) linear block code devised for error control. Important parameters and procedure to calculate Hamming check bits and decoding of Hamming code are explained with suitable examples. We analyze Hamming weight, Hamming distance, and error detecting and correcting capability of Hamming code. Then, we describe briefly several attributes of cyclic codes, BCH codes, Hadamard codes, and LDPC codes.

What We Discuss Here ...

5.3.1 Types of Errors and Error-Control Codes

A system's noise environment can cause errors in the received message. Properties of these errors depend upon the noise characteristics of the channel. Errors which are usually encountered fall into three broad categories:

Three Types of Errors

- ***Random Errors:*** Random errors occur in the channel when individual bits in the transmitted message are corrupted by noise. Random errors are generally caused by thermal noise in communications channels. The bit error probabilities are nearly independent of each other.
- ***Burst Errors:*** Burst errors happen in the channel when errors occur continuously in time. Burst errors can be caused by fading in a communications channel. For some codes burst errors are difficult to correct. However, block codes including Reed–Solomon codes handle burst errors very efficiently.
- ***Impulse Errors:*** Impulse errors can cause catastrophic failures in the communications system that are so severe they may be unrecognizable by FEC using present-day coding schemes. In general, all coding systems fail to reconstruct the message in the presence of catastrophic errors. However, certain codes like the Reed–Solomon codes can detect the

[4] For a given data rate, error-control coding can reduce the probability of error, or reduce the required S/N ratio to achieve the specified probability of error. However, this increases the transmission bandwidth and complexity of channel decoder.

presence of a catastrophic error by examining the received message. This is very useful in system design because the unrecoverable message can at least be flagged at the decoder.

Error detection and correction codes use redundancy to perform their function. This means that extra code symbols are added to the transmitted message to provide the necessary detection and correction information. Basically, there are three types of errors which occur during transmission of binary data in digital communications.

- ***Single Bit Errors*:** Only one bit is in error within a given data sequence. Single bit errors affect only one character within a message.
- ***Multiple Bit Errors*:** Two or more bits are in error within a given data sequence. Multiple bit errors can affect one or more characters within a message.
- ***Burst Errors*:** Two or more consecutive bits are in error within a given data sequence. Burst errors generally affect more characters within a message. They are more difficult to detect and even more difficult to correct than single bit errors.

Types of Error-Control Codes

The task of channel coding is to encode the information sent over a communication channel in such a way that in the presence of channel noise, errors can be detected and/or corrected. We distinguish between two coding methods:

- *Backward Error Correction* (*BEC*) requires only error detection. If an error is detected, the sender is requested to retransmit the message. While this method is simple and sets lower requirements on the code's error-correcting properties, but it requires duplex communication and causes undesirable delays in transmission.
- *Forward Error Correction* (*FEC*) requires that the decoder should also be capable of correcting a certain number of errors, i.e., it should be capable of locating the positions where the errors occurred. Since FEC codes require only simplex communication, they are especially attractive in wireless communication systems, helping to improve the energy efficiency of the system. The power of FEC is that the system can, without retransmissions, find and correct limited errors caused by communications channel. Redundancy is used by all FEC codes to perform error detection and correction. FEC codes allow a receiver in the system to perform it without requesting a retransmission. Here we deal with binary FEC codes only.

Depending on types of errors need to be detected and corrected in digital communication, there are several categories of error-control codes such as *linear block codes, convolution codes*, and *burst-error correction codes*.

- A *linear block code* is a set of codewords that has a well-defined mathematical structure, where each codeword is a binary sequence of a fixed number of bits. It involves encoding a block of information sequence bits into another block of encoded bits with addition of error control bits. Examples of linear block codes include Hamming, Cyclic, BCH, Hadamard and LDPC codes are discussed in this section.
- *Convolution code* is an alternative to linear block codes, where n-symbol output codeword at any given time unit is derived not only by present k-bit blocks of input data but also m-bit previous information block. Thus, the error control bits are continuously dispersed within information data bits. Convolution codes and their decoding algorithms are covered in Section 5.4.
- *Interleaving* is the process of dispersing the *burst errors* into multiple individual errors which can then be detected and corrected by error control coding. Burst-error correction codes such as Block and Convolution Interleaving, RS codes, and Turbo codes are covered in Section 5.5.

Design Objectives of Channel Codes

Designing a channel code is always a tradeoff between energy efficiency and bandwidth efficiency. Codes with lower rate (i.e., more redundancy) can usually correct more errors. If more errors can be corrected, the communication system can operate with a lower transmit power, transmit over longer distances, tolerate more interference, use smaller antennas and transmit at a higher data rate. These properties make the code energy efficient. On the other hand, low-rate codes have a large overhead and are hence heavier on bandwidth consumption. Also, decoding complexity grows exponentially with code length, and long (low-rate) codes set high computational requirements to conventional decoders. According to Viterbi in convolution encoder, this is the central problem of channel coding: encoding is easy but decoding is hard.

Linear Block Codes

Linear block code involves encoding a block of source information bits into another block of bits with addition of error-control bits to combat channel errors induced during transmission. An (n, k) linear block code encodes k information data bits into n-bit codeword.

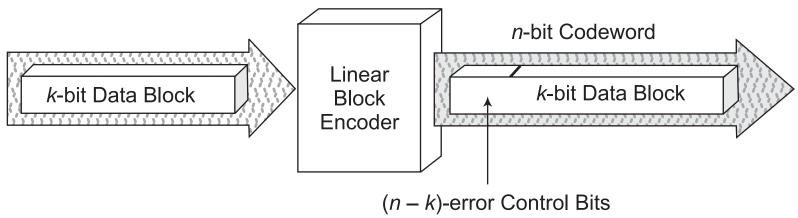


Figure 5.3.2 A Simple Linear Block Code Operation

Interpretation

The information binary data sequence is divided into sequential message blocks of fixed length in block coding. Each message block consists of k-information bits. Thus, there may be 2^k distinct messages. The error control bits $(n - k)$ are derived from a block of k- information bits, where n is total number of bits in encoded data block $(n > k)$. The channel encoder adds $(n - k)$ error control bits to each *k-bit* information block.

Thus, in linear block code, each valid codeword reproduces the original k data bits and adds to them $(n - k)$ check bits to form n-bit codeword. In a nutshell, we can say that

- Number of possible codewords = 2^n
- Number of valid codewords = 2^k
- Redundancy of code (ratio of redundant bits and data bits) = $\dfrac{n-k}{k}$

Hence, the encoding process in linear block code process preserves the k data bits and adds $m = (n - k)$ check bits.

Essential Features of Linear Block Codes

The general considerations in design of a linear block code include that it should be relatively easy to encode and decode. A code should require minimal memory and minimal processing time. Moreover, the number of error-control bits should be small so as to reduce the required transmitted bandwidth, and large enough so as to reduce the error rate. So there is always a trade-off between performance parameters in terms of bandwidth and error rate to select the number of error control bits.

A block of n encoded bits $(n > k)$ at any instance depends only on the block of data consisting of k-information bits present at that time. So there is no built-in memory.

- When the k information data bits appears at the beginning of a codeword, the code is called a *systematic code*.

- The *n-bit* data block is called a *codeword.*

The number of error-control bits depends on the size of the block of information data bits and the error-control capabilities required.

How is a Linear Block Encoder Realized?

A (n, k) linear block code represents k information bits encoded into n-bit codeword ($n > k$) by adding ($n - k$) error-control bits, derived from a block of k information bits by using well-defined algorithm. The position of error-control bits within a codeword is arbitrary. They can be either appended at the either side of the information data bits (thereby keeping the information bits together, known as *systematic coding*), or dispersed within the information data bits (known as *non-systematic coding*). In a block code, the n-bit output codeword depends only on the corresponding k-bit input information. It means that the encoder is memoryless and, hence, can be implemented with a *combinational* logic circuit.

Data Rate of Linear Block Encoder

The data rate at which the linear block encoder produces bits is given by

$$R_o = \left(\frac{n}{k}\right) R_s$$

where R_o is the channel data rate at the output of block encoder and R_s is the information data rate of the source. Since $n > k$, the channel data rate is greater than the information data rate ($R_0 > R_s$).

IMPORTANT!

A desirable property for linear block codes is linearity, which means that its codeword satisfy the condition that the sum of any two codewords gives another codeword.

Important Definitions

- ***Bit Error Rate*** **(BER)** is the theoretical expected value of the error performance which can be estimated mathematically. For example, a specified BER of 10^{-6} means that it is expected to have one bit in error for every 1,000,000 bits transmitted.
- ***Error detection*** is the process of monitoring data transmission and detecting when errors in data have occurred. Error-detection technique can mainly indicate when an error has occurred. However, it does not identify or correct bits in error, if any.
- A ***parity bit*** is an additional bit added to the bit pattern to make sure that the total number of 1's is even (even parity) or odd (odd parity). For example, the information bit pattern is 01001100 and even parity is used. Since the number of 1's in the original bit pattern is 3, a '1' is added at the end to give the transmitted (encoded) bit pattern of 010011001. The decoder counts the number of 1s (normally done by Exclusive OR) in the bit stream to determine if an error has occurred. A single parity bit can detect (but not correct) any odd number of errors. The parity-check scheme is the simplest form of redundancy for detecting errors in digital binary messages.
- ***Redundancy checking*** is the most common type of error-detection techniques. It involves adding extra bits to the data for the exclusive purpose of detecting transmission data errors. Examples of error-detection techniques based on redundancy checks are Vertical Redundancy Check (VRC) (also called odd/even character parity check), Longitudinal Redundancy Check (LRC), checksum, and Cyclic Redundancy Check (CRC).
- ***Code rate*** is defined as the ratio of number of information data bits (uncoded) and number of encoded data bits. That is,

  Code rate, $\boxed{r = \frac{k}{n}}$

  where k is the input information bits and n is the output encoded bits. The code rate is a dimensionless ratio. The code rate is always less than unity because $n > k$ in any error control code. This can be interpreted as the number of information bits entering the error control encoder per transmitted symbol (codeword). Thus, this is a measure of the redundancy within the codeword.

Significance of Code Rate

Code rate is also a measure of how much additional bandwidth is required to transmit encoded data at the same data rate as would have been needed to transmit uncoded data. For example, $r = 1/2$ means the requirement of double bandwidth of an uncoded system to maintain the same data rate.

For a binary code to have a unique codeword assigned to each information block, $k \leq n$ or $r \leq 1$. When $k < n$, $(n - k)$ redundant bits can be added to each k-bit information block to form a codeword to provide the code the power to combat with the errors induced due to channel noise.

- For a fixed value of k-bits, the code rate tends to be zero as the number of redundant bits increases.
- On the contrary, for the case of no coding, there are no error control bits and hence $n = k$ means the code rate $r = 1$.

Thus, we see that the code rate is bounded by $0 \leq r \leq 1$. Similarly, in a binary convolution code, redundant or error-control bits can be added when $k < n$ or $r < 1$. Generally, k and n are small integers and more redundancy is added by increasing m (number of previous information bits stored in memory) while maintaining k and n and hence the code rate r fixed.

Code Rate of Parity Bit Check Scheme

In even/odd parity bit check schemes, the code rate is quite efficient (8/9 in the above example) but its effectiveness is limited. Single parity bit is often used in situations where the likelihood of errors is small and the receiver is able to request retransmission. Sometimes it is used even when retransmission is not possible. Early PCs employ single bit parity scheme.

Advantages of Error Detection and Correction

- Forward Error Correction (FEC) enables a system to achieve a high degree of data reliability, even with the presence of noise in the communications channel. Data integrity is an important issue in most digital communications systems and in all mass storage systems.
- In systems where improvement using any other means (such as increased transmit power or components that generate less noise) is very costly or impractical, FEC can offer significant error control and performance gains.
- In systems with satisfactory data integrity, designers may be able to implement FEC to reduce the costs of the system without affecting the existing performance. This is accomplished by degrading the performance of the most costly or sensitive element in the system, and then regaining the lost performance with the application of FEC.

IMPORTANT!

In electronic communication systems, the information can be either sent in both directions (telephone channels, some satellite communication channels), or strictly in one direction (broadcast systems, deep space communication systems). Accordingly, there are two basic techniques used for channel error control—Automatic Repeat Request (ARQ) and Forward-Error Correction (FEC).

LET'S RECONFIRM OUR UNDERSTANDING!!

- Distinguish between single-bit errors, multiple-bit errors, and burst errors.
- Give examples of linear block codes and burst error-correction codes.

5.3.2 Hamming Codes

Hamming Code Parameters

If a code uses n bits to provide error protection to k bits of information, it is called a (n, k) block code. *Hamming distance* is a metric used to measure the "closeness" between two bit patterns (codewords). The Hamming distance between two bit patterns is the number of bits that are different. For example, bit patterns 1100 and 0100 differ by one bit (the 1st bit), thus have Hamming distance of one. Two identical bit patterns have Hamming distance of zero. Quite often, the minimum Hamming distance d between any two valid codewords is also included to give a (n, k, d) block code. An example of block codes is Hamming code.

Hamming code is the first class of (n, k) linear block code devised for error control. It is the form of non-systematic code in which the position of error-control bits is arbitrary within a codeword. For any positive integer $m \geq 3$, where m denotes the number of error control bits (also known as check bits), or redundancy, there exists a Hamming code with the following parameters:

- Number of encoded data bits (code length of a codeword), $n = 2^m - 1$
- Number of user data bits (information), $k = 2^m - 1 - m$
- Number of error control or check bits, $m = n - k$
- Minimum hamming distance, $d_{min} = 3$ (number of bits by which two codewords differ).
- Error-correcting capability, $t = 1$

With k bits of data and m bits of redundancy added to it, the length of resulting encoded code has $(k + m)$ bits. Including no error condition as one of the possible states, total $(k + m + 1)$ states must be discoverable by m bits; where m bits can have 2^m different states. Therefore,

$$\boxed{2^m \geq (m + k + 1)}$$

This expression is used to calculate the number of check bits, m. The Hamming code can be applied to data units of any length (k bits) using this relationship.

Note... *The following example illustrates the application of the above expression.*

SOLVED EXAMPLE 5.3.1 **Encoded Bits using Hamming Code**

Using the relationship $2^m \geq (m + k + 1)$, compute the number of encoded bits for a block of data bits having 2, 4, 5, 7, and 16 bits.

Solution The results are tabulated in Table 5.3.1.

Table 5.3.1 Hamming Code for k = 2, 4, 5, 7, 16 bits

Number of Data Bits, k	Number of Check Bits, m	Encoded Bits, n
2	3	5
4	3	7
5	4	9
7	4	11
16	5	21

It is observed that as the number of data bits increases, more error-control bits are needed to generate encoded bits as per Hamming code.

SOLVED EXAMPLE 5.3.2 **Percent Increase in Encoded Bits**

Using the relationship $2^m \geq (m + k + 1)$, compute the number of encoded bits and % increase in encoded bits for

(a) single-error correction only

(b) single-error correction and double error detection

Use the block of data bits having 8, 16, 32, 64, 128, and 256 bits and tabulate the results.

Solution We know that the number of encoded bits for single-error correction and double-error detection is one more than that of single-error correction only. The results are tabulated in Table 5.3.2 and Table 5.3.3 respectively.

Table 5.3.2 Increase in Encoded Bits for Single-error Correction only

Number of Data Bits, k	Number of Check Bits, m	Encoded Bits, n	% Increase in Encoded Bits, $(m/n) \times 100$
8	4	12	50
16	5	21	31.25
32	6	38	18.75
64	7	71	10.94
128	8	136	6.25
256	9	265	3.25

It may be noted that a larger block of data will require less transmitted bandwidth for encoded data as compared to that of required for a smaller block of data.

Table 5.3.3 Single-error Correction and Double-error Detection

Number of Data Bits, k	Number of Check Bits, m	Encoded Bits, n	% Increase in Encoded Bits, $(m/n) \times 100$
8	5	13	63.5
16	6	22	37.5
32	7	39	21.875
64	8	72	12.5
128	9	137	7.03
256	10	266	3.91

Conclusion: As expected, the percentage increase in encoded data bits is more for error-correction plus error-detection as compared to that required for error-correction only.

Procedure to Determine Hamming Check Bits

Let us evolve step-by-step procedure to determine Hamming check bits for the given sequence of data. Consider the scenario where we wish to correct single error using the fewest number of parity bits (highest code rate). Each parity bit gives the decoder a parity equation to validate the received code. With 3 parity bits, we have 3 parity equations, which can identify up to 8 (= 2^3) error conditions. One condition identifies "no error", so 7 error conditions would be left to identify up to 7 places of single error. Therefore, we can detect and correct any single error in a 7-bit word. With 3 parity bits, we have 4 bits left for information. Thus this is a (7, 4) block code. In a (7, 4) Hamming code, the parity equations are determined as follow:

- The first parity equation checks bit 4, 5, 6, 7
- The second parity equation checks bit 2, 3, 6, 7
- The third parity equation checks bit 1, 3, 5, 7

This rule is easy to remember. For example, the location 5 has the binary representation of 101, thus appears in equation 1 and 3. By applying this rule, we can tell which bit is wrong by reading the value of the binary combination of the result of the parity equations, with 1 being incorrect and 0 being correct. For example, if equations 1 and 2 are incorrect and the equation 3 is correct, we can tell that the bit at location 6 (110) is wrong. At the encoder, if locations 3, 5, 6, 7 contain the original information and locations 1, 2, 4 contain the parity bits (locations which are a power of 2) then using the first parity equation and bits at location 5, 6, 7, we can calculate the value of the parity bit at the location 4 and so on. For example, (7, 4) Hamming code can be summarized in Table 5.3.4.

Table 5.3.4 (7,4) Hamming Code

Bit number	7	6	5	4	3	2	1
	D_4	D_3	D_2	C_3	D_1	C_2	C_1
Check bit corresponds to C_1	1	0	1	0	1	0	1
Check bit corresponds to C_2	1	1	0	0	1	1	0
Check bit corresponds to C_3	1	1	1	1	0	0	0

General Procedure

Here, D denotes data bits and C denotes check bits. Let us generalize the procedure to determine Hamming check bits in the following steps with appropriate illustrations.

Step 1 Hamming check bits are inserted with the original data bits at positions that are power of 2, that is at $2^0, 2^1, 2^2, ..., 2^{n-k}$. The remaining positions have data bits.

For example, for $k = 8$. Then using the condition $2^m \geq (m + k + 1)$, we obtain $m = 4$. So, total number of encoded bits will be $k + m = 12$. The position of Hamming check bits will be $2^0, 2^1, 2^2, 2^3$ (that is, 1st, 2nd, 4th, and 8th).

Table 5.3.5 depicts such an arrangement of positions of data bits and check bits.

Table 5.3.5 Positions of Hamming Check Bits

Position # of Encoded Data Bits (n = 12)	12	11	10	9	8	7	6	5	4	3	2	1
Bit type (data or check)	D_8	D_7	D_6	D_5	C_4	D_4	D_3	D_2	C_3	D_1	C_2	C_1

Here, D_k (k = 1 to 8) represents data bits, and C_m (m = 1 to 4) represents check bits.

Step 2 The procedure for computing the binary value of Hamming check bits is that each data position which has a value 1 is represented by a binary value equivalent to its position.

Step 3 All of these position values are then XORed together to produce the check bits of Hamming code.

Note... *This is illustrated with the help of the following example.*

SOLVED EXAMPLE 5.3.3 **Computation of Hamming Code**

Let the given data bits are 0 0 1 1 1 0 0 1. Compute the Hamming code.

Solution Given data bits corresponding to $D_8\ D_7\ D_6\ D_5\ D_4\ D_3\ D_2\ D_1$ is 0 0 1 1 1 0 0 1.

We know that for Hamming code, $2^m \geq (m + k + 1)$

For given value of $k = 8$, $2^m \geq (m + 8 + 1) \Rightarrow 2^m \geq (m + 9)$

This condition can be satisfied only if $m = 4$. $[\because\ 16 > 13]$

Therefore, number of Hamming check bits, $m = 4$

Position of check bits C_1, C_2, C_3, C_4 will be 1st, 2nd, 4th, and 8th respectively.

Total number of encoded Hamming-code bits, $n = k + m = 12$

The position of data bits D_8, D_7, D_6, D_5, D_4, D_3, D_2, and D_1 will be the remaining position in 12-bit encoded data, as given in Table 5.3.6.

Table 5.3.6 Positions of Hamming Check Bits

Position #	12	11	10	9	8	7	6	5	4	3	2	1
Binary Equivalent	1100	1011	**1010**	**1001**	1000	0111	0110	0101	0100	**0011**	0010	0001
Data Bit Position	D_8	D_7	D_6	D_5		D_4	D_3	D_2		D_1		
Given Data Bits	0	0	1	1		1	0	0		1		
Check Bit Position					C_4				C_3		C_2	C_1

The Hamming code is computed by XORing the binary equivalent of position values in which the value of the given data bits is 1. That is,

Given data bits having binary value 1 are D_6, D_5, D_4, and D_1 (other data bits has 0 binary value), placed at positions # 10, 9, 7, and 3, respectively. The binary equivalent of position # 10, 9, 7, and 3 are **1010, 1001, 0111,** and **0011.** Therefore,

$$1010 \otimes 1001 \otimes 0111 \otimes 0011 \Rightarrow 0111;\ \text{where } \otimes \text{ represents Ex-OR operation}$$

That is, 0111 corresponds to Hamming code $\mathbf{C_4C_3C_2C_1}$.

Hence, encoded Hamming data corresponding to $D_8D_7D_6D_5\mathbf{C_4}D_4D_3D_2\mathbf{C_3}D_1\mathbf{C_2C_1}$ will be 0 0 1 1 0 1 0 0 1 1 1 1. **Ans.**

Procedure to Decode Hamming Code

At the receiver, all bit position values for data bits as well as Hamming check bits having binary value 1s are XORed. The resultant bit pattern is known as *Syndrome*. Alternatively, all data bit positions with a binary value 1 plus the Hamming code formed by the check bits are XORed since check bits occur at bit positions that are power of 2.

- If Syndrome contains all 0s, no error is detected.
- If Syndrome contains one and only one bit set to 1 then error has occurred in one of the check bits (Hamming code itself). So no correction is required in received decoded data.

- If syndrome contains more than one bit set to 1 then its decimal equivalent value indicates the position of data bit which is in error. This data bit is simply inverted for correction.

This is illustrated in the following example.

SOLVED EXAMPLE 5.3.4 **Detection of Error with Hamming Code**

For a given data 0 0 1 1 1 0 0 1, the transmitted encoded data using Hamming code is 0 0 1 1 0 1 0 0 1 1 1 1. Show that

(a) if the received data is 0 0 1 1 0 1 0 0 1 1 1 1, then there is no error

(b) if the received data is 0 0 1 1 0 1 1 0 1 1 1 1, then there is error in the 6^{th} bit position

Solution

(a) For the given received data 0 0 1 1 0 1 0 0 1 1 1 1, the Hamming code can be deduced using Table 5.3.7.

Table 5.3.7 Deduction of Hamming Code from Received Data

Position #	12	11	10	9	8	7	6	5	4	3	2	1
Binary Equivalent	1100	1011	**1010**	**1001**	1000	**0111**	0110	0101	**0100**	**0011**	**0010**	**0001**
Received Data	0	0	1	1	0	1	0	0	1	1	1	1
Data/Code Bit Position	D_8	D_7	$\mathbf{D_6}$	$\mathbf{D_5}$	C_4	$\mathbf{D_4}$	D_3	D_2	$\mathbf{C_3}$	$\mathbf{D_1}$	$\mathbf{C_2}$	$\mathbf{C_1}$
Hamming Code					**0**				**1**		**1**	**1**

The XOR operation of binary equivalent of those data bit positions having binary value 1 and Hamming code is carried out to determine whether there is an error or not in the received encoded data.

That is, binary equivalent of position # of D_6 (1010), D_5 (1001), D_4 (0111), D_1 (0011), and Hamming code (0111) are XORed.

$$1010 \otimes 1001 \otimes 0111 \otimes 0011 \otimes 0111 \Rightarrow 0000$$

Since the result is 0000, therefore, it shows that there is no error in the received data.

(b) For the given received data 0 0 1 1 0 1 1 0 1 1 1 1, the Hamming code can be deduced using Table 5.3.8.

Table 5.3.8 Deduction of Hamming Code from Received Data

Position #	12	11	10	9	8	7	6	5	4	3	2	1
Binary Equivalent	1100	1011	**1010**	**1001**	1000	**0111**	0110	0101	**0100**	**0011**	**0010**	**0001**
Received Data	0	0	1	1	0	1	**1**	0	1	1	1	1
Data/Code Bit Position	D_8	D_7	$\mathbf{D_6}$	$\mathbf{D_5}$	C_4	$\mathbf{D_4}$	$\mathbf{D_3}$	D_2	$\mathbf{C_3}$	$\mathbf{D_1}$	$\mathbf{C_2}$	$\mathbf{C_1}$
Hamming Code					**0**				**1**		**1**	**1**

The XOR operation of the binary equivalent of those data bit positions having binary value 1 and Hamming code is carried out to determine whether there is an error or not in the received encoded data.

That is, binary equivalent of the position # of D_6 (1010), D_5 (1001), D_4 (0111), D_3(0110), D_1(0011), and Hamming code (0111) are XORed.

$$1010 \otimes 1001 \otimes 0111 \otimes 0110 \otimes 0011 \otimes 0111 \Rightarrow 0110$$

Since the result is 0110 which corresponds to decimal equivalent value of 6, therefore, it shows that there is an error in the 6th position of received data.

To obtain the correct data, the binary value of the received data at 6th position is inverted. Therefore, the corrected data is 0 0 1 1 1 0 0 1 which is same as given data.

Hence, Hamming code can detect as well as correct errors.

Hamming Weight and Hamming Distance

The *Hamming weight* of a binary n-bit encoded codeword $\boldsymbol{c}$ is defined as the number of non-zero elements (i.e., 1s) present in it and is denoted by $w(c)$. For example, the Hamming weight of a codeword $c = \{1101101\}$ is 5.

By definition, *Hamming distance* is the number of bits by which two codewords (n-bit binary sequence) c_1 and c_2 having same number of bits differ and is denoted by $d(c_1, c_2)$. For example, in the codewords $c_1 = \{1101101\}$ and $c_2 = \{0111001\}$, the bits are different in the zeroth, second, and fourth places. Therefore, the Hamming distance is 3.

Now, the Hamming weight of the sum of two codewords $c_1 = \{0111001\}$ and $c_2 = \{0111001\}$ is given as

$$w(c_1 + c_2) = w[\{1101101\} + \{0111001\}] = w[\{1010100\}] = 3 \text{ (i.e., the number of 1s).}$$

In this case, the Hamming distance is the same as the Hamming weight, i.e., $d(c_1, c_2) = w(c_1 + c_2)$.

Triangle Inequality

It may be shown that the Hamming distance satisfies the *triangle inequality*, i.e., $d(c_1, c_2) + d(c_2, c_3) \geq d(c_1, c_3)$, where c_1, c_2, and c_3 are three distinct n-bit codewords.

Minimum Hamming Distance

The *minimum Hamming distance*, $d_{\min}$ of a block code is the smallest distance between two codewords. Mathematically, it is expressed as

$$d_{\min} = \min\{d(c_1, c_2) : c_1, c_2 \in c, c_1 \neq c_2\};$$

where $\boldsymbol{c}$ is the block code.

In general, for a block code consisting of the codewords $c_1, c_2, c_3, ..., c_n$, the minimum Hamming distance $d_{\min}$ of the code is given by

$$d_{\min} = \min_{i+j}\left[d(c_i, c_j)\right]$$

Thus, we see that the minimum distance of a linear block code is equal to the minimum weight of its non-zero codewords, i.e., $d_{\min} = w_{\min}$. It implies that for a block code c with minimum Hamming distance $d_{\min}$, there cannot be any codeword with Hamming weight less than $d_{\min}$.

Error Detecting and Correcting Capability

The *random-error-detecting capability* of a block code is usually determined with minimum Hamming distance as $(d_{\min} - 1)$.

Let a codeword c_1 be transmitted over a noisy channel and be received as a codeword r_1 such that $r_1 \neq c_1$ due to some error pattern introduced by the channel noise. If the error pattern is less than or equal to $(d_{min} - 1)$ then it cannot change one codeword to another valid codeword within c. Hence, the error will be definitely detected.

The *error correcting capability* of a block code is also determined by its minimum Hamming distance d_{min} which may be either even or odd. Let us consider that the code $\boldsymbol{c}$ can correct all error patterns of t or fewer bits.

- If a code satisfies the condition $d_{min} \geq (2t + 1)$, where t is a given positive integer then the code can correct all bit errors up to and including error of t bits.
- Consequently, if a code satisfies the condition $d_{min} \geq 2t$ then it has the ability of correcting all errors up to $\leq (t - 1)$ bits, and errors of t bits can be detected, but not, in general, corrected.

Thus, maximum number of guaranteed correctable error per codeword satisfies the condition $t_{max} = \left\lfloor \dfrac{d_{min} - 1}{2} \right\rfloor$; where $\lfloor x \rfloor$ means largest integer not exceeding x. For example, $\lfloor 5.4 \rfloor = 5$.

Maximum number of guaranteed detectable error per codeword is then given as $t_{max} = (d_{min} - 1)$. The parameter $(d_{min} - 1)/2$ is known as *random-error-correcting capability* of a block code, referred as t error-correcting code and is usually capable of correcting many error patterns of $(t + 1)$ or more errors. In fact, it may be shown that a t error-correcting (n, k) linear block code is capable of correcting a total $2^{(n-k)}$ error patterns including having t or fewer errors.

Significance of Hamming Distance

By ensuring required Hamming distance, a code can always detect a double-bit error and correct a single-bit error. If an invalid codeword is received then the valid codeword that is closest to it (having minimum Hamming distance) is selected. Thus, the importance of Hamming distance is the requirement of a unique valid codeword at a minimum Hamming distance from each invalid codeword.[5]

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 5.3.5 Determine Hamming code for the codeword 1 0 0 1 1 0 1 following step-by-step procedure. Verify that received bit pattern has no error.

Ex 5.3.6 For the codeword bit pattern 1 0 0 1 1 0 1, generate a Hamming code to detect and correct for single-bit errors assuming each codeword contains a 7-bit data field. If the received bit pattern is 0 0 0 1 1 1 0 0 1 0 1, then find the position of the bit at which the error has occurred. Write the corrected bit pattern.

Ex 5.3.7 For the codewords {000000}, {101001}, {011010}, {110011}, {110100}, {101110}, {000111}, find the Hamming weight and minimum Hamming distance.

Ex 5.3.8 For a linear block code, the minimum Hamming distance is given as 5. How many errors can this code detect and correct?

[5] In a linear block code, error-control bits added with the information bits increase the separation or distance between the codewords. The concept of distance between different codewords, and, in particular, the minimum distance within a code, is fundamental to error-control codes. This parameter determines the random error-detecting and correcting capabilities of a block code. The ability of a linear block code to correct errors is a function of Hamming distance.

5.3.3 Cyclic Codes

Recall

We know that block codes use algebraic properties to encode and decode blocks of data bits or symbols. The codewords are generated in a systematic form in such a way that the original k number of source information data bits are retained as it is. The $(n - k)$ error control bits are either appended or pre-appended to information data bits.

What are Cyclic Codes?

Cyclic codes are widely used in data communication because their structure makes encoder and decoder circuitry simple. Hill (1986) defines Code C as cyclic (n, k) code if C is a linear code of length n over a finite field and any cyclic shift of a codeword is also a codeword. Cyclic code is a subclass of linear block codes. Any cyclic shift of a codeword results in another valid codeword. This feature allows easy implementation of cyclic encoders with linear sequential circuits by employing shift registers and feedback connections. Random error as well as burst error correction is possible to the large extent.

Representation of Cyclic Codes

Let there be a block of k arbitrary information bits represented by $m_0, m_1, ..., m_{k-1}$. This constitutes 2^k distinct information bits be applied to a linear block encoder, producing an n-bit code. Let the elements of n-bit code are denoted by $c_0, c_1, ..., c_{n-1}$. And the $(n - k)$ parity bits in the codeword are denoted by $b_0, b_1, ..., b_{n-k-1}$. The elements of n-bit code can then be represented in the form of algebraic structure as

$$c_i = \begin{cases} b_i & ; i = 0, 1, ..., n-k-1 \\ m_{i+k-n} & ; i = n-k, n-k+1, ..., n-1 \end{cases}$$

For cyclic codes, if the n-bit sequence $c = (c_0, c_1, ..., c_{n-1})$ is a valid codeword then another n-bit sequence $(c_{n-1}, c_0, c_1 ..., c_{n-2})$, which is formed by cyclically shifting $c = (c_0, c_1, ..., c_{n-1})$ one place to the right, is also a valid codeword.

Cyclic Redundancy Check (CRC)

Codes generated by a polynomial are called *cyclic codes*, and the resultant block codes are called *cyclic redundancy check* (*CRC*) codes.

The CRC code is considered a systematic code, and are quite often described as (n, k) cyclic codes where n is number of transmitted encoded data bits, k is number of information data bits, and $(n - k)$ redundant bits appended at the end of information data bits are known as CRC bits. It is an error-detecting code whose algorithm is based on cyclic codes. Mathematically, the CRC can be expressed as

$$\frac{G(x)}{P(x)} = Q(x) + R(x)$$

where $G(x)$ is data message polynomial, $P(x)$ is generator polynomial, $Q(x)$ is quotient, and $R(x)$ is remainder. It is essential that the generator polynomial $P(x)$ must be a prime number. The data message polynomial $G(x)$ is divided by $P(x)$ using modulo-2 division where $R(x)$ is derived from an XOR operation. The remainder $R(x)$ is CRC bits which is appended to the message. The CRC must have exactly one bit less than the divisor. At receiver, the received data sequence is divided by the same generator polynomial, $P(x)$ which was used at the transmitting end. If the remainder is zero then the transmission has no errors. If it is non-zero, then the transmission has some errors.

Encoder for Systematic Cyclic Codes

This class of codes can be easily encoded and decoded using Linear Feedback Shift Registers (LFSRs). A cyclic error-correcting code takes a fixed-length input (k bits) and produces a fixed length check code $(n - k)$ bits. For the encoder, the k data bits are treated as input to produce an $(n - k)$ code of check bits in the shift register. The shift register implementation of CRC error-detecting code takes an input of arbitrary length and produces a fixed-length CRC check code. The shift-register implementation of a cyclic error-correcting code takes a fixed length input k data bits and produces a fixed length check code of $(n - k)$ bits.

A simple operation of an algebraic cyclic coding technique is depicted in Figure 5.3.3.

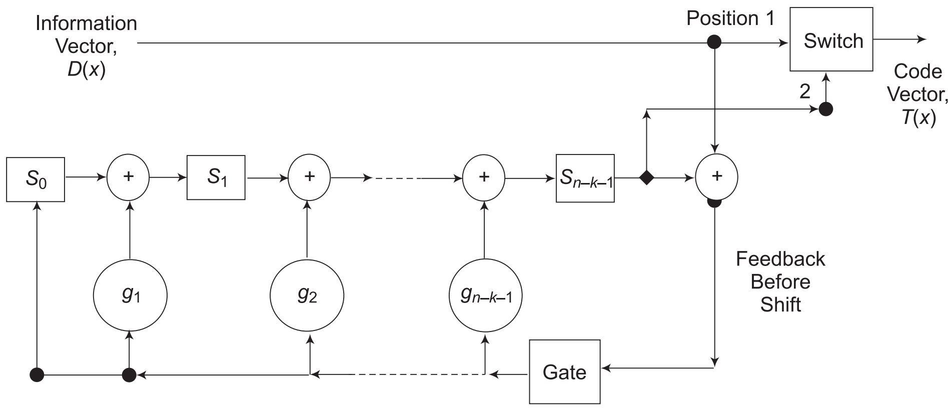


Figure 5.3.3 Encoding of (n, k) Algebraic Cyclic Code

Procedure for Encoding Operation

The blocks $S_0, S_1, ..., S_{n-k-1}$ represent a bank of shift registers consisting of flip-flops, each followed by a modulo-2 adder. The $g_1, g_2, ..., g_{n-k-1}$ denote a closed path when $g_i = 1$ and an open path when $g_i = 0$, where $i = 1, 2, ..., n - k - 1$. The necessary division operation is obtained by the dividing circuit comprising of feedback shift registers and modulo-2 adders. At the occurrence of a clock pulse, the inputs to the shift registers are moved into it and appear at the end of the clock pulse. The encoding operation is carried out in following steps:

Step 1: Initialize the registers.

Step 2: The switch is kept in position 1 and the Gate is turned on.

Step 3: The k information bits are shifted into the register one by one and sent over the communication channel.

Step 4: At the end of the last information bit, the register contains the check bits.

Step 5: The switch is now moved to position 2 and the Gate is turned off.

Step 6: The check bits contained in the register are now sent over the channel. Thus, the codeword is generated and transmitted over the communication channel.

Decoder for Systematic Cyclic Codes

For the decoder, the input data is the received bit stream of n bits, consisting of k data bits followed by $(n - k)$ check bits. If there have been no errors then after the first k steps, the shift register contains the pattern of check bits that were transmitted. After the remaining $(n - k)$ steps, the shift register contains a syndrome pattern. Thus, cyclic codes can be easily encoded and decoded using shift registers.

For decoding, the comparison logic receives the incoming codeword directly as well as from a calculation performed on it. A bit-by-bit comparison by XOR operation yields in a bit pattern known as *syndrome*. By definition, the syndrome $S(x)$ of the received vector $R(x)$ is the remainder resulting from dividing it by generator polynomial $g(x)$. That is,

$$\frac{R(x)}{g(x)} = P(x) + \frac{S(x)}{g(x)}$$

where $P(x)$ is the quotient of the division.

The syndrome $S(x)$ is a polynomial of degree $n - k - 1$. The range of the syndrome is between 0 and $2^{n-k} - 1$. If $D(x)$ represents the message polynomial, then the error pattern $E(x)$ caused by the channel is given by $E(x) = [P(x) + D(x)]g(x) + S(x)$

$$\Rightarrow \qquad S(x) = E(x) - [P(x) + D(x)]g(x)$$

Hence, the syndrome $S(x)$ of received vector $R(x)$ contains the information about the error pattern $E(x)$ which is useful for determining the error correction.

Figure 5.3.4 shows an arrangement for an $(n - k)$ syndrome decoding for an (n, k) cyclic code.

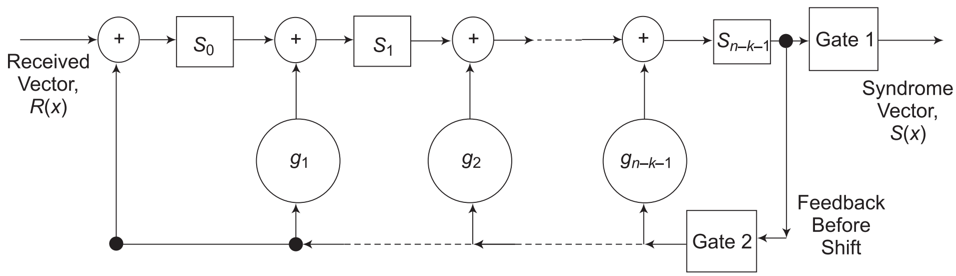


Figure 5.3.4 Syndrome Decoding for an (n, k) Cyclic Code

Procedure for Syndrome Decoding Operation

The decoding operation of a cyclic code is carried out in following steps:

Step 1: Initialize the register.
Step 2: Turn-off Gate 1 and turn-on Gate 2.
Step 3: Enter the received vector $R(x)$ into the shift register.
Step 4: After shifting of the complete received vector into the register, the contents of the register will be the syndrome.
Step 5: Now turn-off Gate 2 and turn-on Gate 1.
Step 6: Shift the syndrome vector out of the register.

The circuit is now ready for processing the next received vector for calculation of next syndrome.

Application

CRC is designed to detect accidental changes in raw computer data, and is commonly used in digital networks and storage devices such as hard-disk drives. Blocks of user data get a short check bits attached, derived from the remainder of a polynomial division of their contents. On retrieval, the calculation is repeated, and corrective action can be taken against presumed data corruption if the checksum do not match.

SOLVED EXAMPLE 5.3.9 **Computation of CRC**

Given a 6-bit data pattern 100100 and a divisor of 1101. How many number of 0s will be appended at the given 6-bit data pattern to serve as dividend at the sender end? Compute CRC for a 6-bit data bit pattern 100100 and a divisor of 1101.

Solution We know that the generator polynomial $P(x)$ must be a prime number and the number of 0s to be appended at the given data pattern is one zero less than the number of bits in the divisor. Therefore, for given 6-bit data pattern 100100 and a 4-bit divisor of 1101, the number of 0s to be appended at the given 6-bit data pattern is 3 (three zeros) to serve as dividend at the sender end. For given 6-bit data pattern 100100, the dividend at the sender end (after appending three 0s) is *100100000.* The CRC can be computed as shown below:

```
                111101 = Quotient
Divisor 1101 | 100100000 = Data plus three extra 0s
               1101
                1000
                1101
                 1010
                 1101
                  1110
                  1101
                   0110
                   0000
                    1100
                    1101
                     001 = Remainder
```

The remainder is CRC. Therefore, the CRC = 001 **Ans.**

SOLVED EXAMPLE 5.3.10 **Error Detection by CRC**

For a particular data sequence, the CRC is 001. If the data pattern is 100100, what will be transmitted data sequence? If there is no transmission error, then what will be the remainder at the receiver end?

Solution We know that the transmitted data sequence is the given data pattern plus CRC bit. Therefore, for given 6-bit data pattern 100100 and CRC 001, the transmitted data sequence is *100100001.* **Ans.**

The received data sequence (assuming no error) is same as the transmitted data sequence. Therefore, the received data sequence is *100100001.* As a rule, if the transmission has no errors, the remainder $R(x)$ will be zero, otherwise it is non-zero in case the transmission has some errors.

Since it is given that there is no error, the remainder at the receiver end is 000. **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 5.3.11 Determine the CRC for the data sequence 10110111 for which the generator polynomial is $+ x^5 + x^4 + x^1 + x^0$ (i.e., 110011).

Ex 5.3.12 At the receiver, the received data pattern is 1011011101001, for which the generator polynomial is $x^5 + x^4 + x^1 + x^0$ (i.e., 110011). Determine the remainder and show that there is no transmission error.

5.3.4 BCH Codes

BCH Code — A Class of Cyclic Code

BCH codes were invented by Raj Chandra Bose and Dwijendra Kumar Ray Chaudhary, and by Alexis Hocquenghem in 1959–60. So, named after its inventors, *Bose–Chaudhuri–Hocquenghem, BCH code* is one of the most important and powerful class of random-error-correcting cyclic polynomial codes over a finite field with a particular chosen generator polynomial which generalized the Hamming code for multiple error correction. In fact, hamming codes are the subset of BCH codes with $k = 2^m - 1 - m$ and have an error correction capability of 1. The main advantage of BCH codes is the ease with which they can be decoded using syndrome and many good decoding algorithms exist.

Design Considerations

One of the major consideration in the design of optimum codes is to make the size (n) of the encoded block as small as possible for a given message block size (k). This is required so as to obtain a desired value of minimum Hamming distance (d_{min}). In other words, for a given value of n and k, a code is required to be designed with largest d_{min}.

Parameters of BCH Codes

For any positive pairs of integers m ($m \geq 3$) and t ($t < 2^m < 1$), where t represents the number of errors that can be corrected, there is a binary (n, k) BCH codes with the following parameters:

Block length, $n = 2^m - 1$

Number of check bits, $(n - k) \leq mt$

Minimum Hamming distance, $d_{min} \geq (2t + 1)$

Salient Features of BCH Codes

- The BCH codes provide flexibility in the choice of parameters (block length and code rate).
- The generator polynomial for this code can be constructed from the factors of $(X^{2m-1} + 1)$.
- BCH code can correct all combinations of less than or equal to t number of errors.
- Single error-correcting BCH code is equivalent to a Hamming code, which is a single-error correcting linear block code.
- BCH code is capable of correcting any combination of t or few number of errors in a block code $n = 2^m - 1$ digits, and hence it is called t-error-correcting BCH code.
- The generator polynomials of BCH code is specified in terms of its roots from Galois field GF(2^m).

BCH codes are widely used in wireless communication applications.

Table 5.3.9 summarizes the basic parameters of three commonly used BCH codes in various digital communication applications.

Table 5.3.9 Parameters of BCH Codes

BCH Code (n, k)	Generator Polynomial Coefficients	Generator Polynomials	n	k	t
(7, 4)	1011	$1+x+x^3$	7	4	1
(15, 11)	10011	$1+x+x^4$	15	11	1
(31, 26)	100101	$1+x^2+x^5$	31	26	1
(15, 7)	111010001	$1+x^4+x^6+x^7+x^8$	15	7	2
(31, 21)	11101101001	$1+x^3+x^5+x^6+x^8+x^9+x^{10}$	31	21	2
(15, 5)	10100110111	$1+x+x^2+x^4+x^5+x^8+x^{10}$	15	5	3
(31, 16)	1000111110101111	$1+x+x^2+x^3+x^5+x^7+x^8+x^9+x^{10}+x^{11}+x^{15}$	31	16	3

Decoding a BCH Codes

BCH codes can be decoded in many way and it is the most common that
- Syndrome values are calculated for the received codeword
- Error polynomials are calculated

- Roots of these polynomials are calculated to obtain the location of errors
- Error values are calculated at these locations

For binary BCH codes it is only necessary to calculate the position, because the error value is always equal to 1. In non-binary BCH codes an additional error value polynomial is needed.

SOLVED EXAMPLE 5.3.13 **Error-correcting (15, 7) BCH Code**

Show that (15, 7) BCH code is a double-error-correcting BCH code.

Solution For (15, 7) BCH code, we have $n = 15$ and $k = 7$.

We know that Block length, $n = 2^m - 1$

$\Rightarrow$ $15 = 2^m - 1;$

$\Rightarrow$ $16 = 2^m;$ or, $m = 4$

We know that in BCH code, number of check bits, $(n - k) \le mt$; where t is the number of error-correcting bits.

$\Rightarrow$ $(15 - 7) \le 4 \times t;$

$\Rightarrow$ $t = 2$

Thus, (15, 7) BCH code is a double-error-correcting BCH code.

SOLVED EXAMPLE 5.3.14 **Error-correcting (31, 16) BCH Code**

Prove that (31, 16) BCH code can correct minimum three number of errors. Also determine the minimum hamming distance.

Solution For (31, 11) BCH code, we have $n = 31$ and $k = 16$.

We know that Block length, $n = 2^m - 1$

$\Rightarrow$ $31 = 2^m - 1$

$\Rightarrow$ $32 = 2^m;$ or, $m = 5$

We know that in BCH code, number of check bits, $(n - k) \le mt$; where t is the number of error-correcting bits.

$\Rightarrow$ $(31 - 16) \le 5 \times t;$

$\Rightarrow$ $t = 3$

Thus, (31, 16) BCH code can correct minimum three number of errors or it is a three-error-correcting BCH code.

We know that minimum Hamming distance, $d_{\min} \ge (2t + 1)$

For $t = 3$, $d_{\min} \ge (2 \times 3 + 1)$; $d_{\min} = 7$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 5.3.15 Determine the number of errors that can be corrected by
(a) BCH code (15, 11)
(b) BCH code (15, 7)

Ex 5.3.16 Show that BCH code (15, 5) is a 3-error correcting BCH code. Also calculate its minimum Hamming distance.

5.3.5 Hadamard Codes

Named after its inventor, the *Hadamard codes* are obtained from the Hadamard matrix. The Hadamard matrix is a square ($n \times n$) matrix, and is described in its general form as

Hadamard Matrix

$$H_{2n} = \begin{pmatrix} H_n & H_n \\ H_n & H_n^* \end{pmatrix}$$

where H_{2n} represents an ($n \times n$) Hadamard matrix provides n codewords each of n bits. H_n^* is the H_n matrix with each element replaced by its complement. For example, the Hadamard matrix for $n = 1$ is

$$H_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

So the codewords are 00 and 01.

And the Hadamard matrix for $n = 2$ is

$$H_4 = \begin{pmatrix} H_2 & H_2 \\ H_2 & H_2^* \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \end{pmatrix}$$

As evident, H_2^* is the H_2 matrix with each element replaced by its complement.

Features of Hadamard Codes

- The codewords in a Hadamard code are given by the rows of a Hadamard matrix.
- One codeword consists of all zeros (corresponding to the first row of the Hadamard matrix).
- All other codewords have equal number of zeros and ones, $\frac{n}{2}$ each.
- If k is the number of bits in the uncoded word then $n = 2^k$.
- Each codeword differs from every other codeword in $\frac{n}{2}$ places.
- The codewords are said to be orthogonal to one another.

An ($n \times n$) Hadamard matrix provides n codewords each of n bits. For example,
- In (2×2) Hadamard matrix, the codewords are 0 0 and 1 1.
- In (4×4) Hadamard matrix, the codewords are 0 0 0 0, 0 1 0 1, 0 0 1 1, and 0 1 1 0.

Error Control Bits in Hadamard Codes

Each codeword contains $n = 2^k$ bits. Hence in the n bit codeword there are $r = n - k = 2^k - k$ error-control bits. As can be seen that as k increases, the number of error control bits becomes extremely large in comparison with the number k of information bits. Thus, the rate of the code becomes quite small since

$$R_c = \frac{k}{n} = \frac{k}{2^k} = k2^{-k}$$

This means that with large value of k, if we transmit coded words at the same rate as uncoded words, the coded transmission will require a bit rate larger than the uncoded bit rate by the factor $1/R_c$. This would result into corresponding increase in the required bandwidth, as in the case of orthogonal signaling.

Hamming Distance in Hadamard Codes

The *Hamming distance* of an orthogonal code such as Hadamard code is given as $d_{\min} = \frac{n}{2} = \frac{2^k}{2} = 2^{k-1}$. Therefore, the number of errors that can be corrected with a Hadamard code can be estimated using $(2t + 1) \le d_{\min}$(odd); $(2t + 2) \le d_{\min}$(even), where t is the number of errors in the received codeword.

For example, for even minimum Hamming distance, we get $(2t + 2) \le 2^{k-1}$. Re-arranging the terms, we have

$$2t \le 2^{k-1} - 2; \Rightarrow t \le \frac{2^{k-1} - 2}{2}; \Rightarrow t \le 2^{k-2} - 1$$

Hence, to provide error correction, k should be greater than 2. However, for larger k, t increases as 2^k. So for larger value of k, significant error correction is possible with Hadamard code.

5.3.6 LDPC Codes

Recall

A binary parity-check code is a block code, i.e., a collection of binary vectors of fixed length n. Each codeword of length n can contain $k = (n–r)$ information digits, where r represents check digits. A parity check matrix is an r-row by n-column binary matrix.

LDPC Codes – A Form of Non-cyclic Codes

Any linear code has a bipartite graph and a parity-check matrix representation. But not all linear code has a sparse representation. A $[n \times m]$ matrix is sparse if the number of 1's in any row, the row weight w_r, and the number of 1s in any column, the column weight w_c, is much less than the dimension ($w_r << m$, $w_c << n$). A code represented by a sparse parity-check matrix is called low density parity check code (LDPC). The sparse property of LDPC gives rise to its algorithmic advantages.

Low-Density Parity Check (LDPC) Codes are a class of non-cyclic linear block codes. The name has been derived from the characteristics of their parity-check matrix which contains only a few 1s in comparison to the amount of 0s. Their main advantage is that they provide a performance which is very close to the capacity for a lot of different channels and linear time complex algorithms for decoding. Furthermore they are best suited for implementations that make heavy use of parallelism. LDPC codes have better performance with large code length. Code lengths are larger than 1000 bits. LDPC codes are easier to generate.

A Glimpse of LDPC History

LDPC codes were first introduced by Gallager in his PhD thesis in 1960. But due to large complexity of computational efforts in implementing them and the introduction of Reed–Solomon codes, they were mostly ignored for almost 30 years. However, in 1981, Michael Tanner generalized LDPC codes and introduced a graphical representation of the codes later called Tanner graph. Since 1993, with the invention of Turbo codes, researchers focused to find low-complexity code which can approach Shannon channel capacity. Mackay and Luby reinvented LDPC which has many applications in modern digital communications systems.

Regular and Irregular LDPC Codes

- For *regular LDPC codes*, all rows as well as all columns have equal weights. In other words, a regular LDPC code has the property that every code digit is contained in the same number of equations, and each equation contains the same number of code symbols.

- *Irregular LDPC codes* have a variable number of 1s in the rows and in the columns. The optimal distributions for the rows and the columns are found by a technique called *density evolution*. The basic idea is to give greater protection to some digits and to have some of the parity equations give more reliable information to give the decoding a jump start. For *irregular LDPC codes*, the rows weights and columns weights exhibit certain weight distributions such as the number of 1s in the i^{th} row of the parity check matrix is known as the *row weight*, and the number of 1s in the j^{th} column of the parity-check matrix is known as the *column weight*. Both row and column weights are much smaller than the code length. Irregular LDPC codes perform better than regular LDPC codes.

Representations for LDPC Codes by a Tanner Graph

Basically, there are two different possibilities to represent LDPC codes. Like all linear block codes, they can also be described using matrices. The second possibility is a graphical representation. Tanner introduced an effective graphical representation for LDPC codes, known as *Tanner graph*. Its concept is quite helpful to understand its iterative decoding algorithms. Tanner graphs are bipartite graphs. That means that the nodes of the graph are separated into two distinctive sets and edges are only connecting nodes of two different types. The two types of nodes in a Tanner graph are called variable nodes (v-nodes), also known as bit nodes, and parity check nodes (c-nodes). Each bit node represents a code symbol and each parity check node represents a parity equation. There is a line drawn between a bit node and a parity check node if and only if that bit is involved in that parity equation.

In a basic Tanner graph, the bit nodes (columns) are connected to their respective parity check nodes (rows) according to the presence of 1s in the parity check matrix [H]. Bit node and a check node are connected together if the corresponding element is 1 in the parity check matrix [H]. This is illustrated with the help of the following example.

SOLVED EXAMPLE 5.3.17 **An Example of a Tanner Graph**

Determine the Tanner graph of a Hamming code (7, 4, 3), having 7 variable nodes $\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$, and 3 check nodes $\{c_1, c_2, c_3\}$, whose parity check matrix [H] is given as

$$[H] = \begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \end{array} \begin{array}{c} \begin{array}{ccccccc} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

Solution As seen from the entries of 1s in given parity check matrix [H], the Tanner graph will have the following connections between the check nodes and the variable nodes, depending on the presence of 1s in the corresponding junctions.

- c_1 is connected to four variable nodes $\{v_1, v_2, v_4, v_7\}$
- c_2 is connected to four variable nodes $\{v_2, v_3, v_4, v_6\}$
- c_3 is connected to four variable nodes $\{v_1, v_2, v_3, v_5\}$

The resulting Tanner graph is given in Figure 5.3.5.

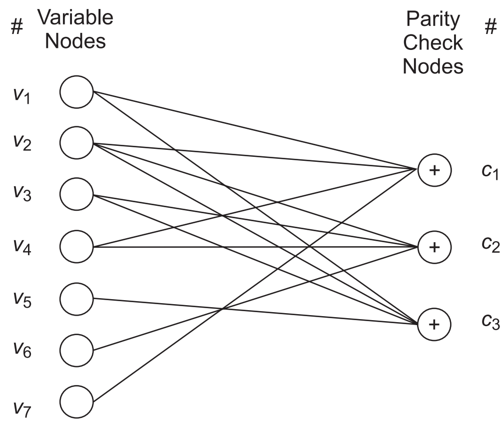


Figure 5.3.5 Tanner Graph for given [H]

Generally, LDPC codes are typically of length more than 1000, their Tanner graphs cannot be illustrated practically.

Interpretation of Tanner Graph for LDPC Codes

A cycle in the Tanner graph is indicated by a closed loop of connected nodes. The loop can originate at any variable or check node and ends at the same originating variable or check node. The number of cycle is defined by the number of nodes. When a Tanner graph is free of short cycles (of lengths 4 and 6), iterative decoding algorithms for LDPC codes can converge and generate desired results. To prevent a cycle of length 4, it is recommended that LDPC code should meet the row-column constraint which states that no two rows or columns may have more than one common component (0 or 1).

Decoding Techniques for LDPC Codes

There are several decoding techniques available. Some of them are mentioned below in order of their increasing complexity and improving performance.

- Majority logic (MLG) decoding
- Bit-flipping (BF) and Weighted bit-flipping (WBF) decoding algorithms
- *Iterative Decoding based on Belief Propagation (IDBP),* also known as the *Sum-Product Algorithm (SPA)*
- *A Posteriori Probability (APP)* decoding

MLG Decoding of LDPC Codes

- Decoding is accomplished by passing messages along the lines of the Tanner graph.
- The messages on the lines that connect to the i^{th} bit node, v_i, are estimates of some equivalent information.
- At the bit nodes the various estimates are combined in a particular way.
- Each bit node is furnished an initial estimate of the probability it is a 1 from the soft output of the channel.
- The bit node broadcasts this initial estimate to the parity check nodes on the lines connected to that bit node.
- But each parity check node must make new estimates for the bits involved in that parity equation and send these new estimates (on the lines) back to the bit nodes.
- Each parity check node knows that there are an even number of 1s in the bits connected to that node.

- But the parity check node has received estimates of the probability that each bit node connected to it is a 1.
- The parity check node sends a new estimate to the i^{th} bit node based upon all the other probabilities furnished to it.
- The channel estimate is always used in all estimates passed to the parity node.
- The process now repeats, i.e., parity-check nodes passing messages to bit nodes and bit nodes passing messages to parity-check nodes.
- At the last step, a final estimate is computed at each bit node by computing the normalized product of all of its estimates.
- Then a hard decision is made on each bit by comparing the final estimate with the threshold 0.5.

Bit-flipping Algorithm for LDPC Decoding

Bit-flipping decoding was introduced by Gallager in 1961. The bit-flipping algorithm for LDPC decoding operates on a sequence of hard-decision bits r = 011001101 ... 110. Parity checks on r generate the syndrome vector given by $s = rH^T$. The procedure for LDPC decoding is given below:

Step 1: Calculate the parity checks on r and generate $s = rH^T$, where H^T denotes the transpose of parity check matrix $[H]$.

Step 2: If all syndromes are zero, stop decoding.

Step 3: Determine the number of failed parity checks for every bit.

Step 4: Identify the set of bits with the largest number of failed parity check bits.

Step 5: Flip the bits to generate a new codeword r'.

Step 6: Let $r = r'$ and repeat above steps until the maximum number of iterations has been reached.

If few errors occur, decoding should commence in a few iterations. However, on a noisy channel, the number of iterations should be allowed to grow large. The inherent parallelism in decoding LDPC codes suggests their use in high data rate systems.

The Belief Propagation Algorithm

The belief propagation algorithm, also known as the sum-product algorithm or the message passing algorithm is based on hard-decision as well as soft-decision schemes. The hard-decision algorithm is as follows:

Step 1: All bit nodes send a bit to their connected parity check nodes.

Step 2: Every parity-check node calculates a response to their connected bit nodes using the bits they receive from step 1. The response bit in this case is the value (0 or 1) that the parity-check node believes the bit node has based on the information of other bit nodes connected to that parity-check node. This response is calculated using the parity-check equations which force all bit nodes connect to a particular parity-check node to sum to 0 (mod 2). At this point, if all the equations at all parity check nodes are satisfied, meaning the values that the parity check nodes calculate match the values they receive, the algorithm terminates. If not, then we follow Step 3.

Step 3: The bit nodes use the bits they get from the parity-check nodes to decide if the bit at their position is a 0 or a 1 by majority rule. The bit nodes then send this hard-decision to their connected parity check nodes.

Step 4: Repeat Step 2 until either exit at Step 2 or a certain number of iterations has been passed.

As an example, let us consider a (4, 8) linear block code whose parity-check matrix is given by

$$[H] = \begin{bmatrix} 0\,1\,0\,1\,1\,0\,0\,1 \\ 1\,1\,1\,0\,0\,1\,0\,0 \\ 0\,0\,1\,0\,0\,1\,1\,1 \\ 1\,0\,0\,1\,1\,0\,1\,0 \end{bmatrix}$$

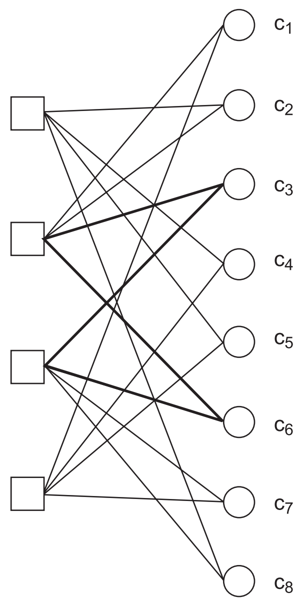


Figure 5.3.6 An Example of Belief Propagation for Linear Block Code [H]

The soft-decision decoder operates with the same principle as the hard-decision decoder, except that the bits are the conditional probability that the received bit is a 1 or a 0 given the received vector. In practice, belief propagation is executed for a maximum number of iterations or until the passed likelihoods are closed to certainty, whichever comes first.

Interpretation of Belief Propagation Algorithm

One very important aspect of belief propagation is that its running time is linear to the code length. Since the algorithm traverses between parity-check nodes and bit nodes, and the graph is sparse, the number of traversals is small. Moreover, if the algorithm runs a fixed number of iterations then each edge is traversed a fixed number of times, thus the number of operations is fixed and only depends on the number of edges. If we allow the number of parity-check nodes and bit nodes increases linearly with the code length, the number of operations performed by belief propagation also increases linearly with the code length.

Other Methods of Decoding LDPC

- Use weighted MLG decoding technique to include soft decision, i.e., reliability information in the decoding decision.
- *Density Evolution (DE)* is a technique to analyze the performance of belief propagation decoding of a certain ensemble of LDPC code transmitted over a particular (static,

memoryless, symmetric) channel. DE computes, at each iteration starting with 0, the message distribution along each class of edge relative to the value of the variable associated with each edge. For a given channel, DE can determine whether the bit error probability approaches 0 as block length and number of iterations goes to infinity.
- The SPA decoding technique is also iterative decoding based on belief propagation. It can be performed on the Tanner graph.

Salient Features of LDPC Codes

Given a specific channel, the performance of LDPC codes is measured in terms of either bit-error probability or block-error probability. From a practical perspective, block-error probability is a more useful measure although, perhaps for historical reasons, bit error probability is widely quoted. Often, the error probability is plotted on a log scale against a parameter that specifies one of a class of channels ranging from low-fidelity to high-fidelity. For example, error probability is often plotted against the signal-to-noise ratio (S/N) of an *Additive White Gaussian Noise (AWGN)* channel, or against the probability of crossover on a *Binary Symmetric Channel (BSC)* or probability of erasure on an erasure channel. Some of the salient features of LDPC codes are listed below:
- Better block error performance.
- Iterative decoding and simpler operations in each iteration step.
- Decoding is not based on trellises.
- Performance similar to Turbo codes.
- More iterations than Turbo decoding.
- Do not require long interleaver to achieve good performance.
- Error floor (a region in which the error probability does not approach zero as fast as it might be expected to) occurs at lower BER. Many LDPC codes, especially those that have been designed to perform very close to channel capacity, do exhibit error floors.

Applications of LDPC Codes

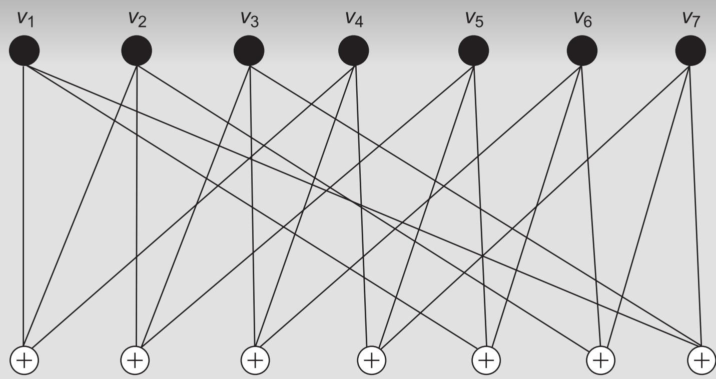
Low density parity-check code (LDPC) is an error correcting code used in noisy communication channel to reduce the probability of loss of data. With LDPC, this probability can be reduced to as small as desired, thus the data transmission rate can be as close to Shannon limit as desired. LDPC have made its way into some modern applications such as 10GBase-T Ethernet, WiFi, WiMAX, and Digital Video Broadcasting (DVB).

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 5.3.18 Construct the Tanner graph of a code having 7 bit nodes and 7 parity check nodes whose parity check matrix [H] is given as

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

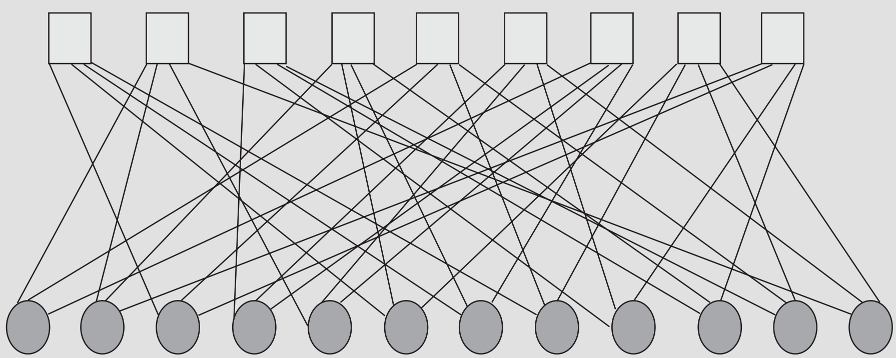
Hint:



Ex 5.3.19 Construct the Tanner graph of a code having 9 bit nodes and 12 parity check nodes whose parity check matrix [H] is given as

$$\begin{matrix}
0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0
\end{matrix}$$

Hint:



Squares represent parity check nodes and circles represent bit nodes. Note that each bit node has 3 lines connecting it to parity-check nodes and each parity check node has 4 lines connecting it to bit nodes.

Low-Density Parity-Check (LDPC) codes have recently received a lot of attention for all the next generation communication standards due to their excellent error-correcting capability. LDPC codes are well worthwhile investigating. Some issues to be resolved are performance for channel models of interest, optimization of irregular LDPC codes, and implementation in VLSI. These have been adopted as an optional error correction coding scheme by Mobile WiMAX (IEEE802.16e) – broadband wireless standards, and digital video broadcast (DVB)-S2 standards.

Application

LET'S RECONFIRM OUR UNDERSTANDING!!

- If t is the error-correction capability of a BCH code, then what would be the minimum distance of the code?
- A BCH code over GF(2^6) can produce the code with maximum error capability of 10. *True* or *False*?
- Do irregular LDPC codes have a fixed or variable number of 1's in the rows and in the columns?

Self-Assessment Exercise linked to LO 5.3

Q5.3.1 Highlight the need of channel coding in a digital communication system. Mention the chief objective(s) of a channel code.

Q5.3.2 Define the term: Single-bit error. "Single-bit errors in data communications are the least likely type of error in serial data transmission". Justify this statement with the help of an example.

Q5.3.3 In GSM cellular communication system, a data block of 184 bits is encoded into 224 bits of codeword on the control channel before sending it to a channel encoder. Determine the number of error control bits added and the code rate of the encoder.

Q5.3.4 Give a brief account of advantages and disadvantages of block codes. What happens if we use a large block lengths of user data?

Q5.3.5 Distinguish between the desirable properties of linear block codes and cyclic codes.

Q5.3.6 What is meant by a Syndrome word? Hypothesize the Syndrome decoding procedure in cyclic code by taking appropriate example data.

Q5.3.7 For the codeword bit pattern 1 0 0 1 1 0 1, find out a Hamming code to detect and correct for single-bit errors assuming each codeword contains a 7-bit data field. Verify that the received bit pattern has no error. Also find the corrected bit pattern if the received bit pattern is 0 0 0 1 1 1 0 0 1 0 1.

Q5.3.8 For a particular data sequence, the CRC is 001. If the data pattern is 100100, what will be transmitted data sequence? If there is no transmission error, then what will be the remainder at the receiver end?

Q5.3.9 Construct a double-error-correcting BCH code over Galois field GF(2^3). Also find the minimum Hamming distance.

Q5.3.10 What is the error-correction capability of a (15, 5) BCH code over GF(2^4)?

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/114

> ✓ **If you have been able to solve the above exercises then you have successfully mastered**
>
> **LO 5.3:** *Implement error-control channel-coding techniques such as Hamming codes, cyclic codes, BCH code, Hadamard code, and LDPC code.*

LO 5.4

5.4 CONVOLUTION CODING AND DECODING

Convolution Codes—Historical Development

Convolution codes were first introduced by Elias in 1955 as an alternative to block codes. Wozencraft proposed sequential decoding algorithm for convolution codes. In 1967, Viterbi introduced a decoding algorithm for convolution codes which has since become known as *Viterbi algorithm*. Later, Omura showed that the Viterbi algorithm was equivalent to finding the shortest path through a weighted graph. Forney recognized that it was, in fact, a maximum likelihood decoding algorithm for convolutional codes; i.e., the decoder output selected is always the codeword that gives the largest value of the log-likelihood function. Viterbi decoding algorithm is relatively easy to implement for codes with small memory orders. Viterbi decoding along with improved versions of sequential decoding led to the application of convolution codes to satellite and deep-space communication in early 1970s and in cellular mobile communications recently. Convolution codes are applied in applications that require good performance with low implementation cost.

What We Discuss Here ...

- The concept of convolution coding technique with an illustration of a feed-forward convolution encoder.
- Characterization of convolution coding with state diagram, Trellis diagram, and code tree.
- Sequential and Viterbi decoding algorithms of a convolution code.

Essence of Convolution Coding

Convolution codes operate on data stream, not pre-defined data block. Convolution codes have memory that uses previous bits to encode or decode the following bits. A convolution code is generated by passing the information sequence through a linear finite-state shift register. In general, the shift register consists of K (k-bit) stages and n linear algebraic function generators. Unlike block codes, *convolution codes* do not map individual blocks of input information data bits into blocks of codewords. A *convolution encoder* accepts a fixed number of information symbols and produces a fixed number of encoded symbols, but its computation depends not only on the current set of input symbols but also on some of the previous input symbols. In other words, convolution encoder accepts a continuous sequence of data bits and maps them into an output data sequence, adding redundancies in the convolution process. Most convolution codes are designed to combat random independent errors.[6]

How do Convolution Codes differ from Block Codes?

Convolution codes differ from block codes in that the encoder contains memory and the n encoder outputs at any given time unit depend not only on the k inputs at that time unit but also on m previous input blocks. An (n, k, m) convolution code can be constructed with k-bit input (the value of k may be 1 also), n-*bit* ($n > k$) output, and m-memory elements. The memory order m must be large enough to achieve low error probability, whereas n and k are typically small

---

[6] The GSM system uses conventional convolutional encoding in combination with block codes. The code rate varies with the type of input; in the case of speech signal, it is 260/456 < ½.

integer values. In the special case when $k = 1$, the information sequence is not divided into blocks and can be processed continuously. Rather, $k = 1$ is usually used!! Since the convolution encoder has memory, it is implemented with *sequential* logic circuits.

Convolutionally Encoding the Data

Convolutionally encoding the data is accomplished using a shift register (a chain of flip-flops) and associated combinational logic that performs modulo-two addition (e.g., cascaded exclusive-OR logic gates). In convolution encoder, data bits are provided at a rate of k bits per second. The input bit is stable during the encoder cycle which starts when an input clock edge occurs. Channel symbols are output at a rate of $n = 2k$ symbols per second. Each input bit has an effect on successive pairs of output symbols. That is an extremely important point and that is what gives the convolution code its error-correcting power.

Code Rate of the Convolution Code

The code rate R_c of the convolution encoder is determined by input data rate and output data rate and is always less than unity. A k-bit information data sequence produces a coded output sequence of length $n(k + m)$ bits. Therefore, code rate of the convolution code,

$$\boxed{R_c = \frac{k}{n(k+m)}}$$

If $k >> m$, the code rate reduces to $1/n$ bits per symbol. In general, the code rate, $R_c = \frac{k}{n}$.

Remember each encoded bit is a function of the present input bits and their previous bits.

Polynomial Description of a Convolutional Encoder

A polynomial description of a convolution encoder includes constraint lengths, generator polynomials, and feedback connection polynomials (for feedback encoders only).

Constraint length (K) of a convolution code is defined as the number of shifts over which a single information bit can influence the encoder output. It is expressed in terms of information bits, and is an important parameter in the design of convolution encoding. The constraint lengths of the encoder form a vector whose length is the number of inputs in the encoder diagram. The number of the preceding bits is same as the number of memory elements in the convolution encoder. The elements of this vector indicate the number of bits stored in each shift register, including the current input bits.

In an encoder with an m-stage shift register, the memory of the encoder equals m information bits. We see that $K = m + 1$ shifts are required for an information bit to enter the shift register and finally come out, where K is the constraint length of the convolution encoder.

If the encoder diagram has k inputs and n outputs, then the *code generator* matrix is a *k-by-n* matrix. The element in the i^{th} row and j^{th} column indicates how the i^{th} input contributes to the j^{th} output. For systematic bits of a systematic feedback encoder, the entry in the code generator matrix is matched with the corresponding element of the feedback connection vector.

Feed-forward Convolution Encoders

An (n, k, m) convolution code can be developed with k information bits and m memory elements to produce n output encoded bits. The redundancy bits depend on not only the current k bits but also several of the preceding k bits. An important special case is when $k = 1$, the information sequence is not divided into blocks and can be processed continuously. The check bits are continuously interleaved with information data bits. A binary convolution encoder consists of m-stage shift register with prescribed connections to a modulo-2 adders, and a multiplexer that converts the outputs of adders in serial output data sequence. The convolution encoder distinguishes itself by the use of feed-forward paths only (called *non-recursive non-systematic convolution encoder*) because the information bits lose their distinct identity as a result of convolution process.

An (n, k, m) convolution code can be described by generator sequences $g_1(x)$, $g_2(x)$, …, $g_n(x)$ that are impulse responses of each coder output branch. Generator sequences specify convolution code completely by the associated generator polynomials. Encoded convolution code is produced by multiplication of input data sequence and the generator polynomial.

Note... *The generation of encoded data sequence is illustrated with the help of following examples.*

SOLVED EXAMPLE 5.4.1 **Generation of Convolution Code**

Draw a typical binary convolution encoder with constraint length $K = 3$, $k = 1$, and $n = 3$. Also express their generator functions in binary, octal and polynomial forms. What is the code rate of this convolution encoder?

Solution For specified constraint length $K = 3$, $k = 1$, and $n = 3$, Figure 5.4.1 shows the convolution encoder having 3 memory elements (registers), 2 modulo adders, 1 input, and 3 outputs.

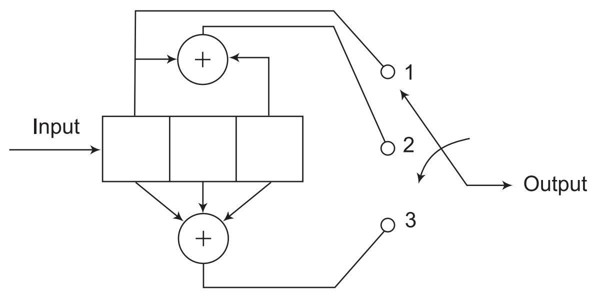


Figure 5.4.1 Convolution Encoder ($r = 1/3$)

Generator functions in binary form are $g_1 = [100]$, $g_2 = [101]$, and $g_3 = [111]$.

Generator functions in octal form are $g_1 = [4]$, $g_2 = [5]$, and $g_3 = [7]$.

Generator functions in polynomial forms are $g_1 = x^2$, $g_2 = 1 + x^2$, and $g_3 = 1 + x + x^2$

Since $k = 1$ and $n = 3$, therefore code rate, R_c or $r = 1/3$ **Ans.**

SOLVED EXAMPLE 5.4.2 **Generation of Convolution Code**

Using the generator polynomials, $g_1 = 1 + x + x^2$, and $g_2(x) = 1 + x^2$. Write the convolution code for data sequence 101011.

Solution Given input bit pattern, $k = 101011$

This corresponds to polynomial, $m(x) = 1 + x^2 + x^4 + x^5$

The given generator polynomials, $g_1(x) = 1 + x + x^2$ corresponds to generator sequence of (1 1 1), and $g_2(x) = 1 + x^2$ corresponds to generator sequence of (1 0 1).

This is a feed-forward convolution code encoder which has one input, two outputs, and two shift registers M_1 and M_2, as shown in Figure 5.4.2.

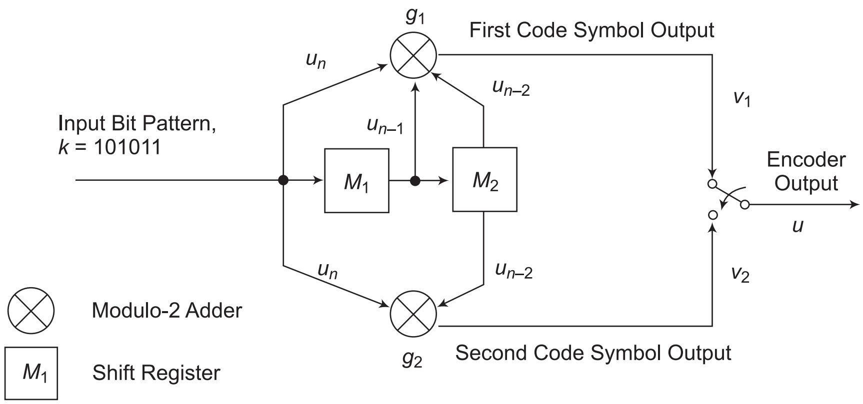


Figure 5.4.2 Convolution Encoder (r = 1/2)

Constraint length, $K = 3$ (No. of shift registers plus one)

Number of modulo-2 adders, $n = 2$

Therefore, code rate, $r \approx \dfrac{1}{n} \approx \dfrac{1}{2}$ bits/symbol

This means that for every single input bit, there will be two bits (v_1, v_2) at the output where $v_1 = u_n \otimes u_{n-1} \otimes u_{n-2}$ and $v_2 = u_n \otimes u_{n-2}$

We know that $v_1(x) = m(x)\ g_1(x)$

$\Rightarrow \quad v_1(x) = (1 + x^2 + x^4 + x^5)(1 + x + x^2)$

$\Rightarrow \quad v_1(x) = 1 + x + x^2 + x^2 + x^3 + x^4 + x^4 + x^5 + x^6 + x^5 + x^6 + x^7$

$\Rightarrow \quad v_1(x) = 1 + x + x^3 + x^7$

The corresponding bit sequence, v_1 = 11010001

Similarly, $v_2(x) = m(x)g_2(x)$

$\Rightarrow \quad v_2(x) = (1 + x^2 + x^4 + x^5)(1 + x^2)$

$\Rightarrow \quad v_2(x) = 1 + x^2 + x^2 + x^4 + x^4 + x^6 + x^5 + x^7$

$\Rightarrow \quad v_2(x) = 1 + x^5 + x^6 + x^7$

The corresponding bit sequence, v_2 = 10000111

Hence, the convolution code u for the given data sequence 101011 is

u = 11, 10, 00, 10, 00, 01, 01, 11 **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 5.4.3 Figure 5.4.3 depicts a rate 2/3 convolution encoder. Express the generator functions in binary form and octal form. Write their polynomial forms also.

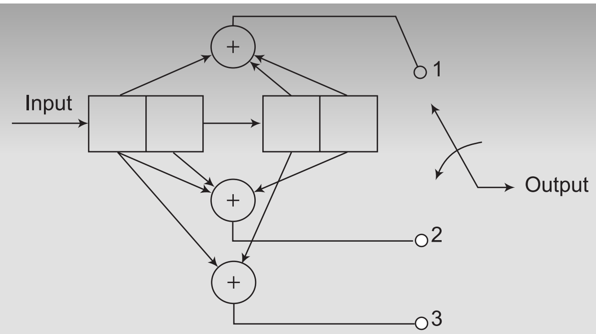


Figure 5.4.3 A Feed-Forward Convolution Encoder (r = 2/3)

Ex 5.4.4 Figure 5.4.4 depicts a feed-forward convolution encoder that has one input, two outputs, and two shift registers (M_1 and M_2).

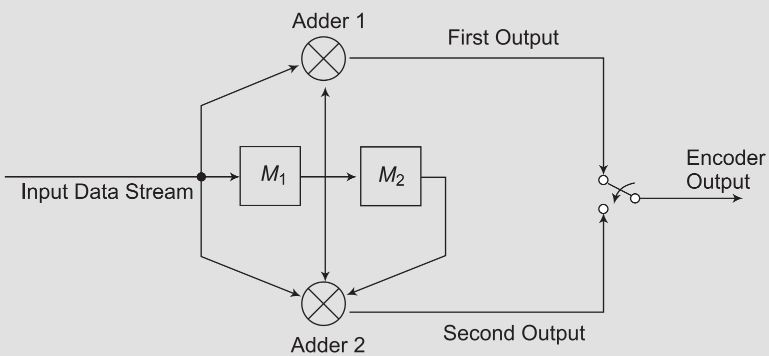


Figure 5.4.4 A Feed Forward Convolution Encoder (r = 1/2)

Compute the encoder output for the input data stream 101011.

Ex 5.4.5 For the convolution encoder given in Figure 5.4.5, compute the codeword for input bit sequence 11010.

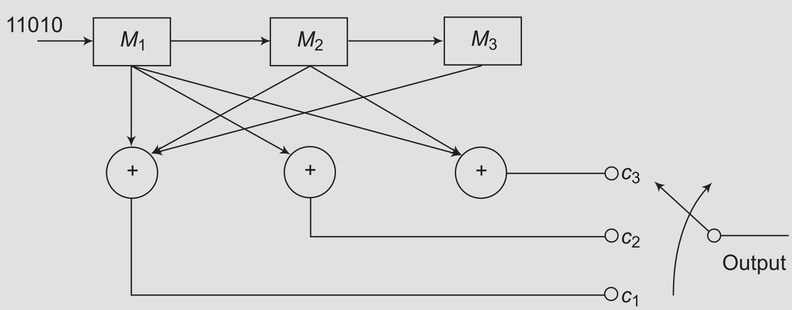


Figure 5.4.5 Convolution Encoder (r = 1/3)

How to Describe a Convolution Code

There are three distinct methods that are often used to describe a convolution code. These are:

- State diagram
- Trellis diagram
- Tree diagram or code tree

Convolution Encoder State Diagram

Generally, the convolution encoder can be completely characterized by the *state diagram* except that it cannot be used easily to track the encoder transitions as a function of time. The states represent the possible contents of the rightmost two stages of the shift register and the paths between the states represent the output bit pattern resulting from such state transitions. As a convention

- a solid line denotes a path associated with an input bit 0, and
- a dashed line denotes a path associated with an input bit 1.

Since a convolutional encoder is a sequential circuit, its operation can be described by a state diagram. The state of the encoder is defined as its shift register contents. For an (n, k, m) code with $k > 1$, the i^{th} shift register contains K_i previous information bits where K is total encoder memory. There are total 2^K different possible states. For a (n, 1, m) code, $K = K_1 = m$. Each new block of k inputs causes a transition to a new state. There are $2k$ branches leaving each state, one corresponding to each different input block. Note that for an (n, 1, m) code, there are only two branches leaving each state. Assuming that the encoder is initially in state S_0 (all-zero state), the codeword corresponding to any given information sequence can be obtained by following the path through the state diagram and noting the corresponding outputs on the branch labels. Following the last nonzero information block, the encoder is returned to state S_0 by a sequence of m all-zero blocks appended to the information sequence. This is illustrated with the help of following examples.

SOLVED EXAMPLE 5.4.6 **State Diagram of 1/3 Convolution Code**

Represent a convolution code state diagram for a convolution encoder having rate 1/3, $K = 3$.

Solution The state diagram for rate 1/3, $K = 3$ convolution code is shown in Figure 5.4.6.

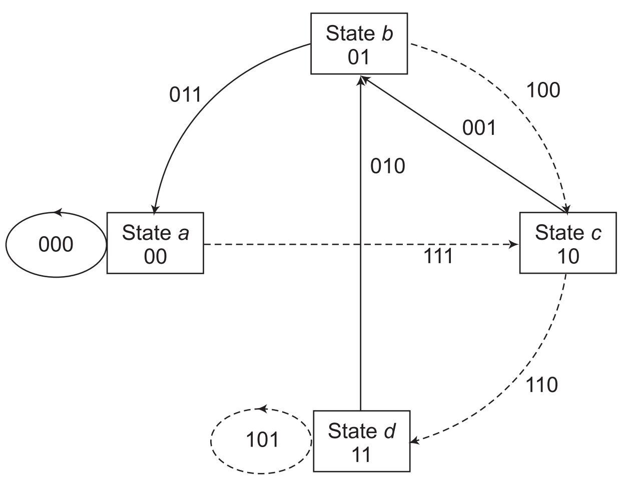


Figure 5.4.6 Convolution Encoder State Diagram (r = 1/3)

SOLVED EXAMPLE 5.4.7 **State Diagram of Convolution Code**

For the convolution encoder shown in Figure 5.4.7, draw the state diagram representation for an input bit pattern 101011.

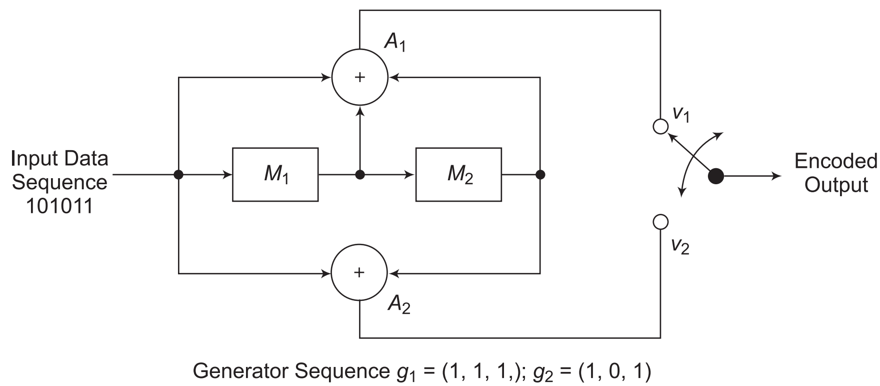


Figure 5.4.7 Convolution Encoder for $r = 1/2$ and $K = 3$

Solution The state diagram for the convolution encoder shown in Figure 5.4.7 for an input bit pattern 101011 is depicted in Figure 5.4.8.

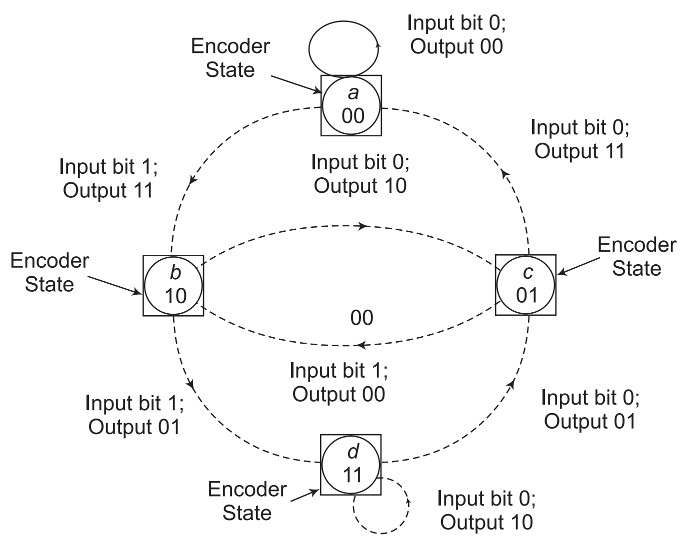


Figure 5.4.8 Convolution Encoder State Diagram

It can be seen from state diagram that the convolution code for given data sequence 101011 is 11, 10, 00, 10, 00, 01, 01, 11.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 5.4.8 Figure 5.4.9 depicts the state diagram of a (2, 1, 3) convolution code. Show that the codeword for an input bit sequence of 1 1 1 0 1 is 1 1, 1 0, 0 1, 0 1, 11, 10, 11, 11.

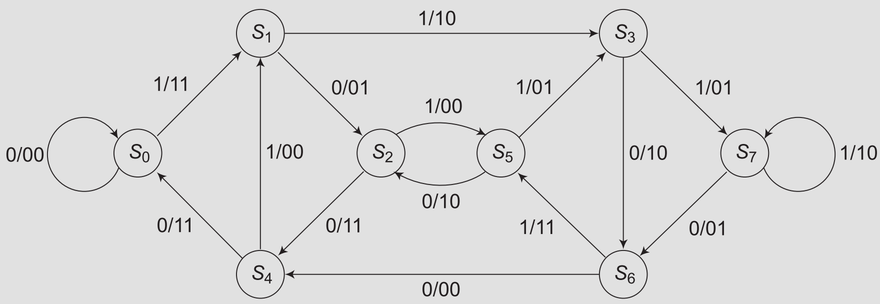


Figure 5.4.9 A Feed-Forward Convolution Encoder (2, 1, 3)

Convolution Encoder Trellis Diagram

A *trellis diagram* of a convolution encoder shows how each possible input to the encoder influences both the output and the state transitions of the encoder. Figure 5.4.10 depicts a trellis diagram for a convolution encoder ($r = 1/2$, $k = 3$).

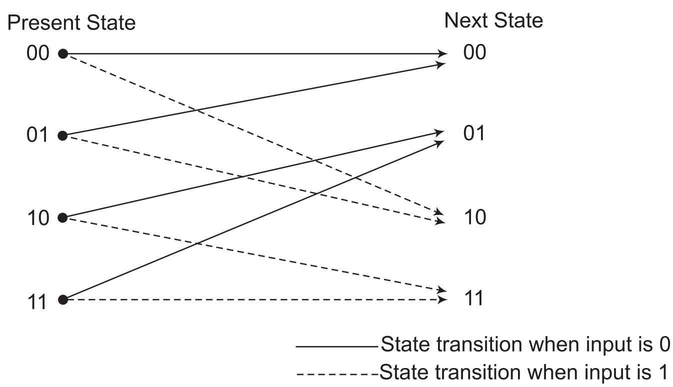


Figure 5.4.10 A Trellis Diagram for the Convolution Encoder

Interpretation of Trellis Diagram

As seen from the trellis diagram, the convolution encoder has four states (numbered in binary from 00 to 11), a one-bit input, and a two-bit output. The ratio of input bits to output bits makes this encoder a rate-1/2 encoder.

- Each solid arrow shows how the encoder changes its state if the current input is zero.
- Each dashed arrow shows how the encoder changes its state if the current input is one.

If the encoder is in 10 state and receives an input of zero then it changes to 01 state. If it is in 10 state and receives an input of one then it changes to 11 state.[7]

[7] Any polynomial description of a convolution encoder is equivalent to some trellis description, although some trellis have no corresponding polynomial descriptions.

Figure 5.4.11 shows a typical trellis diagram for rate 1/3, $K = 3$ convolution code.

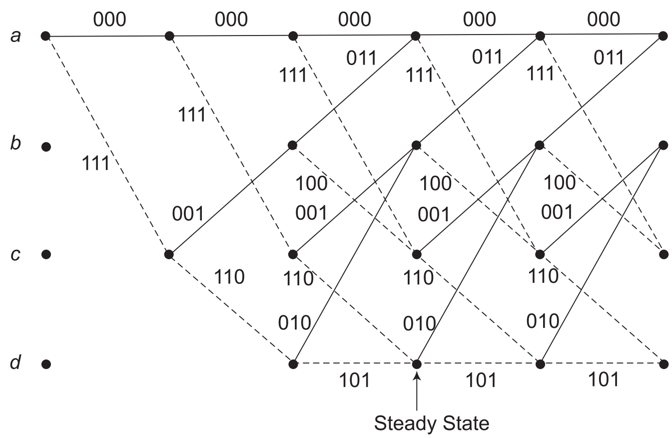


Figure 5.4.11 Trellis Diagram for Rate 1/3, K = 3 Convolution Code

Code Tree—An Expanded Version of Trellis Diagram

The *code tree* is an expanded version of trellis diagram where every path is distinct and totally different from every other path. The dimension of time can be added in the state diagram which is then called *tree diagram*, or more specifically, *code tree*. At each successive input bit, the associated branch of tree moves either upward or downward depending upon whether the input bit is 0 or 1 respectively. The corresponding output symbol (codeword) is mentioned on the branch itself. Each node in the tree corresponds to any one out of four possible states (a = 00, b = 10, c = 01, and d = 11) in the shift register. At each successive branching, the number of nodes doubles.

Interpretation

Note that the tree diagram repeats itself after the third stage. This is consistent with the fact that the constraint length $K = 3$. The output sequence at each stage is determined by the input bit and the two previous input bits. In other words, we may say that the 3-bit output sequence for each input bit is determined by the input bit and the four possible states of the shift register, denoted as a = 00, b = 01, c = 10, and d = 11.

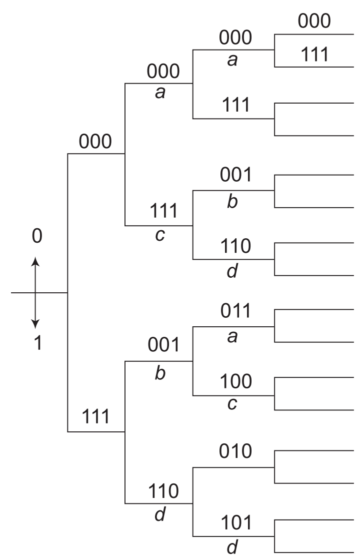


Figure 5.4.12 Code Tree for Rate 1/3, K = 3 Convolution code

SOLVED EXAMPLE 5.4.9 **Generation of Code Tree**

Create a code tree for a convolution encoder ($r = 1/2$, $k = 3$) for the input bit pattern 11011. Find the output encoded codeword sequence from the code tree.

Solution

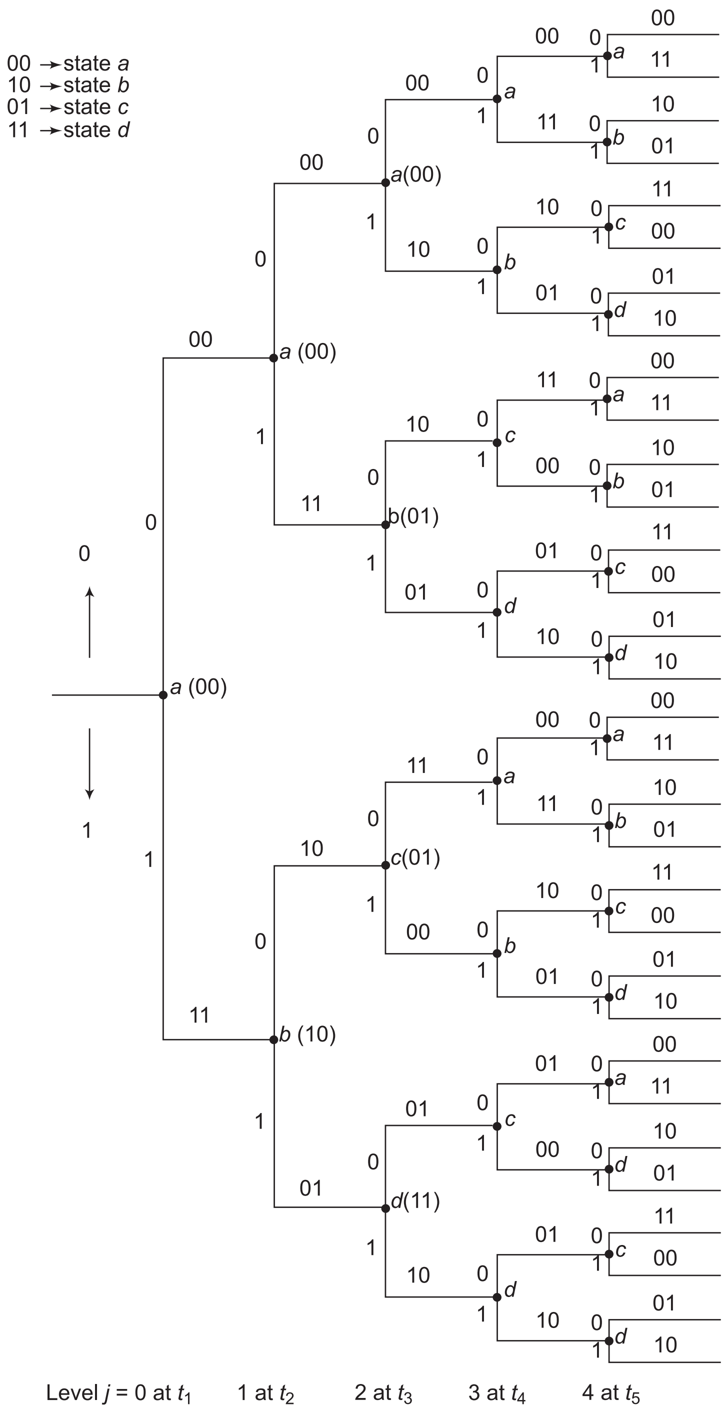


Figure 5.4.13 Code Tree for Convolution Encoder (r = 1/2, k = 3)

From the code tree, it can be easily traced that the output encoded codeword sequence is 11 01 01 00 01 for the input bit sequence 11011.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 5.4.10 Create a code tree for the convolution encoder ($r = 2/3$, $K = 2$).

Ex 5.4.11 Consider the input bit sequence 10101. Create a code tree for the convolution encoder ($r = 1/2$, $K = 3$). Determine the output-encoded codeword sequence from the code tree.

Convolution Decoding

We have seen that convolution encoding is a simple procedure. But decoding of a convolution code is much more complex task. In general, maximum likelihood decoding means finding the code branch in the code trellis that was most likely to be transmitted. Therefore, maximum likelihood decoding is based on calculating the hamming distances for each branch forming codeword. Assume a three bit message is to be transmitted. To clear the encoder, two zero-bits are appended after message. Thus, 5 bits are inserted into the convolution encoder and 10 (ten) bits are produced at the encoder output. Assume channel error probability is $p = 0.1$. After transmission through the channel, the bit sequence 10, 01, 10, 11, 00 is received. Now the question arises: what is the output of the convolution decoder, e.g., what was the most likely transmitted sequence? See trellis diagram in Figure 5.4.14.

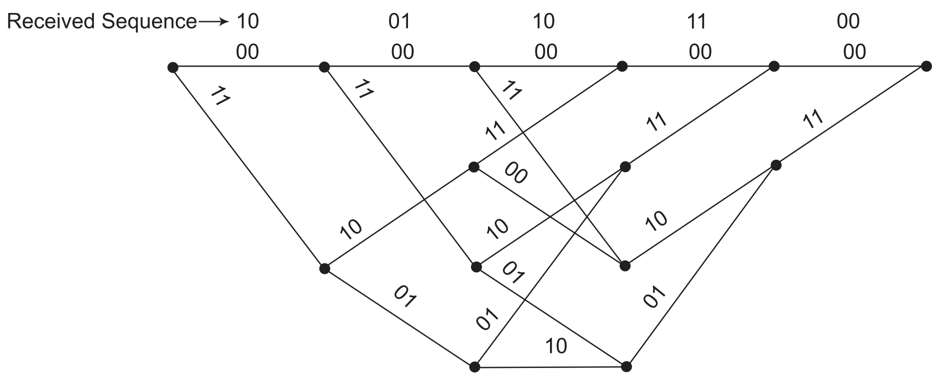


Figure 5.4.14 Maximum Likelihood Detection

Interpretation

The Maximum Likelihood algorithm is too complex to search all available paths due to requirement of end to end calculations. Viterbi algorithm performs maximum likelihood decoding by reducing its complexity. It eliminate least likely trellis path at each transmission stage and reduce decoding complexity with early rejection of unlike paths. Viterbi algorithm gets its efficiency via concentrating on survival paths of the trellis.

Sequential and Viterbi Decoding

There are two popular methods of decoding of a convolution code—sequential decoding and Viterbi decoding.

Convolution Decoding Techniques—Sequential and Viterbi

- The *sequential decoding* is a class of algorithms which is independent of encoder memory and, hence, large constraint lengths can be used. Therefore, it can achieve a desired bit error probability. Its main advantage is that decoding complexity is virtually independent from the length of the particular code although it is suboptimal. However, this exhibits unpredictable decoding latency. Its main disadvantage is that large amount of computation and decoding time is required, sometimes causing information to be lost or erased, for decoding data received in a noisy environment.

- On the other hand, the *Viterbi decoding*, conceived by Andrew Viterbi in 1967 as a decoding algorithm for convolution codes over noisy digital communication link, is an optimal algorithm. In fact, convolution encoding with Viterbi decoding is a Forward Error Correction (FEC) technique that is particularly suited to a channel in which the transmitted signal is corrupted mainly by additive white Gaussian noise (AWGN). Its main drawback is that the decoding complexity grows exponentially with the code length, thereby limiting its usage only for relatively short codes.[8]

Sequential Decoding Algorithm

The sequential decoding algorithm can be achieved by using a code tree. The leftmost node in the tree is called *origin node* that represents the start state of the encoder. There are 2^k branches leaving every node in the first L levels termed as dividing part of the code tree. The upper branch of each node represents input for the bit level 1, while the lower branch represents input for the bit level 0. After L levels, there is only one branch leaving each node. The rightmost nodes are termed *terminal nodes*.

The purpose of the sequential decoding algorithm is to find the maximum likelihood path by searching through nodes of the code tree depending on its *metric value* (a measure of the closeness of a path to the received sequence).

- In the *stack or ZJ algorithm* approach of sequential decoding, the stack or ordered list of previously examined paths of different lengths is kept in storage. Each stack entry contains a path along with its metric. Each decoding step computes the branch metrics of its 2^k succeeding branches and then updating the same till the end of the code tree. Thus, the number of stack computation may be higher than the Viterbi algorithm. This stack algorithm has certain disadvantages such as the speed factor, ratio of speed of computation to the rate of incoming data, finite stack size, time-consuming if the number of stacks is high.
- In the *Fano algorithm* approach, the decoder examines a sequence of nodes in the code tree from the origin to the terminal node and it never jumps from node to node. The metric at each node is calculated in the forward direction.

IMPORTANT!

The overall performance of the sequential decoding algorithm can be optimized by obtaining trade-off among three parameters—average computational cut-off rate of the channel, erasure probability, and undetected error probability.

Viterbi Decoding Algorithm

The Viterbi decoding algorithm is a dynamic programming algorithm that produces the maximum likelihood estimate of the transmitted sequence over a band-limited channel with intersymbol interference. It finds the most likely sequence of hidden states, called the *Viterbi path* that results in a sequence of observed events. The Viterbi algorithm operates on a state machine assumption. This means that while multiple sequences of states (paths) can lead to a given state, at least one of them is the most likely path to that state, called the *survivor path*.

Viterbi decoding compares the Hamming distance between the branch code and the received code. Path producing larger Hamming distance is eliminated after constraint length. Thus, Viterbi decoding algorithm is an efficient search algorithm that performs maximum likelihood decoding rule while reducing the computational complexity.

---

[8] The decoding procedure is referred to as an "algorithm" because it has been implemented in software form on a computer or microprocessor or in VLSI form. The Viterbi decoding algorithm is a maximum likelihood decoder, which is optimum for a white Gaussian noise channel.

In order to understand Viterbi's decoding algorithm, it is convenient to expand the state diagram of the encoder in time (i.e., to represent each time unit with a separate state diagram). The resulting structure is called a trellis diagram. Assuming that the encoder always starts in state S_0 and returns to state S_0, the first m time units correspond to the encoder's departure from state S_0, and the last m time units correspond to the encoder's return to state S_0. The basic concept of Viterbi decoding algorithm can be described in the following steps:

Step 1: Generate the code trellis at the decoder.

Step 2: The decoder penetrates through the code trellis level by level in search for the transmitted code sequence.

Step 3: At each level of the trellis, the decoder computes and compares the metrics of all the partial paths entering a node.

Step 4: The decoder stores the partial path with the larger metric and eliminates all the other partial paths. The stored partial path is called the survivor.

The trellis diagram provides a good framework for understanding decoding algorithm, as shown in Figure 5.4.15 for information bit sequence 101011, at each transition.

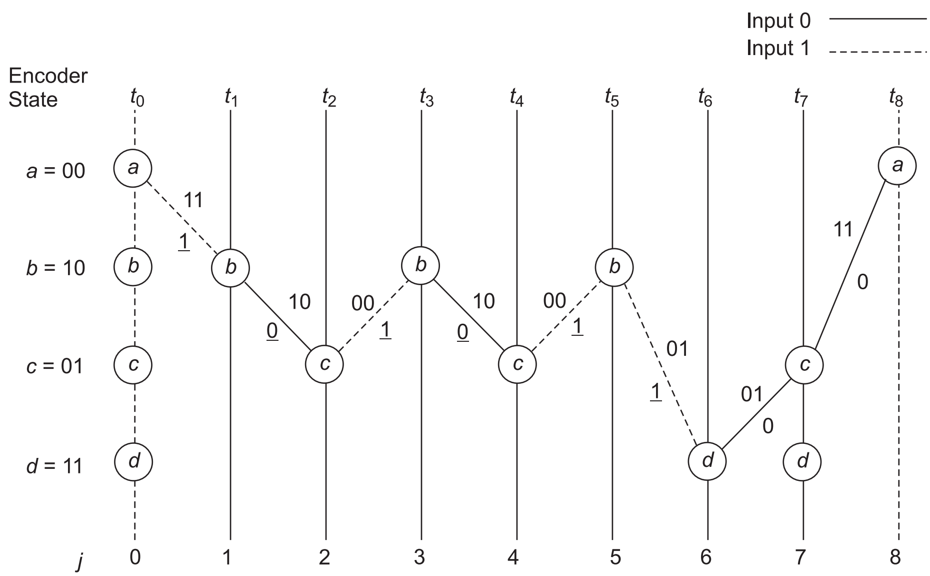


Figure 5.4.15 Trellis Diagram for Viterbi Decoding

Interpretation of Trellis Diagram for Viterbi Decoding

On the convolution decoder side, corresponding to the encoded path sequence 11 10 00 10 00 01 01 11, the decoded output is traced as path sequence *a-b-c-b-c-b-d* (ignoring the last two states as they correspond to additional input bits 0s used to flush out the registers), the decoded output bit pattern is 101011. In fact, the Viterbi decoding algorithm is a procedure which compares the received convolution-encoded sequence with all possible transmitted sequences. The algorithm chooses a path in the Trellis diagram whose code sequence differs from the received sequence in the fewest number of places. The algorithm operates by computing a metric for every possible path in the Trellis. The *metric* for a particular path is defined as the Hamming distance between the coded sequence represented by that path and the corresponding received sequence.

For each node in the trellis, the Viterbi decoding algorithm compares the two paths entering the node. The path with the lower metric is retained, and the other path is discarded. This computation is repeated for every level j of the Trellis in the range $M \leq j \leq L$, where M is the encoder's memory and L is the length of the incoming bit sequence. The path that are retained by the algorithm are called *survivors*. It may be noted that for a convolution code of constraint length $K = 3$ (that is, $M = 2$), no more than $2^{K-1} = 4$ survivor paths and their metrics will ever be stored. This relatively small list of paths is always guaranteed to contain the maximum-likelihood choice.

In case the metrics of two paths are found to be identical then either one can be chosen. Once a valid path is selected as the correct path, the decoder can recover the data bits from the output code bits. Common metric of Hamming distance is used to measure the differences between received and valid sequences.

Step-by-step Procedure of Viterbi Decoding

For an L-bit data sequence, and a decoder of memory M, the Viterbi algorithm proceeds as follows (assuming that the decoder is initially in the all-zero state at $j = 0$).

Step 1: Starting at time unit level $j = M$, the metric for the single path entering each state of the decoder is computed. The path (survivor) and its metric for each state is stored.

Step 2: The level j is then incremented by 1. The metric for all the paths entering each state is computed by adding the metric of the incoming branches to the metric of the connecting survivors from the previous level. The path with the lowest metric is identified for each state. The new path (survivor) and its metric is stored.

Step 3: If the condition $j < (L + M)$ is satisfied, Step 2 is repeated. Otherwise the process is stopped.

Thus, it can be seen that the number of operations performed in decoding L bits is $L \times 2^{n(k-1)}$, which is linear in L. However, the number of operations per decoded bit is an exponential function of the constraint length K. This exponential dependence on K limits the utilization of Viterbi algorithm as a practical decoding technique to relatively short constraint-length codes (typically in the range of 7-11). Thus, a survivor path is one that has a chance of being the most likely path, and the number of survivor paths is much smaller than the total number of paths in the Trellis.

Soft-decision Viterbi Decoding Algorithm

The soft-decision Viterbi decoding algorithm is different from hard-decision Viterbi decoding algorithm in that the soft-decision Viterbi decoding algorithm cannot use Hamming distance metric due to its limited resolution. A distance metric, known as *Euclidean distance*, provides the required resolution. In order to use distance metric, the binary number are transformed to the octal numbers—binary 1 for octal number 7 and binary 0 for octal number 0. The Euclidean distance, d_1 between the noisy point $\{x_2, y_2\}$ and the noiseless point $\{x_1, y_1\}$ on a soft-decision plane can be computed from the expression, $d_1 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$, where $\{x_1, y_1\}$ is the point of origin in the octal plane as (0, 0). For example, the Euclidean distance d_1 between the noisy point {6, 5} and the noiseless point {0, 0} on a soft-decision plane will be

$$d_1 = \sqrt{(6-0)^2 + (5-0)^2} = \sqrt{61}$$

Similarly, the Euclidean distance, d_2 between the noisy point {6, 5}and the noiseless point {7, 7} will be $d_2 = \sqrt{(6-7)^2 + (5-7)^2} = \sqrt{5}$.

Thus, the process of soft-decision Viterbi decoding algorithm is same as hard-decision Viterbi decoding algorithm except that Euclidean distance metric is used in place of the Hamming distance metric.

SOLVED EXAMPLE 5.4.12 **Convolution Decoding with Viterbi Algorithm**

For the convolution encoder ($r = 1/2$, $k = 3$) as given previously, draw the Trellis diagram and decode the received encoded bit sequence 01 00 01 00 00.

Solution Figure 5.4.16 shows the Trellis diagram using Viterbi decoding algorithm for the given encoded bit sequence.

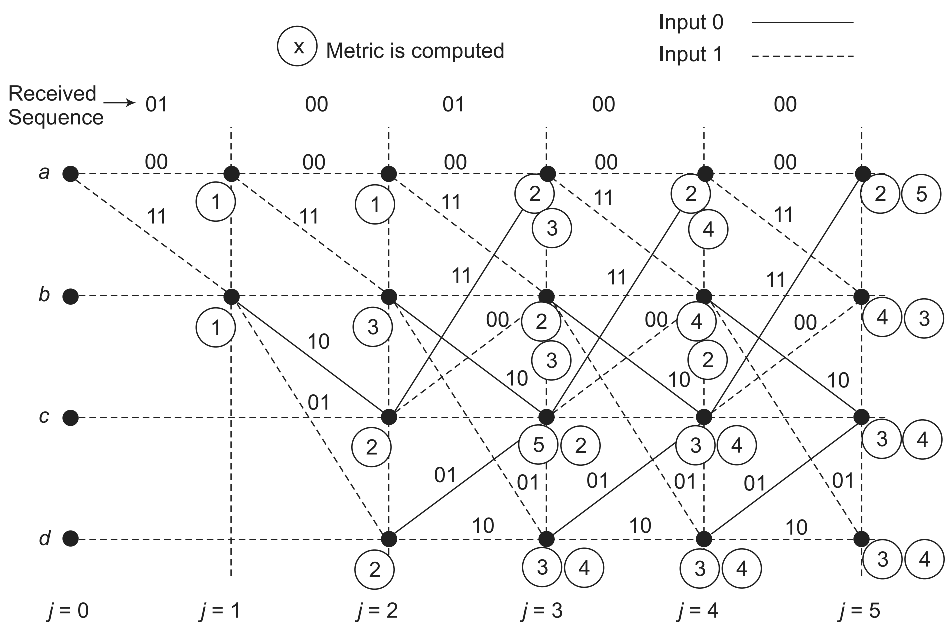


Figure 5.4.16 Trellis Diagram for Viterbi Decoding Algorithm

SOLVED EXAMPLE 5.4.13 **Viterbi Decoding**

For the input data 1 1 0 1 1, the codeword generated by convolution encoder is 11 01 01 00 01. If the received codeword is 11 01 01 10 01, then draw the Trellis diagram and decode the received data.

Solution Figure 5.4.17 shows the Trellis diagram using Viterbi decoding algorithm for the given encoded bit sequence 11 01 01 00 01.

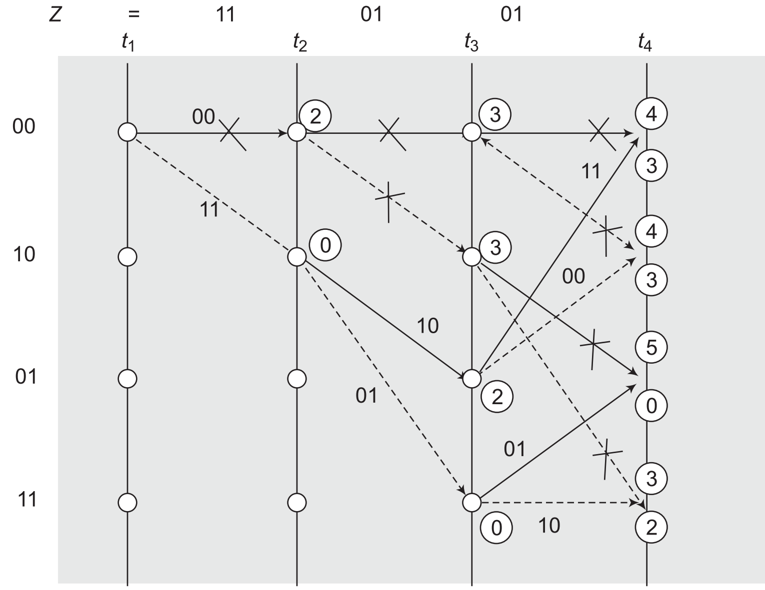


Figure 5.4.17 Trellis Diagram for Viterbi Decoding Algorithm

Figure 5.4.18 shows the decoded bits following Viterbi decoding algorithm procedure for the Trellis diagram given in Figure 5.4.17.

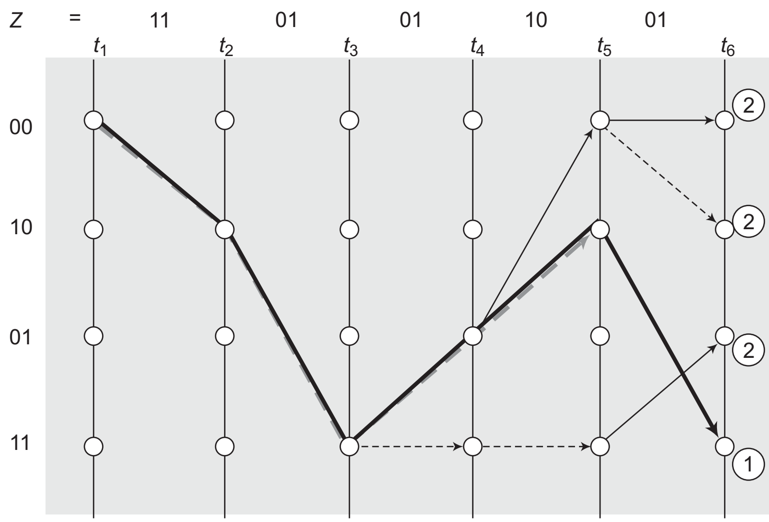


Figure 5.4.18 Decoded bit sequence as 11 01 01 00 01

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 5.4.14 Draw the Trellis diagram and decode the received encoded sequence 11 10 11 00 01 10. Show the paths of minimum path metric and two likely paths or survivor paths and find the corresponding message bits.

Application

Due to its high data rate, convolution code has several practical applications in digital transmission over radio and wired channels. The improved version of convolution code along with Viterbi algorithm is widely used in satellite and deep-sea communication applications. The Viterbi decoding algorithm has found universal application in decoding the convolution codes used in both GSM and CDMA digital cellular systems, IEEE 802.11 wireless LAN, satellite, dial-up modem, deep space communication, etc. It is commonly used even now in bioinformatics, speech recognition, computational linguistics, and keyword spotting.

Table 5.4.1 presents the comparison of coding gain on a Gaussian channel for various error control coding techniques discussed so far.

Table 5.4.1 Comparison of Coding Gain

S. No.	Coding Technique	Coding Gain (dB) at $P_b = 10^{-5}$	Coding Gain (dB) at $P_b = 10^{-8}$	Type of Decoding
1.	Block codes	3–4	4.5–5.5	hard decision
2.	Block codes	5–6	6.5–7.5	soft decision
3.	Convolution codes with sequential decoding	4–5	6–7	hard decision
4.	Convolution codes with sequential decoding	6–7	8–9	soft decision
5.	Convolution codes with Viterbi decoding	4–5.5	5–6.5	hard decision

Self-Assessment Exercise linked to LO 5.4

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/112

Q5.4.1 How is the code rate of a convolutional coder different from that of a linear block code?

Q5.4.2 Describe the characteristics of State and Trellis diagrams in context of convolution-coding techniques, and their applications.

Q5.3.3 Give significance of code tree. Under what situation is a Trellis diagram preferred over a code tree?

Q5.3.4 The received codeword of binary convolution encoder (3, 1, 2) follows the trellis diagram as 000, 111, 110, 010, 100, 011, 001, 000 as survival path. Draw the Trellis diagram and find the information code after decoding.

Q5.3.5 Outline the error-detecting and error-correcting capabilities of convolution codes.

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 5.4: *Illustrate the concept of convolution codes, Viterbi and sequential decoding algorithms.*

5.5 BURST-ERROR-CORRECTION TECHNIQUES

LO 5.5

Recap

We have discussed that block and convolution codes are designed to combat random-independent errors. The error-control coding described so far can correct individual bit errors, but not a burst error. In many situations, errors are not distributed at random but occur in bursts. Burst errors means two or more consecutive bits in error within a given data sequence. Burst errors are no longer characterized as single randomly distributed independent bit errors. For example, scratches, dust or fingerprints on a compact disc (CD) introduce errors on neighbouring databits.

What We Discuss Here ...

- We begin with the process of dispersing the burst error into multiple individual errors, known as *interleaving*. Interleaving does not introduce any redundancy into the information data sequence. We describe two popular types of interleaving—block interleaving and convolution interleaving.
- Then, we introduce another effective way of dealing with burst errors—Reed-Solomen (RS) codes which makes it possible to employ long codes. Cross-Interleaved Reed–Solomon Codes (CIRC) are particularly well suited for detection and correction of burst errors and data erasures.
- Finally, we have a look at Turbo codes which use a Turbo interleaver to separate two convolution encoders. Turbo codes provide significant improvements in the quality of data transmission over a noisy channel.

5.5.1 Interleaving

Essence of Interleaving

Most well-known error control codes have been designed for additive white Gaussian noise (AWGN) channels, i.e., the errors caused by the channel are randomly distributed and statistically independent. Thus, when the channel is AWGN channel, these codes increase the reliability in the transmission of information. If the channel exhibits burst-error characteristics (due to the time-correlated multipath fading), these error clusters are not usually corrected by these codes. One of the possible solution is to use an interleaver. Its objective is to interleave the coded data in such a way that the bursty channel is transformed into a channel having independent errors and thus a code designed for independent channel errors (short burst) can be used. However, it suffers from high latency (the entire interleaved block must be received before the critical data can be returned).

Interleaving is a process to rearrange code symbols so as to spread bursts of errors over multiple codewords that can be corrected by random error correction codes. By converting bursts errors into randomlike errors, interleaving thus becomes an effective means to combat burst errors. *Interleaving* is the process of dispersing the *burst error* into multiple individual errors which can then be detected and corrected by error control coding. The main idea is to mix up the code symbols from different codewords so that when the codewords are reconstructed at the receiving end, burst errors encountered in the transmission are spread across multiple codewords. Consequently, the errors occurred within one codeword may be small enough to be corrected by using a simple random error correction code. Since interleaving does not have error-correcting capability, so the information data bits are first encoded, followed by interleaving. It may be noted that interleaving does not introduce any redundancy into the information data sequence. It does not require any additional bandwidth for transmission. However, it may introduce extra delay in processing information data sequence.

ATTENTION There are many types of interleavers such as block interleaver, semi-random and random interleaver, pseudorandom (Turbo) interleaver, circular interleaver, odd-even interleaver, and near-optimal interleaver. Each one has its advantages and drawbacks in the context of noise. Block interleavers, convolutional interleavers, and Turbo interleavers are commonly used interleavers in wireless communication systems.

Block Interleaver

A simple interleaving method is block interleaving. In block interleaving, a block of data symbols is written row by row as a $n \times m$ matrix and read column by column. It is obvious that the interleaver requires a memory capacity of $n \times m$ symbols. A *block interleaver* accepts a predetermined block of symbols. It rearranges them without repeating or omitting any symbol in the block.

There are various types of block interleavers such as

- *General block interleaver* that uses the permutation table for selection of block of information data bits.
- An *odd-even interleaver* in which firstly the bits are left uninterleaved and encoded, but only the odd-positioned coded bits are stored. Then, the bits are scrambled and encoded, but now only the even-positioned coded bits are stored. Odd-even encoders can be used, when the second encoder produces one output bit per one input bit.
- *Algebraic interleaver* that derives a permutation table algebraically.
- *Helical scan interleaver* that fills a matrix with data row by row and then sends the matrix contents to the output in a helical manner. In other words, in a helical scan interleaver, data is written row-wise and read diagonally.
- *Matrix interleaver* that fills a matrix with data row by row and then sends the matrix contents to the output column by column. While it is very simple but it also provides little randomness.

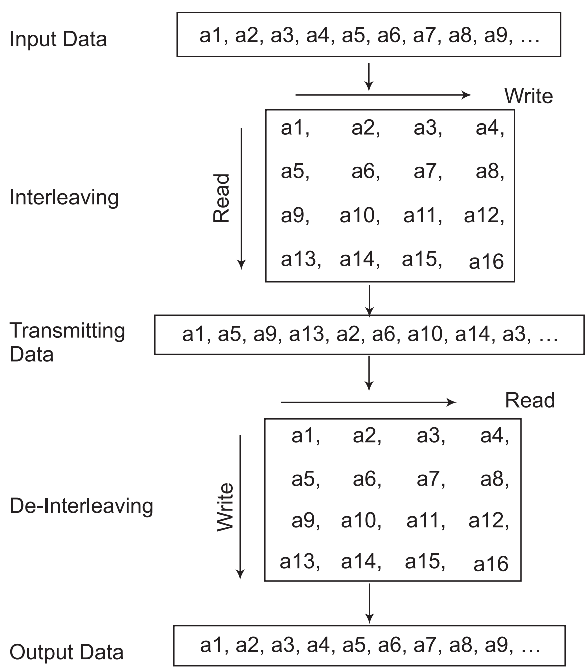


Figure 5.5.1 An Example of Block Interleaving

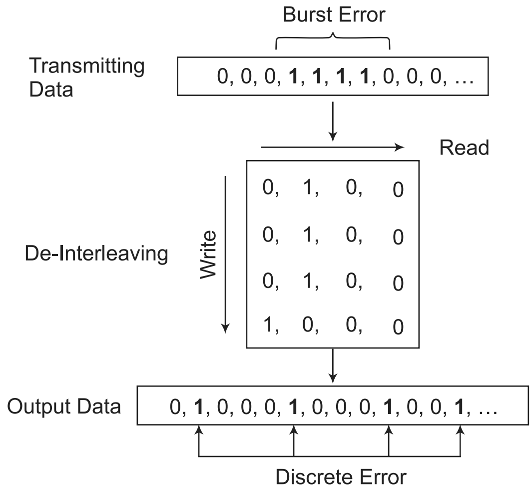


Figure 5.5.2 An Example of Burst Error Correction by Block Interleaver

- *Random interleaver* that chooses a permutation table randomly using the initial state input provided. A *pseudo-random interleaver* is defined by a pseudo-random number generator or a look-up table.

IMPORTANT!

There is no such thing as a universally best interleaver. For short block sizes, the odd-even interleaver has been found to outperform the pseudo-random interleaver, and vice versa. The choice of the interleaver has a key part in the success of the code and the best choice depends on the code design.

Interleaver used in GSM Application

As we know that the purpose of interleaving user traffic data is to improve the signal quality by distributing the effects of fading among several mobile subscribers receiving data simultaneously from base stations. For example, in GSM, the user traffic data in frames of 456 bits are interleaved into the transmitted normal bursts in blocks of 57 bits plus one flag bit. The 456 bits are produced every 20 ms.

The interleaving process specifies the method that maps the 20 ms of the traffic into the 456 bits, as shown in Figure 5.5.3.

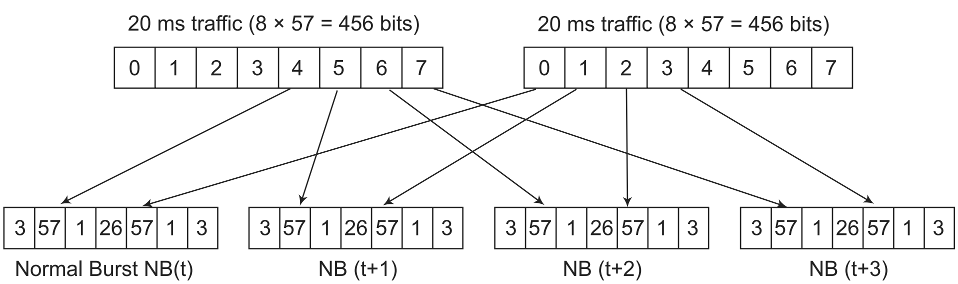


Figure 5.5.3 Interleaving Traffic Frames onto TDMA GSM Frame

Effect of Block Interleaving

To add protection against burst errors during wireless transmission, each 456-bit encoded data within each 20 ms traffic frame is divided into eight 57-bit sub-blocks forming a single speech frame, which are transmitted in eight consecutive time slots. Because each time slot can carry two 57-bit user data bits, each burst carries data from two different speech samples. The speech data are encrypted 114 bits at a time, assembled into time slots, and finally modulated for transmission. If a burst is lost due to interference or fading, channel coding ensures that enough bits will still be received correctly to allow the error correction to work satisfactorily.

SOLVED EXAMPLE 5.5.1 Block Interleaving in GSM

In GSM, the output of the convolution encoder consists of 456 bits for each input of 228 bits. How is block interleaving of encoded data implemented? Determine the delay in reconstructing the codewords corresponding to the reception of 8 TDMA frames. Comment on the results obtained.

Solution Number of data bits at the input of convolution encoder = 228 bits (Given)

Number of data bits at the output of convolution encoder = 456 bits (Given)

Hence, code rate of the convolution encoder = 456/228 = 1/2

Number of TDMA frames = 8 (Given)

Number of encoded data bits in each TDMA frame = 456/8 = 57

Therefore, the encoded 456 data bits are split into 8 uniform blocks of 57 bits each. These 57 bits in each block are spread over eight TDMA frames so that even if one frame out of five is lost due to channel burst error, the voice quality is not affected.

One TDMA frame time duration = 4.6 ms

Time taken to transmit 8 TDMA frames = 8 × 4.6 = 36.8 ms

Therefore, the delay in reconstructing the codewords corresponding to the reception of 8 TDMA frames is 36.8 ms. **Ans.**

Comment on the Results: Usually, a delay of 50 ms is acceptable for voice communication. Hence, the delay introduced due to block interleaving in GSM system does not degrade the voice quality performance while enhancing BER performance to combat channel errors.

Random Interleaving

A block of N input bits is written into the interleaver in the order in which the bits are received, but they are read out in a random manner. Each bit is read once and only once. For example, let N = 16 bits marked as a_k, where k varies from 1 to 16, as mentioned below:

Input bits: $[a_1a_2a_3a_4\ a_5a_6a_7a_8a_9a_{10}a_{11}a_{12}a_{13}a_{14}a_{15}a_{16}]$

Interleaved bits: $[a_{16}\ a_{11}\ a_2a_{13}a_3a_1a_{14}a_{10}a_6a_4\ a_9a_{12}a_8a_5a_{15}\ a_7]$

Convolution Interleaver

A *convolution interleaver* has memory, that is, its operation depends not only on current symbols but also on the previous symbols. The sequence of encoded bits to be interleaved is arranged in blocks of L bits. For each block, the encoded bits are sequentially shifted into and out of a bank of N registers by means of two synchronized input and output commutators.

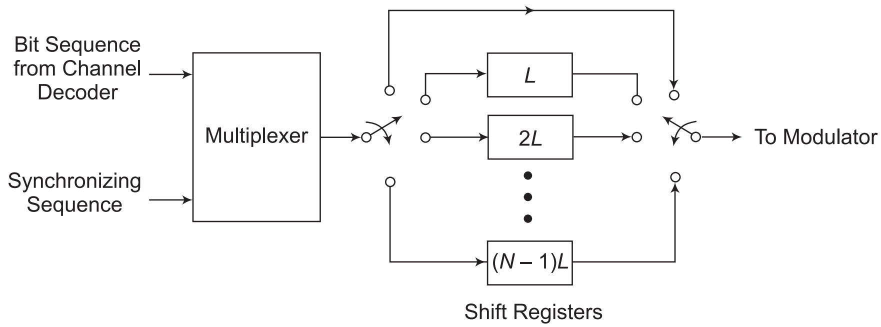


Figure 5.5.4 Block Diagram of Convolution Interleaver

Description of Convolution Interleaver

The zeroth shift register does not have any memory element to store bits. It implies that the incoming encoded symbol is transmitted immediately. Each successive shift register provides a memory capacity of L symbols more than the preceding shift register. Each shift register is scanned regularly on a periodic basis. With each new encoded symbol, the commutator switches to a new shift register. The new symbol replaces the oldest symbol in that register. After scanning the last $(N - 1)^{th}$ shift register, the commutator return to the zeroth shift register. The whole process is repeated again and again.

Convolution Deinterleaver

The *convolution deinterleaver* in the receiver also uses N shift registers and a pair of input/output commutator synchronized with those used in the interleaver. However, the shift registers are stacked in the reverse order. It may be noted that the end-to-end delay and memory requirement in case of convolution interleaver and deinterleaver is one-half that of block interleaver and deinterleaver. In practice, RAM is used instead of shift registers in the design and implementation of convolution interleaver and deinterleaver.

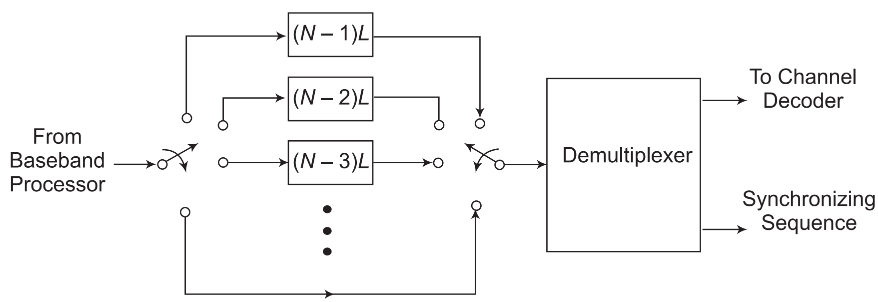


Figure 5.5.5 Block Diagram of Convolution De-interleaver

Comparison of Block and Convolution Interleaving

In comparison with a block interleaver, the convolution interleaver has the distinct advantage that for the same interleaving, the interleaving structure can be changed easily and conveniently by changing the number of lines and/or the incremental number of elements per line. The CD uses a more effective interleaver, a periodic or convolutional interleaver, known as a *cross-interleaver*. Before transmission the symbols of the codewords are multiplexed over delay lines with differing delays, combined (demultiplexed) and sent to the channel.

Application

In the wireless and mobile radio channel environment, the burst error occurs quite frequently. In order to correct the burst error, Interleaving exploits time diversity without adding any overhead in wireless digital cellular communication systems such as GSM, IS-95 CDMA, 3G cellular systems.

5.5.2 RS Codes

RS Codes—Historical Background

Irving Reed (1923–2012) is an American mathematician and engineer who is best known for co-inventing a class of algebraic codes known as Reed–Solomon codes (RS codes) in collaboration with Gustave Solomon (1930–1996). RS codes are seen as a special case of the larger class of BCH codes, regarded as cyclic BCH codes, that an efficient decoding algorithm gave them the potential to their widespread applications. Today RS codes are widely used in many applications that involve data transmission, such as wireless computer networks, telephony, GSM, GPRS, UMTS, digital video broadcasting: DVB-T, DVC-C, and data storage including hard disk drives (HDD) in computers. Memory cards in cameras and telephones, and optical storage like Compact Discs (CD), Digital Versatile Discs (DVD) and Blu-ray Discs (BD) also use Reed–Solomon codes.

RS Codes—A Subclass of Non-binary BCH Codes

Reed–Solomen (RS) codes are widely used subclass of non-binary BCH codes. The use of *non-binary codes* (codes based on symbols rather than on individual bits) is another effective way of dealing with burst errors other than interleaving. The RS code is important because an efficient hard-decision decoding algorithm is available which makes it possible to employ long codes. So the RS code encoder differs from binary encoders in that it operates on *multiple bits* (referred to as symbols) rather than on individual bits. An RS code can be regarded as a primitive BCH code of length n.

Parameters of RS Codes

With RS codes, data are processed in small blocks of m bits, called symbols. An (n, k) RS code has the following parameters:

- Symbol length = m bits/symbols (m is an integer power of 2)
- Block length, $n = (2^m - 1)$ symbols, or $m(2^m - 1)$ bits
- Data length or message size = k symbols
- Error-correcting capability in t symbols where $t = r/2$
- The size of check code, $(n - k) = 2t$ symbols = $m(2t)$ bits
- Minimum Hamming distance, $d_{\min} = [(n - k) + 1] = (2t + 1)$ symbols
- Number of correctable symbols in error, $t = \dfrac{(n-k)}{2}$

Therefore, an RS code may be described as an (n, k) code where $n < (2^m - 1)$, and $(n - k) > 2t$. RS codes operate on multi-bit symbols rather than on individual bits like binary codes. For example, in the RS (255, 235) code, the encoder groups the message into 235 8-bit symbols and adds 20 8-bit symbols of redundancy to give a total block length of 255 8-bit symbols. In this case, 8% of the transmitted message is redundant data. In general, due to decoder constraints, the block length cannot be arbitrarily large. The number of correctable symbol errors (t), and block length (n) is set by the user.

The code rate (efficiency) of a code is given by k/n, where k is the number of information (message) symbols per block, and n is total number of code symbols per block. Codes with high code rates are generally desirable because they efficiently use the available channel for information transmission. RS codes typically have rates greater than 80% and can be configured with rates greater than 99% depending on the error correction capability needed.

An Example of RS Code

As an example, let $m = 8$, then $n = 2^8 - 1 = 255$ symbols in a codeword.

If we require $t = 16$ then we have $r = 2t = 32$.

Thus, $k = (n - r) = (255 - 32) = 223$.

RS Encoding Algorithm

The *RS encoding algorithm* expands a block of k symbols to n symbols by adding $(n - k)$ redundant check symbols. The encoder for an RS (n, k) code divides the incoming binary data streams into blocks, each of $(k \times m)$ bits long. Each block is treated as k symbols with each symbol having 8 bits. Typically $m = 8$, RS codes are well suited for burst error correction.

Properties and Benefits of RS Codes

RS block codes have four basic properties which make them powerful codes for digital communications:

- The RS codes with very long block lengths tend to average out the random errors and make block codes suitable for use in random error correction.
- RS codes are well-matched for the messages in a block format, especially when the message block length and the code block length are matched.
- An RS decoder acts on multi-bit symbols rather than on single bits. Thus, up to eight bit-errors in a symbol can be treated as a single symbol error. Strings of errors, or bursts, are therefore handled efficiently.
- The complexity of the decoder can be decreased as the code block length increases and the redundancy overhead decreases. Hence, RS codes are typically large block length, high code rate, codes.

RS codes make highly efficient use of redundancy, and block lengths and symbol sizes can be easily adjusted to accommodate a wide range of message sizes.

RS code is also called *maximum distance separable code* because the minimum distance is always equal to the design distance of the code. Efficient coding techniques are available for RS codes.

Concatenation of Codes

However, interleaving and RS codes are not particularly effective in dealing with random errors which affect only a single bit or only a small number of consecutive bits. When we must contend with both random and burst errors it is useful to cascade coding known as *concatenation.* In cross-interleaved RS code (CIRC), cross-interleaving separates the symbols in a codeword, as codewords undergo a second encoding on a symbol basis. It becomes less likely that a burst from the outer decoder disturbs more than one Reed–Solomon symbol in any one codeword in the inner code. Since the information in CIRC is interleaved in time, errors that occur at the input of the error correction system are spread over a large number of frames during decoding. The error correction system can correct a burst of thousands of data bits because the errors are spread out by interleaving. If more than the permitted amount of errors occur, they can only be detected.

The audio signal degrades gracefully by applying interpolation or muting the output signal. Key parameters to the success of concealment of errors are thoughtful positioning of the left and right audio channels as well as placing audio samples on even- and odd-numbered instants within the interleaving scheme. There are several interleaved structures used in the CD which allow for error detection and correction with a minimum of redundancy.

Application

Reed-Solomen codes in association with efficient coding techniques make highly efficient use of redundancy, and symbol sizes and block lengths which can be easily adjusted to accommodate a wide range of information data sizes. RS codes, which are BCH codes, are used in applications such as space-craft communications, compact disc players, disk drives, and two-dimensional bar codes.

SOLVED EXAMPLE 5.5.2 **Parameters of RS Codes**

For (31, 15) Reed–Solomon code, determine the number of bits per symbol, block length in terms of bits, and minimum distance of the code.

Solution We know that block length, $n = (2^m - 1)$ symbols; where m is number of bits per symbol

$\Rightarrow$ $2^m = n + 1 = 31 + 1 = 32$ (n = 31… given)

$\Rightarrow$ $2^m = 2^5$

Therefore, number of bits per symbol, m = 5 bits **Ans.**

Now, we know that block length, $n = m(2^m - 1)$ bits

$\Rightarrow$ $n = 5(2^5 - 1) = 5 \times 31 = 155$ bits **Ans.**

Minimum Hamming distance, $d_{\min} = [(n - k) + 1]$ symbols; (n = 31, k = 15… given)

$\Rightarrow$ $d_{\min} = [(31 - 15) + 1] = 17$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 5.5.3 For (31, 15) Reed–Solomon code, how many symbols in error can the code correct? If data burst containing 5–bits is in error, can this code correct it?

5.5.3 Turbo Codes

Turbo Codes—Historical Background

In 1948, Shannon formulated that the AWGN channel's signal-to-noise ratio (S/N) determines the capacity C, in the sense that for code rates $R_c < C$ we can have error-free transmission. For each code rate R_c we can compute the Shannon limit. However, it is difficult to approach the Shannon limit by classical methods. But Gallager (1961), Tanner (1981), and Berrou, Glavieux, and Thitimajshima invented Turbo codes in 1993. Turbo codes are based on concatenated coding.

Concatenated Codes

The theory of error correcting codes has presented a large number of code constructions with corresponding decoding algorithms. However, for applications where very strong error correcting capabilities are required, these code constructions result in far complex decoder algorithms. The way to combat this is to use concatenated coding, where two (or more) constituent codes are used after each other or in parallel—usually with some kind of interleaving. A *concatenated code* consists of two separate codes which are combined to form a larger code. Concatenated coding techniques were first proposed by Forney as a method for achieving large coding gains by combining two or more relatively simple building block or component codes, sometimes called constituent codes. The resulting codes had the error-correction capability of much longer codes, and they were endowed with a structure that permitted relatively easy to moderately complex decoding. The most popular of these techniques consists of a Reed–Solomon (RS) outer code followed by a convolutional inner code. A serial concatenation of codes is most often used for power-limited systems such as transmitters on deep-space probes.

Examples of Concatenated Coding Schemes

The convolution code is well-suited for channels with random errors, and the Reed–Solomon code is well suited to correct the bursty output errors common with a Viterbi decoder. An interleaver is used to spread the Viterbi output burst errors across multiple RS codewords. Figure 5.5.6 shows an example of serial concatenation.

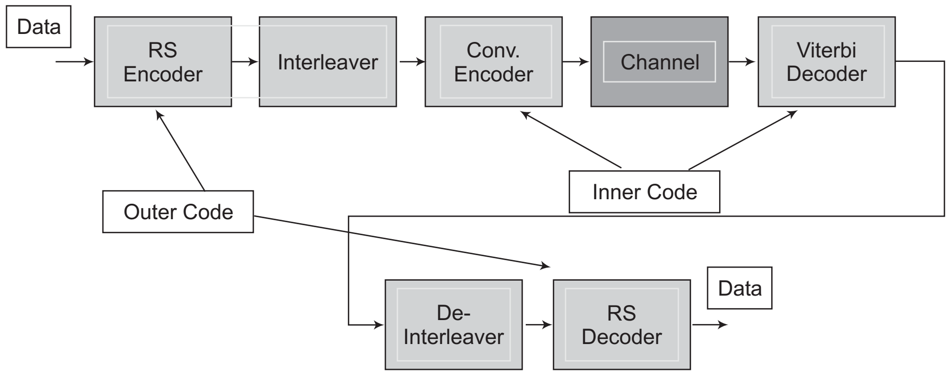


Figure 5.5.6 An Example of Serial Concatenated Coding Scheme

Similarly, we can have parallel concatenation coding scheme, as depicted in Figure 5.5.7.

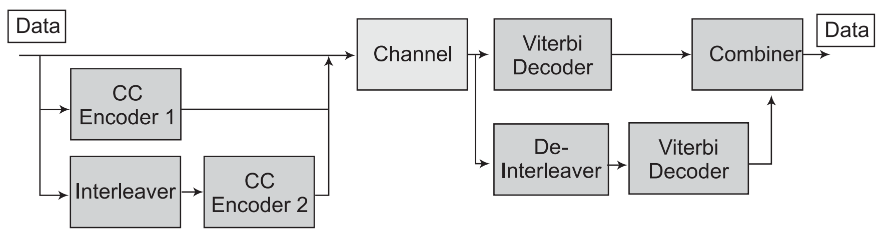


Figure 5.5.7 An Example of Parallel Concatenated Coding Scheme

Turbo codes are described as parallel-concatenated systematic convolution codes. The encoders operate on the same set of input bits, rather than one encoding the output of the other. This is one reason for excellence performance.

Turbo Codes

A turbo code can be thought of as a refinement of the concatenated encoding structure plus an iterative algorithm for decoding the associated code sequence. *Turbo codes* are inherently large block codes which use a Turbo interleaver to separate two convolution encoders. The term 'Turbo' refers to the decoding process which is analogous to the principle of 'turbo engine' involving feedback. Turbo codes are constructed by using two or more component codes (usually systematic convolution codes having identical rate 1/2) on different interleaved versions (Pseudo-random interleavers; length of the code is determined by the interleaver) of the same information sequence. In turbo codes, the interleaver is used to permute the input bits such that the two encoders are operating on the same set of input bits, but different input sequences. Whereas, for conventional codes, the final step at the decoder yields hard-decision decoded bits for a concatenated technique such as a turbo code to work properly, the decoding algorithm (usually Soft-output iterative decoder) should not limit itself to passing hard decisions among the decoders. To best exploit the information learned from each decoder, the decoding algorithm must effect an exchange of soft decisions rather than hard decisions. For a system with two component codes, the concept behind turbo decoding is to pass soft decisions from the output of one decoder to the input of the other decoder, and to iterate this process several times so as to produce more reliable decisions.

In general, Turbo code is systematic, i.e., input bits also occur in the output and the interleaver is pseudo-random in nature, also known as *Recursive Systematic Convolution (RSC) coder*. The block size of Turbo codes is determined by the size of the Turbo interleaver. The operation of *Turbo encoder* is based on the use of a pair of convolution encoders, separated by Turbo interleaver. Convolution codes can be used to encode a continuous stream of data, but in this case we assume that data is configured in finite blocks—corresponding to the interleaver size.

Salient Features of Turbo coding

Turbo codes are a recent development in the field of forward-error-correction channel coding. The codes make use of three simple ideas: parallel concatenation of codes to allow simpler decoding; interleaving to provide better weight distribution; and soft decoding to enhance decoder decisions and maximize the gain from decoder interaction.

- Parallel concatenated codes.
- Distance properties: Not exceptionally high minimum distance but few codewords of low weight.
- Trellis complexity: Usually extremely high.
- Decoding: Suboptimum (but close to maximum likelihood) iterative (turbo) decoding.
- Performance: Low error probability at signal-to-noise ratio close to the Shannon limit.

How are Turbo Codes Formed?

A turbo code is formed from the parallel concatenation of two codes separated by an interleaver. The two encoders used are normally identical. The code is in a systematic form, i.e., the input bits also occur in the output. The interleaver reads the bits in a pseudo-random order. The choice of the interleaver is a crucial part in the turbo code design. The task of the interleaver is to scramble bits in a pseudo-random, albeit predetermined fashion. This serves two purposes. Firstly, if the input to the second encoder is interleaved, its output is usually quite different from the output of the first encoder. This means that even if one of the output codewords has low weight, the other usually does not, and there is less chance of producing an output with very low weight. Higher weight is beneficial for the performance of the decoder. Secondly, since the code is a parallel concatenation of two codes, the divide-and-conquer strategy can be employed for decoding. If the input to the second decoder is scrambled, also its output will be different, or uncorrelated from the output of the first encoder. This means that the corresponding two decoders will gain more from information exchange.

Figure 5.5.8 illustrates the functional block diagram of a Turbo encoder.

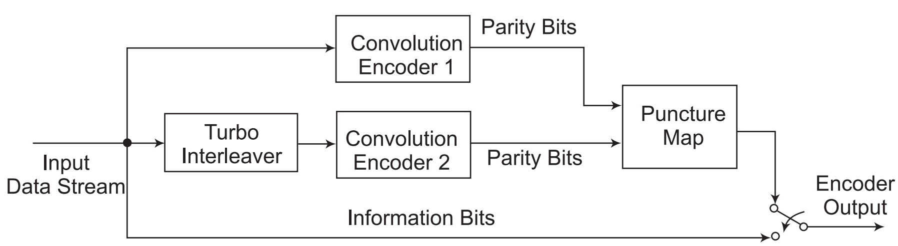


Figure 5.5.8 Block Diagram of a Turbo Encoder

Interpretation

Encode information by a systematic encoder which is usually a recursive systematic rate 1/2 convolutional encoder, followed by reordering of information bits. Encode permuted information bits again, using a recursive systematic encoder (may be the same). Delete the systematic bits this time. The Turbo interleaver rearranges the information bits which are encoded by the convolution encoder 2. The parity bits generated by two encoders are punctured in a repeating pattern. The information bits and the punctured parity bits are then multiplexed to generate the encoded output bits.

Implications

If we transmit the systematic part from both encoders, this would just be a repetition, and we know that we can construct better codes than repetition codes. The information part should only be transmitted from one of the constituent codes, so if we use constituent codes with rate 1/2 the final rate of the Turbo code becomes 1/3. If more redundancy is needed, we must select constituent codes with lower rates. Likewise, we can use puncturing after the constituent encoders to increase the rate of the turbo codes. Parity or information bits can be punctured to adjust the rate. We can add more interleavers and convolution encoders to lower the rate. Large information blocks give better distance properties as well as working decoding algorithm. Simple convolution codes are the best for moderate BERs. Interleaver design is difficult, and there is no known technique to design the best one. Design criteria are implementation complexity, performance at low S/N (pseudorandom-like) and performance at high S/N (high minimum distance). However, there may be delay in decoding.

Classical coding approach is to maximize the minimum distance. But new approach is to have few codewords with low weights. It may be recalled that

- In a feed-forward convolution encoder, a low-weight codeword is usually generated by a low-weight input sequence.
- In a feedback convolution encoder, a low-weight codeword is usually generated by an input information sequence that is a multiple of the feedback polynomial.

Turbo Decoder

In a typical communications receiver, a demodulator is often designed to produce soft decisions, which are then transferred to a decoder. The error-performance improvement of systems utilizing such soft decisions compared to hard decisions are typically approximated as 2 dB in AWGN environment. Such a decoder could be called a *soft input/hard output* decoder, because the final decoding process out of the decoder must terminate in bits (hard decisions). With turbo codes, where two or more component codes are used, and decoding involves feeding outputs from one decoder to the inputs of other decoders in an iterative fashion, a hard-output decoder would not be suitable. That is because hard decisions into a decoder degrades system performance (compared to soft decisions). Hence, what is needed for the decoding of turbo codes is a *soft input/soft output* decoder. The iterative detection in *Turbo decoder* involves the use of feedback around a pair of convolution decoders, separated by a Turbo deinterleaver and a Turbo interleaver.

Figure 5.5.9 illustrates the functional block diagram of a Turbo decoder.

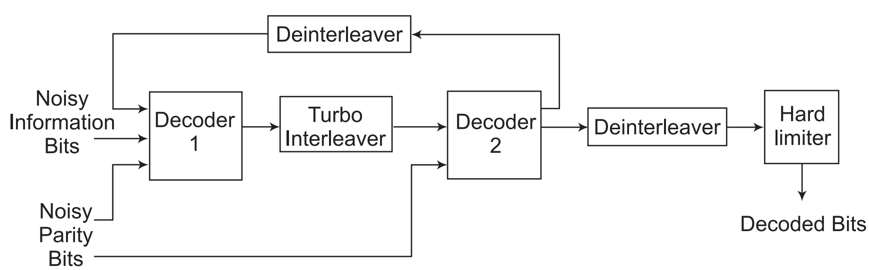


Figure 5.5.9 Block Diagram of a Turbo Decoder

Interpretation

The first decoder, the Turbo interleaver, the second decoder, and the deinterleaver constitute a single-hop feedback system. This arrangement makes it possible to iterate the decoding process in the receiver many times so as to achieve desired performance. The inputs to the first decoder are received noisy information and parity bits as well as the noisy parity bits fed back after processing by second decoder and deinterleaver. The Viterbi decoding algorithm is an optimal decoding method for minimizing the probability of sequence error. Unfortunately, it is not suited to generate the soft-decision output for each decoded bit.

A *Soft-In-Soft-Out (SISO)* decoder receives as input a soft (i.e., real) value of the signal. The decoder then outputs for each data bit an estimate expressing the probability that the transmitted data bit was equal to one. In the case of turbo codes, there are two decoders for outputs from both encoders. Both decoders provide estimates of the same set of data bits, though in a different order. If all intermediate values in the decoding process are soft values, the decoders can gain greatly from exchanging information, after appropriate reordering of values. Information exchange can be iterated a number of times to enhance performance. At each round, decoders reevaluate their estimates, using information from the other decoder, and only in the final stage hard decisions will be made, i.e., each bit is assigned the value 1 or 0. Such decoders, although more difficult to implement, are essential in the design of turbo codes.

Turbo Interleavers

Now comes the question of the interleaving. A first choice would be a simple block interleaver, i.e., to write by row and read by column. However, two input words of low weight would give some very unfortunate patterns in this interleaver. Input patterns which produce low-weight words in one component code should map through the interleaver to patterns which produce high-weight words in the other component code. Interleavers with traditional structure is usually not good for turbo codes. Interleavers with a random-like structure achieve the desired objective to a larger extent. Pseudorandom interleavers (i.e., to read the information bits to the second encoder in a random but fixed order) which are with constraints on spreading properties, and with additional constraints based on the particular component encoders, can provide good results. But such random-like interleavers may be hard to implement in an efficient manner. Still the pseudorandom interleaver is superior to the block interleaver, and the pseudo-random interleaving is one of the standard for the turbo codes. *Dithered Relative Prime (DRP)* and *Quadratic Permutation Polynomial (QPP)* interleavers are easy to implement and have very good properties as well.

Comparison of Turbo and Convolution Codes

For low S/N conditions, Turbo codes exhibit better performance than traditional convolution codes. Turbo codes and Convolution codes perform better with soft decisions but convolution codes can work with hard decisions also. In Turbo codes, the BER drops very quickly in the beginning but eventually settles down and decreases at a much slower rate. Turbo codes are decodable, and hence they are of more practical importance.

Applications of Turbo Codes

Turbo codes provide significant improvements in the quality of data transmission over a noisy channel. They improve the error performance of bandwidth-limited channels without any increase in bandwidth. Turbo codes gets very closer to Shannon limit.

Nowadays, Turbo codes are used in many commercial applications, including both third generation cellular systems such as UMTS and cdma2000 for high speed data rate applications. The UMTS turbo encoder in which the starting building block of the encoder is the simple convolutional encoder. This encoder is used twice, once without interleaving and once with the use of an interleaver. In order to obtain a systematic code, desirable for better decoding, a systematic output is added to the encoder. Secondly, the second output from each of the two encoders is fed back to the corresponding encoder's input. The resulting turbo encoder produces one systematic output bit and two parity bits for each input bit it.

The performance of a Turbo encoder is nearly optimal for high S/N ratio and converges quickly. For low S/N, it takes relatively more iterations to converge and the error rate reduces with more number of iterations. Nowadays, turbo codes are popular in cellular and deep space communications.

Application

Self-Assessment Exercise linked to LO 5.5

Q5.5.1 "Interleaving exploits time diversity without adding any overhead bits in wireless digital cellular communication systems". Comment on this statement with illustration of suitable interleaving technique.

Q5.5.2 Distinguish between block interleaving and convolution interleaving. Which one is used in digital cellular communication systems?

Q5.5.3 In a GSM TDMA frame, there are 57 encoded user data bits per time slot. Create block interleaved traffic frames.

Q5.5.4 Let the symbol length in an RS code be 16. Determine the block length and data length for 32 correctable symbols in error.

Q5.5.5 Can the block size of a Turbo code be determined by the size of the Turbo interleaver in the Turbo encoder? List the advantages of Turbo codes.

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/120

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 5.5: *Outline burst-error correction codes such as Interleaving, RS codes, and Turbo codes*.

Key Concepts

- average codeword length
- block codes
- block interleaving
- burst-error-correction codes
- channel coding
- coding efficiency
- code redundancy
- convolution codes
- convolution interleaving
- cyclic codes
- entropy coding
- error-control channel coding
- Huffman coding
- LDPC codes
- Lempel–Ziv codes
- linear block coding
- prefix code
- sequential decoding algorithm
- Shannon–Fano source codes
- source-coding theorem
- Turbo codes
- Viterbi decoding algorithm

Learning Outcomes

LO 5.1

- In digital communication systems, the source coding can be applied to transfer maximum possible source information to the receiver through discrete communication channel.
- The average length of the codeword of an optimum source code is the entropy of the source itself.
- The source-coding theorem (Shannon's first theorem) specifies that the average length of codeword per source symbol cannot be smaller than the entropy of the source.

LO 5.2

- Shannon–Fano source coding is a sub-optimal (that does not achieve the lowest possible expected codeword length) source-coding technique for constructing a prefix code with fairly efficient variable-length codewords based on a set of discrete source symbols and their probabilities.
- Huffman coding uses fewer bits to represent more frequently occurring message symbols or bit patterns and more bits to represent those that occur less frequently.
- The Huffman source code is uniquely decodable. If a long sequence of Huffman-coded messages is received, it can be decoded in one way only without any ambiguity.
- Lempel–Ziv codes use fixed-length codes to represent a variable number of source symbols.

LO 5.3

- Some noise is present in all communication systems, so errors will occur known as channel error. Errors can be detected and corrected, within specified limits, by adding redundant information.
- Errors can be categorized as a single-bit error, a multiple-bit error, or a burst error. Error control is both error detection and error correction.
- Redundancy is the concept of appending extra bits per data unit for use in error detection. CRC is the most effective redundancy checking technique which appends a sequence of redundant bits computed from modulo-2 division to the data unit.
- In systematic code, such as linear block code, the coded message has original uncoded words followed by parity bits while in non-systematic codes (Hamming, cyclic, BCH, Hadamard, LDPC), it is not necessary.
- Hamming code is the first class of (n, k) linear block code devised for error control. The random-error-detecting as well as error-correcting capability of a block code is determined with minimum Hamming distance.
- BCH code is one of the most important and powerful class of random-error-correcting cyclic codes which generalized the Hamming code for multiple error correction. Significant error correction is possible with Hadamard code.
- Typically, LDPC codes are non-cyclic linear block codes with code lengths are larger than 1000 bits. Due to their excellent error-correcting capability, they have been used by broadband wireless and DVB standards.

LO 5.4

- Errors in data communications can be detected and corrected through forward error-correction techniques such as convolution coding. Most convolution codes are designed to combat random independent errors.
- Convolution encoder accepts a continuous sequence of data bits and maps them into an output data sequence, adding redundancies in the convolution process.
- Convolution encoders can be completely characterized by the state diagram, Trellis diagram and Code tree.
- Convolution encoding is a simple procedure but decoding of a convolution code (by sequential or Viterbi algorithms) is much more complex task.

LO 5.5

- The basic function of an interleaver (block or convolution) is to protect the transmitted data from burst errors.
- Reed–Solomen (RS) codes use codes based on symbols rather than on individual bits. It is another effective way of dealing with burst errors other than interleaving.
- The operation of a Turbo encoder is based on the use of a pair of convolution encoders, separated by a Turbo interleaver. The iterative Turbo detection involves the use of feedback around a pair of convolution decoders, separated by a Turbo interleaver and a Turbo deinterleaver.

Objective-Type Questions

For Interactive Quiz with answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/164

5.1 For an M-ary source, each symbol is directly encoded without using longer sequences of symbols. The average length of the codeword per message will be ______________ source entropy.
(a) greater than
(b) less than
(c) equal to
(d) independent of

5.2 By grouping longer sequences and proper source coding, one of the following is correct.
(a) Reduce delay in transmission
(b) Increase code efficiency
(c) Equate entropy with channel capacity
(d) Reduce transmission errors

5.3 Shannon's theorem emphasizes the fact that for high reliability, the
(a) symbol rate need not be very high
(b) code rate need not be zero
(c) code rate need not be more than unity
(d) code rate is independent of the error probability

5.4 For a discrete memoryless source, the entropy of the specified source is 2.33. If the average codeword length is 2.36 as computed by Shannon–Fano source-coding technique then the coding efficiency is
(a) 1.3%.
(b) 2.33%.
(c) 2.36%.
(d) 98.7%.

5.5 *Statement I:*The Huffman code is a source code in which the average length of the codewords approaches the fundamental limit set by the entropy of a discrete memoryless source.
Statement II: The Lempel–Ziv code (LZ code) uses variable-length codes to represent variable number of source symbols.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

5.6 If transmission data rate is 1 Mbps then how many bits will be in errors if the burst noise of duration of 1/100 second occurs?
(a) 10 bits
(b) 100 bits
(c) 1000 bits
(d) 10,000 bits

5.7 Using the relationship $2^m \geq (m + k + 1)$, the number of encoded bits using the Hamming code for a block of 7 data bits will be
(a) 9
(b) 11
(c) 15
(d) 21

5.8 At the CRC generator, __________________ added to the data unit after the division process.
(a) a polynomial is
(b) a CRC remainder is
(c) 0s are
(d) 1s are

5.9 *Statement I:* Single error-correcting BCH code is equivalent to a Hamming code, which is a single error-correcting linear block code.
Statement II: Low-Density Parity Check (LDPC) codes are linear block codes in which parity check matrix consists of mostly 0s and very few 1s.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

5.10 The convolution encoder can be completely characterized by the ___________ except that it cannot be used easily to track the encoder transitions as a function of time.
(a) state diagram
(b) transition table
(c) Trellis diagram
(d) code tree

5.11 *Statement I:* The sequential decoding is a class of algorithms which is independent of encoder memory and hence large constraint lengths cannot be used.
*Statement II:*Convolution encoding with Viterbi decoding is a forward error correction technique that is particularly suitable for AWGN channel.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

5.12 ________________ is the process of dispersing the burst error into multiple individual errors which can then be detected and corrected by error control coding.
(a) Convolution coding
(b) Interleaving
(c) Cascade coding
(d) Turbo encoding

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/117

Short-Answer-Type Questions

5.1 Source-encoding is the key aspect in the design of an efficient digital communication system. Give an account of the underlying concept and objectives of source encoding.

5.2 Define the term *code efficiency* for an *M*-ary discrete memoryless source. Mention clearly the basic requirements that must be satisfied by an efficient source encoder.

5.3 Distinguish between source coding and channel coding. List some important source-encoding and channel-coding techniques used in digital wireless communication systems.

5.4 Which source code is said to be an instantaneous code? Give an example of such a code. Specify the relationship between the source entropy and the average codeword length of this code. ○○●

5.5 What happens if a source code violates the Kraft–MacMillan inequality condition? Illustrate the application of the Kraft–MacMillan inequality by taking an appropriate example data. ○○●

5.6 State the reason(s) for limited applications of the Shannon–Fano source-encoding algorithm. ●●●

5.7 Which source code is also known as optimum code or the minimum redundancy code? Mention the specific application areas of this code. ○○●

5.8 List the desirable conditions for *M*-ary Huffman source code. ○○●

5.9 How is Lempel–Ziv code different from Huffman code? Specify the process of encoding and decoding using the LZ algorithm. ○○●

5.10 Compare and contrast block code and convolution code as used for channel coding in digital wireless communications. ○●●

5.11 What does the CRC generator polynomial append to the data unit? How does the CRC checker know that the received data unit is having some errors? ●●●

5.12 What do you mean by a binary cyclic code? Discuss the features of encoder and decoder used for cyclic code using an $(n - k)$ bit shift register. ●●●

5.13 How can LDPC codes be represented graphically? Give step-by-step procedure for LDPC decoding using the bit-flipping algorithm. ○○●

5.14 Comment on the error-detecting and error-correcting capabilities of Hamming and convolution codes. ○●●

5.15 In the context of channel noise, which type of interleavers are more useful in wireless communication systems? ○●●

Discussion Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/113

5.1 The redundant data bits in the source symbols are reduced by applying the concepts of information theory in the source encoder. Paraphrase the specific benefit achieved with source encoding applied to a discrete memoryless source. **[LO 5.1]**

5.2 The Shannon–Fano source-encoding algorithm produces fairly efficient variable-length encoding. However, it does not always produce optimal prefix codes. On the other hand, in the Huffman source code, the average length of the codewords approaches the fundamental limit set by the entropy of a discrete memoryless source. Justify it with the help of a suitable example for same source data. **[LO 5.2]**

5.3 The general consideration in the design of a linear block code is that the number of error-control bits should be as small as possible so as to reduce the required transmitted bandwidth and also large enough so as to reduce the error rate. Discuss the trade-off between performance determining parameters of error-control coding techniques. **[LO 5.3]**

5.4 The sequential decoding algorithms is independent of convolution encoder memory and hence large constraint lengths can be used. Can it achieve desired bit error probability? Review their main drawbacks and check whether these are overcome in Viterbi decoding algorithms. **[LO 5.4]**

5.5 A convolution interleaver has memory, that is, its operation depends not only on current symbols but also on the previous symbols. Critically analyze the advantages of using convolution interleaver as compared to block interleaver. **[LO 5.5]**

Problems

For answer keys, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/118

5.1 The four symbols produced by a discrete memoryless source has probability 0.5, 0.25, 0.125, and 0.125 respectively. The codewords for each of these symbols after applying a particular source coding technique are 0, 10, 110, and 111. Calculate the entropy of the source, the average codeword length, and the code efficiency.

5.2 Consider an alphabet of a discrete memoryless source having eight source symbols with their respective probabilities as given below:

$[s_k] = [s_0 \quad s_1 \quad s_2 \quad s_3 \quad s_4 \quad s_5 \quad s_6 \quad s_7]$

$[p_k] = [0.48 \quad 0.15 \quad 0.10 \quad 0.10 \quad 0.07 \quad 0.05 \quad 0.03 \quad 0.02]$

Determine the following:

(a) Entropy of the specified discrete memoryless source.
(b) Maximum and average length of the codewords for the simple binary encoding technique.
(c) Efficiency of simple binary encoding technique used.
(d) Redundancy of simple binary encoding technique used.

5.3 The codewords for four symbols produced by a discrete memoryless source having probability 0.5, 0.25, 0.125, and 0.125 respectively, after applying a particular source coding technique are 0, 1, 10, and 11. Show that the average codeword length is less than the source entropy. The Kraft inequality for this code is specified as 1.5. Will this code have any deciphering problem?

5.4 For a discrete memoryless source, the average codeword length is 2.36 as computed by Shannon–Fano source-coding technique. If the entropy of the specified source is 2.33, determine the coding efficiency and redundancy of the code.

5.5 Consider an alphabet of a discrete memoryless source generating 8 source symbols with probabilities as 0.02, 0.03, 0.05, 0.07, 0.10, 0.10, 0.15, 0.48. Generate a Shannon–Fano source codeword for each symbol and compute the respective length of the codewords.

5.6 For the Huffman tree shown below, evaluate the average codeword length.

Step I	Step I	Step II	Step III	Step IV	Step V
Symbol, s_k	Probability, p_k	Binary Logic	Combined Probability	Binary Logic	Codeword
s_0	0.73			1	1
s_1	0.25	1	0.27	0	01
s_2	0.02	0			00

5.7 Consider an alphabet of a discrete memoryless source having eight source symbols with their respective probabilities as given below:

$[s_k] = [s_0 \quad s_1 \quad s_2 \quad s_3 \quad s_4 \quad s_5 \quad s_6 \quad s_7]$

$[p_k] = [0.48 \quad 0.15 \quad 0.10 \quad 0.10 \quad 0.07 \quad 0.05 \quad 0.03 \quad 0.02]$

Create a Huffman tree and show that the corresponding codes are 1; 001; 010; 011; 0001; 00000; 000000; 000001.

5.8 Consider an alphabet of a discrete memoryless source having eight source symbols with their respective probabilities as given below:

$[s_k] = [s_0 \quad s_1 \quad s_2 \quad s_3 \quad s_4 \quad s_5 \quad s_6 \quad s_7\,]$

$[p_k] = [0.48 \quad 0.15 \quad 0.10 \quad 0.10 \quad 0.07 \quad 0.05 \quad 0.03 \quad 0.02]$

Assume 2 number of symbols in an encoding alphabet, construct a Huffman Tree. ●●●

5.9 For a third-order extension, the binary Huffman source code for a discrete memoryless source generates two symbols with probability 0.6 and 0.4, respectively. How many possible messages are available? Solve the probabilities of all new message symbols. ○●●

5.10 A discrete memoryless source generates six source symbols with their respective probabilities as 0.15, 0.12, 0.30, 0.08, 0.10, and 0.25. Design the 4-ary Huffman tree for each symbol and determine the respective codewords and their lengths. ●●●

5.11 Calculate the maximum number of data bits that will be received with errors if a 2 ms burst noise affect the data transmitted at (a) 1500 bps; (b) 12 kbps; and (c) 96 kbps. ○●●

5.12 As the number of data bits increases, more error control bits are needed to generate encoded bits as per Hamming code. Outline step-by-step procedure to calculate Hamming check bits. Depict the position of data bits and check bits for $k = 10$. ●●●

5.13 For the codeword bit pattern 1 0 0 1 1 0 1, construct a Hamming code to detect and correct for single-bit errors assuming each codeword contains a 7-bit data field. Verify that received bit pattern has no error. ○○●

5.14 The generator polynomial of a (7, 4) cyclic code is $g(x) = 1 + x + x^3$. Find the 16 codewords of this code. ○●●

5.15 Determine the CRC for the data sequence 10110111 for which the generator polynomial is $x^5 + x^4 + x^1 + x^0$ (i.e., 110011). ○○●

5.16 At the receiver, the received data pattern is 1011011101001, for which the generator polynomial is $x^5 + x^4 + x^1 + x^0$ (i.e., 110011). Find the remainder and show that there is no transmission errors. ○○●

5.17 Given a 6-bit data pattern 100100 and a divisor of 1101, answer the following.

(a) What will be the dividend at the sender end?
(b) Compute the quotient and the remainder.
(c) Determine the CRC.
(d) What will be the transmitted data sequence?
(e) What will be the received data sequence (assuming no error)?
(f) What will be the remainder at the receiver end? ●●●

5.18 For a standard BCH code (15, 5) and given generator polynomial coefficients 10100110111, judge the information data length, encoded data length, number of check bits and number of possible errors that can be detected. ●●●

5.19 Construct a Hadamard matrix for $n = 4$. ●●●

5.20 Using the generator polynomials $g_1(x) = 1 + x + x^2$ and $g_2(x) = 1 + x^2$, evolve the convolution code for the given data sequence 101110. ●●●

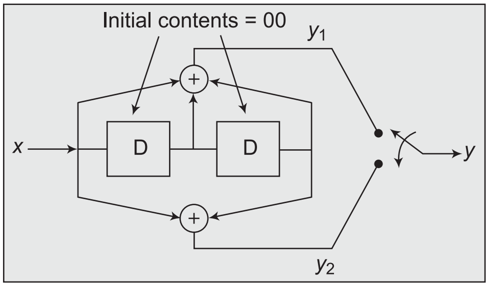
5.21 Represent a convolution encoder ($r = ½$, $K = 3$) with state diagram for the given input data sequence 101011. ○●●

5.22 Model a code tree for a convolution encoder ($r = ½$, $K = 3$) for the given input data sequence 10101. ○○●

5.23 Generate the Trellis diagram for Viterbi decoding of the given data sequence: 10101. ●●●

5.24 For the convolution encoder ($r = ½$, $K = 3$), draw the Trellis diagram and decode the received decode data pattern: 01 11 10 11 00. ○○●

5.25 For the convolution encoder shown below, draw the state and Trellis diagram. Use Viterbi algorithm to decode the received sequence 11 10 01 10 01.



5.26 For (255, 233) Reed–Solomen code, show that 11 symbols in error can be corrected. Find the minimum Hamming distance in this case. ○○●

Critical Thinking Questions

5.1 The entropy represents a fundamental limit on the average number of bits (codeword) per source symbol necessary to represent a discrete memoryless source. Moreover, in source encoding, the average codeword length cannot be made smaller than entropy. In view of these considerations, relate the code efficiency of fixed-length and variable length source codes. **[LO 5.1]**

5.2 "When a combined probability of two lowermost symbols is placed as high as possible, the resulting Huffman source code has a significantly smaller value of variance as compared to when it is placed as low as possible". Using the relationship between variance and the average length of the codeword, show the coherence between two variations in the process of Huffman source encoding for five different source symbols. Assume appropriate data for their respective probabilities and codewords. **[LO 5.2]**

5.3 Cyclic code is a subclass of linear codes in which any cyclic shift of a codeword results in another valid codeword. This feature allows easy implementation of cyclic encoders with linear sequential circuits by employing shift registers and feedback connections. Design a general model of encoding (n, k) algebraic cyclic code and organize the encoding operation. **[LO 5.3]**

5.4 In order to obtain the required resolution, the soft-decision Viterbi decoding algorithm uses a distance metric, known as Euclidian distance. Articulate its usage with the help of an appropriate example data. **LO [5.4]**

5.5 "The Reed–Solomen (RS) code is important because an efficient hard-decision decoding algorithm is available which makes it possible to employ long codes". Generate design parameters of a typical RS code. **[LO 5.5]**

References for Further Reading

[SI12] Singal, TL; *Analog and Digital Communications.* Tata McGraw-Hill, 2012.

[Wel08] Wells Richard B; *Applied Coding and Information Theory for Engineers*. Dorling Kindersley India, 2008.

6 Spread-Spectrum Communications

Learning Objectives

To master spread-spectrum communications, you must cover these milestones:

LO 6.1 Understand the principle of spread-spectrum modulation including pseudo-noise sequence

LO 6.2 Illustrate the concept of frequency-hopping spread-spectrum (FHSS) and direct-sequence spread-spectrum (DSSS) with coherent BPSK

LO 6.3 Implement DSSS in code division multiple access (CDMA) multiuser communication systems

LO 6.4 Outline multiple access techniques including FDMA, TDMA, CDMA, and Multi-Carrier FDMA

Essence of Spread-spectrum Communications

The design of baseband pulse-shaping as well as bandpass digital modulation techniques achieves the prime objective of conserving bandwidth resource and better spectrum efficiency. However, it is accompanied with possible interception and detection by unintended users. Moreover, because of its narrow band, it is more susceptible to signal jamming. Thus, the modulated signals transmitted by narrowband digital communication systems are more susceptible to jamming and can be easily intercepted by unauthorized eavesdropper. In spread-spectrum communications, which was initially developed for military and intelligence purpose, the information-carrying data signal is spreaded over a wider range of frequency spectrum than required. For given transmitted power, a broader spectrum means lower signal power (which makes signal detection and interception difficult), and higher spectral redundancy (which makes the signals more resistant to partial band jamming).

INTRODUCTION

In a conventional wideband FM, the bandwidth required by an FM signal is a function not only of the baseband bandwidth, but also of the amount of modulation deviation. The noise and interference reduction advantage of FM becomes significant only when the frequency deviation from the unmodulated carrier signal frequency is large as compared with the modulating (baseband) signal frequency.

Basis of Spread-Spectrum Communications

Spread-spectrum is a means of transmission where information data sequence occupies much wider bandwidth than necessary to transmit it over a communication channel. Spectrum spreading is achieved through the use of a Pseudo-Noise (PN) code that is independent of the data sequence. The same PN code is used in the receiver, operating in synchronization with the transmitter, to despread the received signal so that the original data sequence can be recovered. Shannon's channel capacity theorem is the basis for spread-spectrum communications. It shows that the capacity of a communication channel to transfer error-free information is enhanced with increased bandwidth, even though the signal-to-noise ratio (S/N) is decreased because of the increased bandwidth.

Criteria for Spread-spectrum Communications

The first criterion for spread-spectrum communications is that the bandwidth of the transmitted signal must be much greater than the information bandwidth. And the second one is that the transmitted bandwidth must be determined by some function that is independent of the information and is known to the receiver only.

Since wideband FM produces a spectrum much wider than that required for the baseband information, it might be considered a spread-spectrum technique. In wideband FM, the information signal itself is used to broadband the signal transmitted, whereas the spread-spectrum technique uses PN code independent of the information signal. However, spread-spectrum techniques have a processing gain, analogous to that of wideband FM, permitting reduced carrier-to-noise power ratio, or low transmitted power levels, to achieve satisfactory performance.

Spread-Spectrum Signals are Transparent to Interfering Signals

An important attribute of spread-spectrum modulated signal is that the transmitted signal appears like a noise. This is how the spread-spectrum signal is enabled to propagate through the communication channel, and remain undetected by anyone listening in the same channel except the desired listener.

So, we can say that the primary advantage of spread-spectrum communications is its ability to reject either intentional interference (jamming from hostile environment), or unintentional interference by another user trying to transmit through the same channel.

In fact, Spread-Spectrum (SS) is a technique whereby an already modulated signal (baseband modulation) is modulated a second time (bandpass modulation) in such a way so as to produce a signal which interferes in a barely noticeable way with any other signal operating in the same frequency band. Thus, a particular AM or FM broadcast receiver would probably not notice the presence of a spread-spectrum signal operating over the same frequency band. Similarly, the SS receiver would not notice the presence of the AM or FM signal. Thus, it can be said that the spread-spectrum signals are transparent to the interfering signals and vice versa.

IMPORTANT!

The larger the ratio of the bandwidth of the spreaded signal to that of the information signal, the smaller the effect of undesired signal interference. Spread-spectrum communication systems are useful for suppressing interference, making secure communication.

In this chapter...

- We begin with general principle of spread-spectrum communications and define processing gain which represents an approximate measure of its interference rejection capability. Then, we describe various types of PN sequences employed in generation of spread-spectrum signal. **LO 6.1**
- Next, we illustrate the concept of two prominent types of spread-spectrum techniques—frequency hopping and direct sequence. **LO 6.2**
- Then, we discuss the implementation of Direct Sequence Spread-Spectrum (DSSS) technique in code division multiple-access multiuser detection communication systems. **LO 6.3**
- Finally, we briefly describe multiple-access techniques such as FDMA, TDMA, CDMA, and orthogonal FDMA used in cellular mobile communication systems. **LO 6.4**

6.1 PRINCIPLES OF SPREAD-SPECTRUM MODULATION

LO 6.1

A Spread-Spectrum Signal

A spread-spectrum signal is one that expands the information signal bandwidth much beyond what is required by the underlying coded-data modulation. Spread-spectrum signals are difficult to detect on narrow band RF receivers. Spread-spectrum signals are less likely to interfere with other narrowband radio communications due to spread of energy over a wide band. Moreover, they are more secure than narrowband radio communications.

We reiterate that spread-spectrum is a transmission technique in which the transmitted signal occupies a larger bandwidth than required. The transmitted bandwidth is many times larger than the bandwidth required transmitting baseband signal.

Figure 6.1.1 illustrates the functional block diagram of a general model of Spread-Spectrum (SS) communication system.

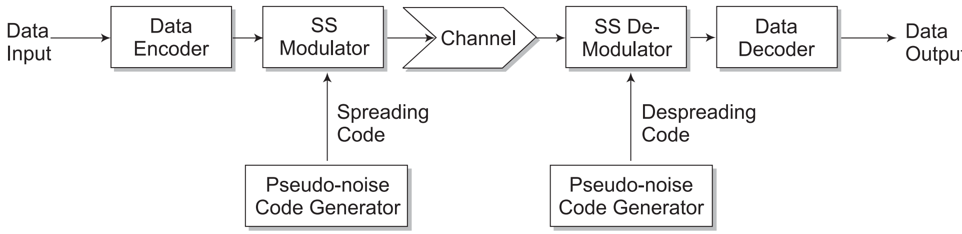


Figure 6.1.1 Spread-Spectrum Communication System

Operation

The input data is fed into a data encoder. The encoded signal is modulated using the spreading code generated by a pseudo-noise code generator. On the receiving end, the same despreading code is used to demodulate the spread-spectrum signal. Finally, the demodulated signal is processed by the data decoder to recover the data output.

Figure 6.1.2 illustrate the basic concept of spread-spectrum process.

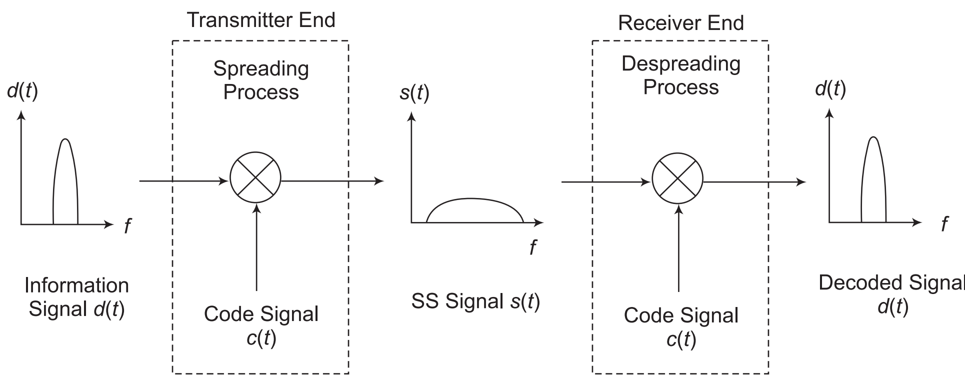


Figure 6.1.2 Basic Concept of Spread-Spectrum Process

Interpretation

The narrowband information signal is converted to wideband spread-spectrum signal as the spreading process at transmitter. The received spread-spectrum signal is despread into original narrowband information signal as the despreading process at receiver. The code signal used for spreading and dispreading process is the same. Thus, the effect of spread-spectrum modulation is to increase the bandwidth (spread the spectrum) of the information signal significantly for transmission.

Processing Gain

The *processing gain* is defined as the gain in signal-to-noise ratio (S/N) obtained by the use of spread-spectrum modulation technique. Processing gain represents the gain achieved by processing a spread-spectrum signal over an unspreaded signal. It is an approximate measure of the interference rejection capability.

Figure 6.1.3 (a) shows the received spectra of the desired spread-spectrum signal and the interference at the output of the receiver wideband filter. Multiplication by the spreading code produces the spectra of Figure 6.1.3 (b) at the demodulator input.

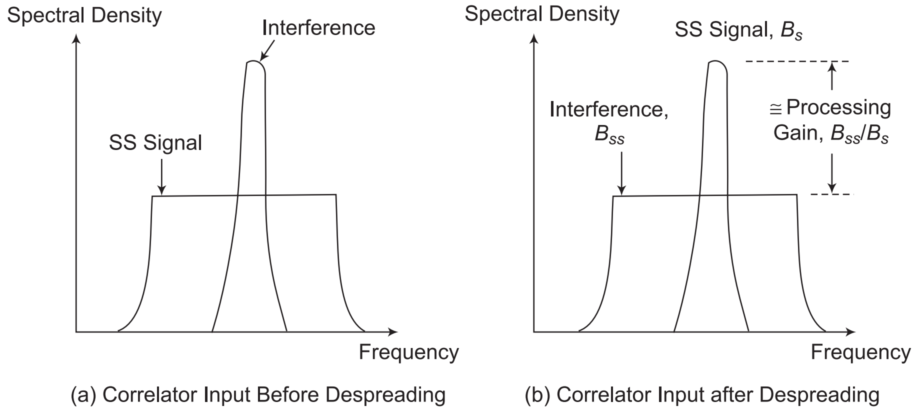


Figure 6.1.3 Spread-Spectrum Approach in Frequency-domain

Mathematical Expression

The signal bandwidth is reduced to B_s, while the interference energy is spreaded over an RF bandwidth exceeding B_{ss}. The filtering action of the demodulator removes most of the interference spectrum that does not overlap with the signal spectrum. Thus, most of the original interference energy is eliminated by spreading and minimally affects the desired receiver signal.

As stated previously, the *processing gain* is expressed as

$$\boxed{G_p = \frac{B_{ss}}{B_s}}$$

where B_{ss} is the spreaded bandwidth and B_s is the bandwidth of information signal.[1]

Figure 6.1.4 depicts a general approach of spread-spectrum represented in time domain in which a data bit is encoded by 15-chip PN code.

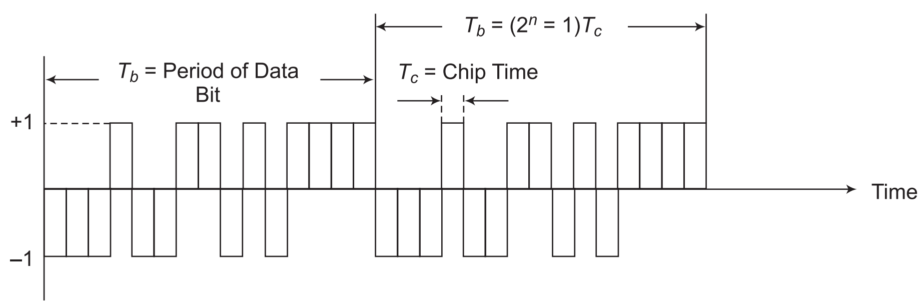


Figure 6.1.4 Spread-Spectrum Approach in Time Domain

[1] Processing gain is the ratio of the spreaded RF bandwidth to the original information bandwidth. The effect of multipath fading as well as interference can be reduced by a factor equivalent to the processing gain. In fact, it quantifies the degree of interference rejection.

Mathematical Expression

Specifically, the *processing gain* of the spread-spectrum signal can also be expressed in time domain as

$$\boxed{G_p = \frac{T_b}{T_c}}$$

where T_b is the bit duration of the information data, and T_c is the chip duration of the PN sequence used in spread-spectrum signal.

IMPORTANT!

Greater will be the processing gain provided the chip duration is made smaller or the length of the PN sequence is made longer.

Another Way of Expressing G_p

Using information data rate, $R_b = 1/T_b$ and PN code rate, $R_c = 1/T_c$, processing gain of spread-spectrum signal can also be expressed as

$$\boxed{G_p = \frac{R_c}{R_b}}$$

LET'S RECONFIRM OUR UNDERSTANDING!!

- Is pseudo-random noise (PN) sequence same at spread-spectrum modulator and demodulator?
- Specify significance of processing gain in spread-spectrum signal.

SOLVED EXAMPLE 6.1.1 **Processing Gain of Spread-Spectrum**

A spread-spectrum communication system has information bit duration of 32.767 ms, and chip duration of 1 μs. Calculate the processing gain in dB.

Solution Specifically, the processing gain can be written as

$$G_p = \frac{T_b}{T_c}$$

where T_b is the bit duration of the information data, and T_c is the chip duration of the PN sequence used in spread-spectrum signal.

For given T_b = 32.767 ms and T_c = 1 μs, we get

$$G_p = \frac{32.767\times10^{-3}}{1\times10^{-6}} = 32767;\text{ or } 10\log(32767) = 45.15\text{ dB} \qquad \textbf{Ans.}$$

SOLVED EXAMPLE 6.1.2 **Processing Gain of Spread-Spectrum**

A direct sequence spread-spectrum communication system has a binary information data rate of 10 kbps, and a PN code rate of 192 Mcps. If BPSK modulation is used, compute the processing gain.

Solution The processing gain, G_p is given as $G_p = \dfrac{R_c}{R_b}$

For given R_c = 192 Mcps and R_b = 10 kbps, we get

$$G_p = \frac{192 \text{ Mcps}}{10 \text{ kbps}} = 19200$$

Expressing it in dB, we have $G_p = 10 \times \log_{10}(19200) = 42.83$ dB **Ans.**

SOLVED EXAMPLE 6.1.3 — Spread-Spectrum Signal Bandwidth

A speech signal is band-limited to 3.3 kHz, and uses 128 quantization levels to convert it into digitized analog information data. This is required to be transmitted by a pseudorandom noise spread-spectrum communication system. Determine the spread-spectrum signal bandwidth in order to obtain a processing gain of 21 dB.

Solution We know that the processing gain (G_p) quantifies the degree of interference rejection in a pseudorandom noise spread-spectrum communication system. It is simply the ratio of transmitted RF bandwidth to the information signal bandwidth, and is represented as

$$G_p = \frac{B_{ss}}{B_s}$$

For given G_p = 21 dB or 126, and B_s = 3.3 kHz, we get

$$B_{ss} = 126 \times 3.3 \text{ kHz} = 416 \text{ kHz}$$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 6.1.4 A spread-spectrum communication system has information bit duration of 32.767 ms, and chip duration of 1 μs. Calculate the processing gain in dB.

Ex 6.1.5 An analog information signal is bandlimited to 3 kHz, and uses 128 quantization levels. What is the minimum sampling rate? What is the minimum chip rate required to obtain a processing gain of 20 dB.

Pseudo-Noise (PN) Sequence

Spreading of narrow bandwidth of baseband signal is accomplished through the use of a spreading code, called pseudo-noise (PN) code. A PN code is independent of the baseband signal. The same code is used to demodulate the received data at the receiving end.

Pseudo-noise sequence or pseudorandom numbers is a random periodic sequence of binary ones and zeros. A PN generator will produce a periodic sequence that eventually repeats but that appears to be random. PN sequences are generated by an algorithm using some initial value called *seed*. The algorithm produces sequences of numbers that are not statistically random but it will pass many reasonable tests of randomness. Unless the seed and the algorithm is known, it is impractical to predict the PN sequence.

Properties of PN Sequences

There are two important properties of PN sequences—randomness and unpredictability. There are three criteria used to validate that a sequence of numbers is random. These are the following:

- *Uniform Distribution*—the frequency of occurrence of each of the numbers should be approximately same
- *Independence*—no one value in the sequence can be inferred from others
- *Correlation property*—it states that if a period of the sequence is compared term by term with any cycle shift of itself, the number of terms that are same differs from those that are different by at most 1.

Generation of a PN Sequence

A PN sequence is a noiselike spreading code which consists of a periodic binary data sequence that is usually generated by means of a feedback register. The feedback register consists of an ordinary shift register made up of m flip-flops and a logic circuit.

Figure 6.1.5 depicts a simple arrangement to generate PN sequences.

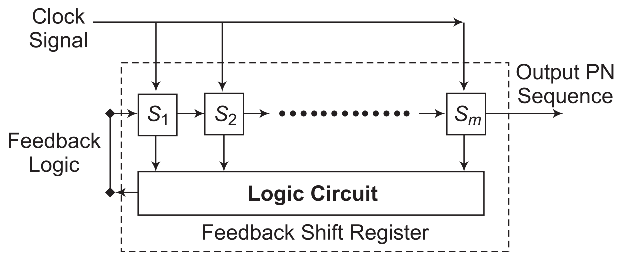


Figure 6.1.5 Block Diagram of PN Sequence Generator

Interpretation

A clock signal is applied to all flip-flops $S_1, S_2, ..., S_m$ which are used to change the state of each flip-flop simultaneously. Also, the logic circuit computes a specified function of the state of all the flip-flops. Its output is then fed back as the input to the S_1 flip-flop. A feedback shift register is said to be linear when the logic circuit consists of modulo-2 adders only. For m flip-flops, the output of logic circuit uniquely determines the subsequent sequence of states of the flip-flops. Therefore, the m^{th} flip-flop in the shift register gives the required PN sequence.

***m*-Sequence**

The PN sequence is called the *maximum-length sequence* (m-sequence), if the period is exactly equal to N. The number of possible states of the shift register is at the most 2^m only, where m is total number of flip-flops used in the shift register.

The PN sequence eventually become periodic with a period of 2^m. Since the linear feedback shift register does not possess the zero state (all 0s), the maximum-length-sequence (m-sequence) is $2^m - 1$. As the length of the shift-register and the repetition period is increased, the resultant PN sequence becomes similar to the random binary sequence.

- The presence of binary symbol 0 or 1 is equally probable.
- It has balance property, run property, and correlation property.
- A good PN sequence is characterized by an autocorrelation that is similar to that of a white noise.
- The cross-correlation among different pairs of PN sequences should be small to reduce mutual interferences.

IMPORTANT!

A PN code is periodic, and m-sequences are the most widely known as binary PN sequences. A class of PN sequence, called Walsh codes, is an important set of orthogonal PN codes.

SOLVED EXAMPLE 6.1.6 **PN Sequence Period**

A pseudo-random (PN) sequence is generated using a feedback-shift register with four number of memory elements. The chip rate ($1/T_c$) is 10^7 chips per second. Determine the PN sequence period and the chip duration of the PN sequence generated.

Solution For given chip rate of 10^7 chips per second, the PN sequence period = 1/chip rate

That is, the PN sequence period, $T_c = 1/10^7$ sec $= 10^{-7}$ sec **Ans.**

For given $m = 4$, the PN sequence length $N = 2^m - 1 = 2^4 - 1 = 15$

Chip duration of the PN sequence generated = $N \times T_c$

Hence, Chip duration of the PN sequence generated 15×10^{-7} sec = 1.5 μs **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 6.1.7 AIf the PN generator is driven by a clock signal at the rate of 10 MHz, and the feedback register has 37 stages, find the total length of sequence in hours.

Gold Sequence

A *Gold sequence*, or sometimes known as Gold code, is constructed by the XOR of two *m*-sequences using the same clock signal. Ordinary PN sequences do not satisfy the requirement of zero cross-correlation function between the codes for all cyclic shifts. Gold sequence provides a viable alternative to PN sequences. Gold sequences is generated by a pair of PN sequences with feedback taps that involves the modulo-2 addition of two PN sequences. Two *m*-sequences are generated in two different chain of shift registers (SRs), and then these two sequences are bit-by-bit XORed.

Figure 6.1.6 shows an arrangement to generate Gold sequences.

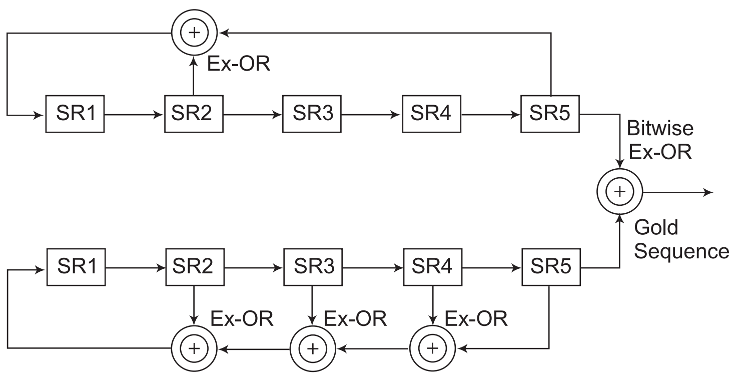


Figure 6.1.6 Gold Sequence Generation

An Example of Gold Sequence

In general, the length of the resulting Gold sequence is not maximal. The desired Gold sequence can only be generated by preferred pairs of *m*-sequences. These preferred pairs can be selected from tables of pairs or generated by an algorithm.

For example,

m-sequence 1:	1111100011011101010000100101100
m-sequence 2:	1111100100110000101101010001110
Gold sequence:	0000000111101101111101110100010

Features of Gold Code

The period of any code in a Gold set generated with two *n*-bit shift registers is $N = 2^n - 1$, which is same as the period of the *m*-sequences. There are a total of $(N + 2)$ codes in any family of Gold codes. For example, there are total 33 unique sequences in Gold code. The Gold sequences are two initial m-sequences plus 31 generated sequences for shift in the initial condition from 0 to 30 in a preferred pair of 5-bit shift registers.

Kasami Sequence

In *Kasami sequence*, there are two sorts of sequences, small sets and large sets. For even value of *n*, a small set of Kasami sequences can be generated which contains $M = 2^{n/2}$ distinct sequences each with period $N = 2^n - 1$. The large set contains both Gold sequences and a small set of Kasami sequences as subsets.

Application

m-sequences are easy to generate and are very useful for Frequency Hopped Spread-Spectrum (FHSS) and non-CDMA direct sequence spread-spectrum systems (DSSS). However, for CDMA DSSS system, m-sequences are not optimal. Kasami sequences are likely to be used in future generation wireless communication systems.

Self-Assessment Exercise linked to LO 6.1

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/160

Q6.1.1 What do you understand by spread-spectrum modulation? Mention the primary advantage of spread-spectrum communication. ○○●

Q6.1.2 Give an example of the benefit of using a modulation carrier bandwidth significantly wider than the baseband bandwidth. ○●●

Q6.1.3 Compare and contrast between wideband FM and spread-spectrum signal. ○●●

Q6.1.4 Calculate the processing gain if the information bit rate is 1500 bps and the bandwidth of spread-spectrum signal is 21.5 MHz. ○●●

Q6.1.5 What is meant by chips of the PN sequence? List different techniques of PN sequences. ○○●

Q6.1.6 Describe briefly Balance property and Run property in uniform distribution as criterion to validate a PN sequence. ○●●

Q6.1.7 Design the Gold sequence for two *m*-sequences given as

m-sequence 1: 1001100111010101010010100101100

m-sequence 2: 1100100100110011101101010101010 ●●●

Note
○○● Level 1 and Level 2 Category
○●● Level 3 and Level 4 Category
●●● Level 5 and Level 6 Category

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 6.1: *Understand the principle of spread-spectrum modulation including pseudo-noise sequence.*

LO 6.2

Recall

6.2 SPREAD-SPECTRUM TECHNIQUES

Spread-spectrum communication is a means of transmitting information data by occupying much wider bandwidth than necessary. Spreading of spectrum is achieved through the use of a pseudo-random noise (PN) code sequence at the transmitter (for spreading) as well as at the receiver (for despreading), operating in synchronization. The PN code is independent of the information data sequence.

An important attribute of spread-spectrum communication is that the transmitted signal assumes a noiselike appearance. Spread-spectrum techniques have a processing gain which allows reduced signal-to-noise ratio (S/N), or low transmitted power levels, to deliver satisfactory performance in a hostile operating environment. Spread-spectrum communication systems typically use an RF bandwidth which is 100 to 1000 times or even more than that required for the information rate.

What We Discuss Here ....

Frequency-hop *M*-ary frequency-shift keying, known as Frequency-Hopped Spread-Spectrum (FHSS) and direct-sequence bi- (or quad-) phase shift keying, known as Direct-Sequence Spread-Spectrum (DSSS) are the two most popular spread-spectrum techniques. Both of them rely on generation of PN sequences.

- In FH/MFSK system, the PN sequence makes the carrier signal hop over a number of frequencies in a pseudo-random fashion, the spectrum being spreaded in a sequential manner.
- For DS/QPSK system, PN sequence makes the transmitted signal assume a noiselike appearance, and data are transmitted using a pair of carriers in quadrature.

The concept of frequency hopping spread-spectrum (FHSS) and direct sequence spread-spectrum (DSSS) with coherent BPSK are illustrated in this section.

6.2.1 Frequency Hopping Spread-Spectrum (FHSS)

Frequency Hopping

Frequency hopping involves a periodic change of transmission frequency over a wide band. The set of possible time-varying, pseudorandom carrier frequencies is called the *hopset*. The rate of hopping from one frequency to another is a function of the information data rate. The specific order in which frequencies are hopped is a function of PN code sequence.

Hopping occurs over a frequency band that includes a number of channels. Each channel is defined as a *spectral region* with a central frequency in the hopset. It has a large bandwidth to include most of the power in a narrowband modulation having the corresponding carrier frequency.

FHSS Technique

In *frequency hopping spread-spectrum* (FHSS) technique, the spectrum of the transmitted signal is spread sequentially (pseudo-random-ordered sequence of the frequency hops) rather than instantaneously. The carrier signal hops randomly from one frequency to another. In FHSS system, the information data signal is transmitted over a large random sequence of radio frequencies, hopping from one frequency to another frequency at pre-determined fixed interval of time.

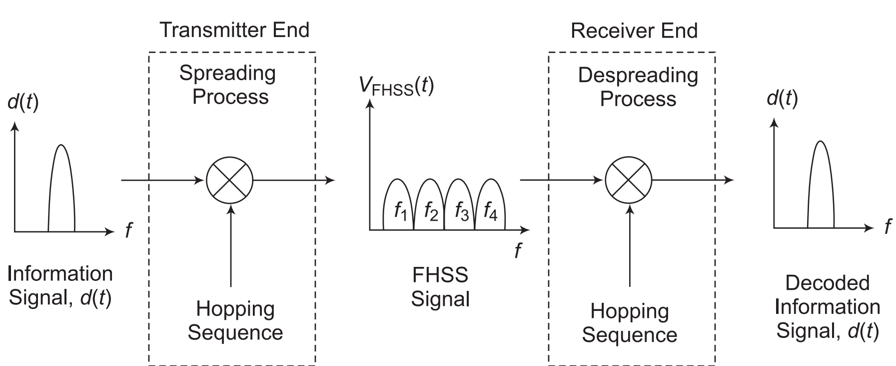


Figure 6.2.1 Basic Concept of FHSS Technique in Frequency Domain

Frequency Hopping and Processing Gain

Recall that in spread-spectrum signals, the ratio B_{ss}/B_s is known as the *processing gain* G_p.

In frequency-hopping systems,

- B_{ss} represents the bandwidth of the spectrum over which the hopping occurs and is known as the *total hopping bandwidth.*
- B_s denotes the bandwidth of a channel used in the hopset and is known as the *instantaneous bandwidth*.

Information data is sent by hopping the transmitter carrier frequencies to seemingly random channels, which are known only to the desired receiver. On each channel, small bursts of data are sent using conventional narrowband modulation before the transmitter hops again. In fact, the bandwidth of FHSS signal is simply B_s times the number of frequency channels available, where B_s is the bandwidth of each hop channel.

Basic Process of Frequency Hopping

A number of different frequency channels are allocated for the FHSS signal. The spacing between frequencies usually corresponds to the bandwidth of the information signal. Figure 6.2.2 shows the basic concept of a frequency-hopping process.

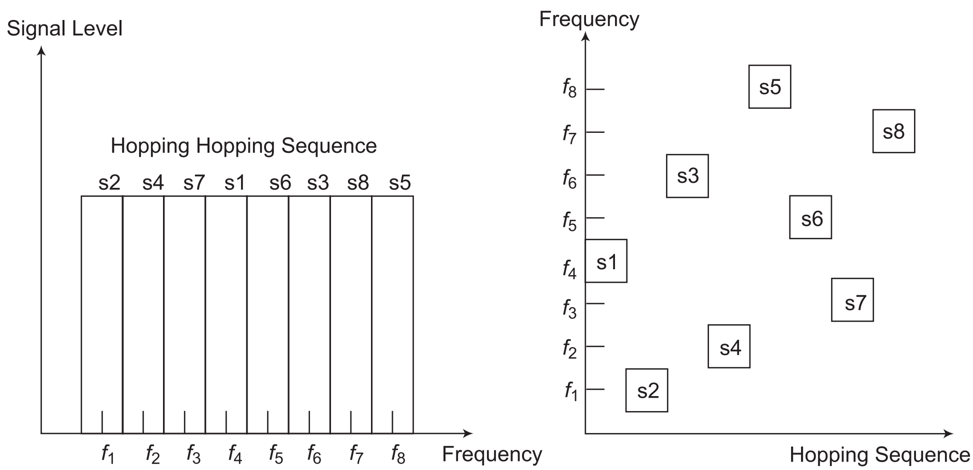


Figure 6.2.2 Basic Process of Frequency Hopping

Processing Gain in FHSS

Let there are 2^k number of different carrier frequencies (that is, distinct channels) where k is the number of codes in PN code generator. The FHSS transmitter operates in one of these channels at a time for a fixed interval of time. During this time, some bits of information data or a fraction of a bit are transmitted. The sequence of frequency channels is determined by the PN code. Both transmitter and receiver use the same PN code in perfect synchronization with a pre-determined sequence of channels.

In that case, the processing gain is given by $G_p = 2^k$.

Representation of FHSS in Time Domain

If only a single carrier frequency is used on each hop, digital data modulation is called single-channel modulation. The time duration between hops is called the *hop duration* or the *hopping period* and is denoted by T_h. The signal frequency remains constant for specified time duration, referred to as a chip time, T_c.

Figure 6.2.3 depicts a general FHSS approach with frequency hopping versus time graph.

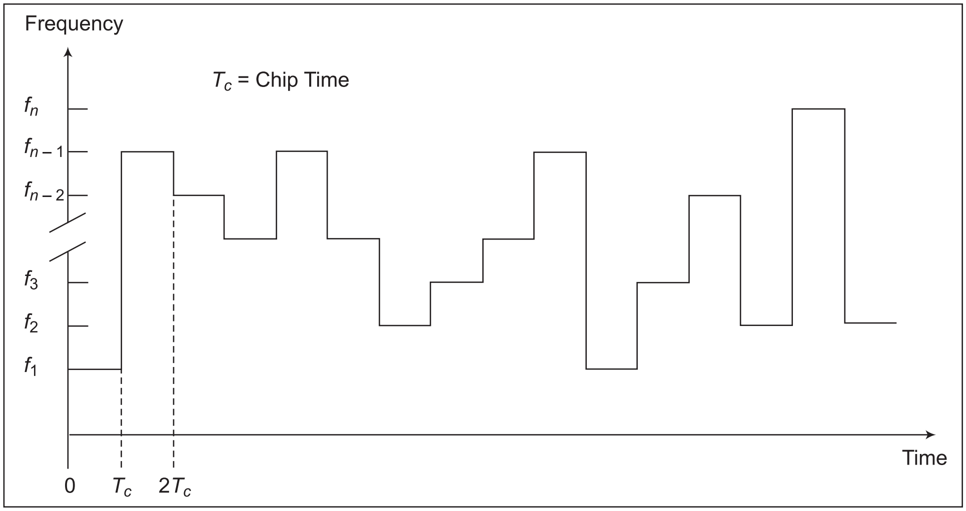


Figure 6.2.3 Basic Concept of FHSS Technique in Time Domain

Fast Frequency Hopping

Fast frequency hopping occurs if there is more than one frequency hop during each transmitted symbol. The frequency hops occur much more rapidly and the hopping rate equals or exceeds the information symbol rate. There are multiple hops per information data bit and the same bit is transferred using several frequencies. In each hop, a very short information data packet is transmitted. Fast frequency hopping systems are used in military communication applications.

Slow Frequency Hopping

Slow frequency hopping occurs if one or more symbols are transmitted in the time interval between frequency hops. There are multiple data bits per hop. In a slow frequency hopping system, long data packets are transmitted in each hop over the wireless channel.[2]

SOLVED EXAMPLE 6.2.1 Number of PN bits in FHSS System

An FHSS system employs a total bandwidth of 400 MHz and an individual channel bandwidth of 100 Hz. What is the minimum number of PN bits required for each frequency hop?

Solution Total bandwidth, B_t = 400 MHz (Given)

Channel bandwidth, B_c = 100 Hz (Given)

Processing gain, $$G_p = \frac{B_t}{B_c} = \frac{400 \text{ MHz}}{100 \text{ Hz}} = 4 \times 10^6$$

In an FHSS system, processing gain is also given by $G_p = 2^k$, where k is number of PN bits

Therefore, $4 \times 10^6 = 2^k; \quad \Rightarrow 2^{22} = 2^k$

Hence, minimum number of PN bits, k = 22 bits **Ans.**

[2] If the FHSS signal is jammed on one frequency then few bits transmitted at that particular frequency are corrupted, while all other bits are received satisfactorily at other hopping frequencies. The jammer must jam all 2^k frequencies in a frequency-hopping system which is not easy because the value of $\boldsymbol{k}$ is usually very large.

SOLVED EXAMPLE 6.2.2 **Number of Frequencies in FHSS**

Calculate the minimum number of frequencies required for a frequency-hopping spread-spectrum communication system if the frequency multiplication factor is 7.

Solution Let there are $N = 2^k$ number of different frequencies which form 2^k number of distinct channels, where k is the frequency multiplication factor or the number of codes in PN code generator in an FHSS communication system. That is,

For given $k = 7$, $N = 2^7 = 128$ **Ans.**

SOLVED EXAMPLE 6.2.3 **Fast-Frequency Hopping**

A frequency-hopping spread-spectrum communication system utilizes fast-hop technique at the hop rate of 10 hops per information bit. If the information bit rate is 2800 bps, what is the frequency separation?

Solution In a frequency-hopping spread-spectrum communication system,

Frequency separation = hop rate × bit rate

For given hop rate = 10 hops per information bit and information bit rate = 2800 bps,

Frequency separation = 10 × 2800 = 28000 Hz or 28 kHz **Ans.**

SOLVED EXAMPLE 6.2.4 **Concept of Slow-Frequency Hopping**

In a slow frequency-hopping system, long data packets are transmitted over the wireless channel. At each hop, the sequence of frequencies programmed in PN code generator is f_3, f_5, f_6, f_1, f_4, f_8, f_2, and f_7 before returning to the first frequency, f_3. Draw a suitable diagram to illustrate the concept of frequency hopping.

Solution Each data packet is transmitted using a different frequency. Figure 6.2.4 shows the given hopping pattern and associated frequencies for a frequency-hopping system.

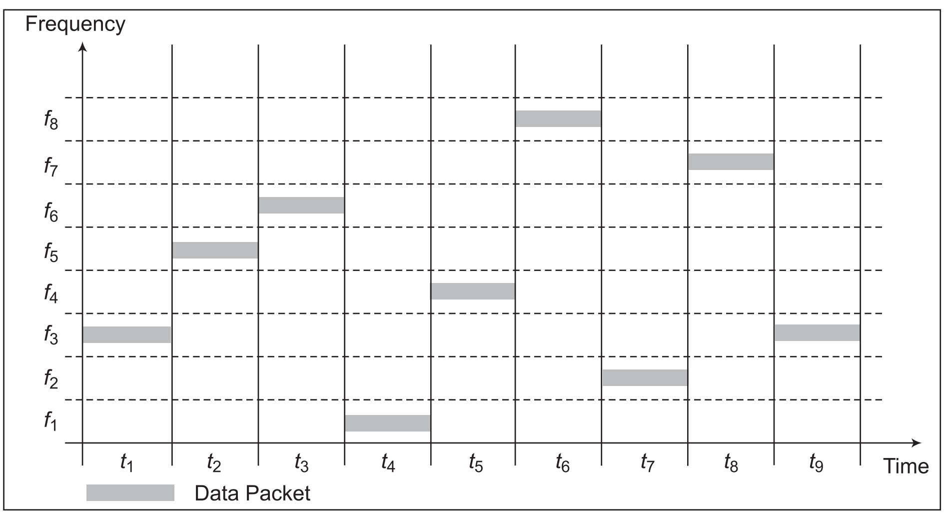


Figure 6.2.4 Illustration of a Slow FHSS Technique

SOLVED EXAMPLE 6.2.5 **Maximum Frequency Hopping in GSM**

In Europe, GSM uses the frequency band 890 to 915 MHz for uplink transmission, and the frequency band 935 to 960 MHz for downlink transmission. Determine the maximum frequency hop from one frame to the next for uplink transmission and downlink transmission. Express it as a percentage of the mean carrier frequency.

Solution Frequency band for uplink transmission = 890 MHz – 915 MHz (Given)

RF bandwidth for uplink transmission = 915 – 890 = 25 MHz

Frequency band for downlink transmission = 935 MHz – 960 MHz (Given)

RF bandwidth for downlink transmission = 960 – 930 = 25 MHz

Therefore, in either case, the maximum frequency hop or change from one frame to the next could be **25 MHz**. **Ans.**

For uplink transmission,
Mean carrier frequency = 890 + (915 – 890)/2 = 902.5 MHz
Maximum frequency hopping = 25/902.5
Hence, maximum frequency hopping = 0.0277 or **2.77%** **Ans.**

For downlink transmission,
Mean carrier frequency = 935 × (960 – 935)/2 = 947.5 MHz
Maximum frequency hopping = 25/947.5
Hence, maximum frequency hopping = 0.0264 or **2.64%** **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 6.2.6 Calculate the minimum number of frequencies required for a frequency-hopping spread-spectrum communication system if the frequency multiplication factor is 8.

Ex 6.2.7 A frequency-hopping spread-spectrum communication system utilizes fast-hop technique at the hop rate of 10 hops per information bit. If the information bit rate is 2500 bps, what is the frequency separation?

6.2.2 FHSS with BFSK or *M*-ary FSK

FHSS Modulation

Most FHSS signals adopt binary or *M*-ary FSK modulation schemes. FSK modulation has the ability to utilize the less complex non-coherent detection (without the need for carrier phase coherence). The use of PN hopping sequence would require the receiver to maintain phase coherence with the transmitter at every frequency used in the hopping pattern. So non-coherent BFSK or *M*-ary FSK are preferred in FHSS systems. Although PAM, QAM, or PSK are more efficient modulation techniques, but they require coherent detection.

Figure 6.2.5 depicts a typical functional block schematic of a *FHSS modulator*.

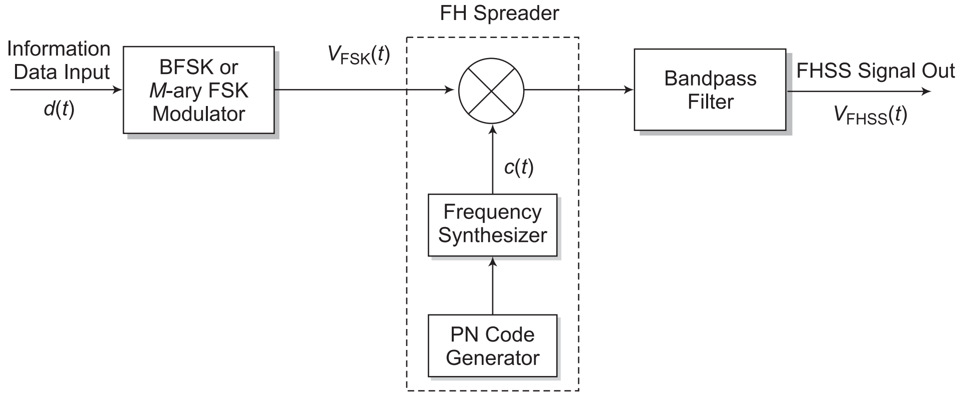


Figure 6.2.5 Block Diagram of FHSS Modulator

Description of FHSS Modulator

The information data $d(t)$ is modulated with a BFSK or M-ary FSK modulator. The carrier frequency of frequency synthesizer $c(t)$ is changed rapidly in accordance with a random orthogonal known hopping pattern generated by a PN code generator. A PN code generator serves as an index to frequency synthesizer into a set of channel frequencies, referred as the spreading code. Each k bits of the PN code generator specifies one of the 2^k carrier frequencies. At each successive k PN bits, a new carrier frequency is selected. The output of FH spreader is passed through a bandpass filter. The resulting FHSS signal $V_{FHSS}(t)$ is finally transmitted.

Figure 6.2.6 depicts a typical functional block schematic of a *FHSS demodulator*.

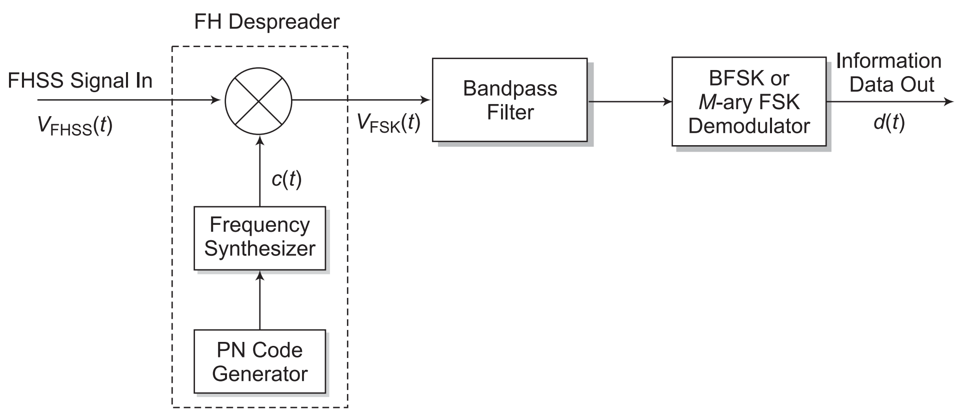


Figure 6.2.6 Block Diagram of FHSS Demodulator

Description of FHSS Demodulator

On reception, the FH spread-spectrum signal is despreaded using the same sequence of PN-coded frequencies by frequency synthesizer. This signal is then demodulated by BFSK or M-ary FSK demodulator to produce the information data output.

6.2.3 Performance of FHSS System

Signal-to-Interference Ratio (SIR)

In an FHSS system, the interference corrupts only a fraction of the transmitted information, and transmission in the rest of the hopped frequencies remains unaffected. FHSS systems provide better security against intentional/unintentional interceptors or potential jammers due to lack of full knowledge of the frequency hopping pattern. To improve the frequency efficiency of FHSS systems, multiple users may be allowed over the same frequency band with little degradation in performance.

The *Signal-to-Interference Ratio* (SIR) per user in FHSS system is given by

$$\text{SIR} = \left(\frac{B}{M-1}\right) \times \left(\frac{\log_2 L}{L \times R_b}\right)$$

where B is the system bandwidth in Hz, M is the number of users in FHSS system, L is the number of orthogonal codes used, and R_b is the signaling rate in bps.

Advantages of FHSS System

- ***High tolerance of narrowband interference.*** Since the FH signal varies its center frequency as per the hopping pattern, a narrowband interference signal will cause degradation only when it aligns with the carrier frequency in every hop.
- ***Relative interference avoidance.*** If it is already known that a particular band of the radio spectrum contains interference, the hopping frequencies can be selected to avoid this band.
- ***Large frequency-hopped bandwidths.*** Current technology permits frequency-hopped bandwidths much greater than that can be achieved with DSSS systems.
- ***More interference tolerance.*** With FHSS, a single interfering frequency will cause degradation only at one hop frequency, regardless of its signal strength relative to the desired signal. The interference tolerance of FHSS thus is more than that of DSSS.

Disadvantages of FHSS System

- ***Non-coherent detection.*** The implementation of frequency synthesizer often limit the continuity of the signal across frequency hops. Therefore, M-ary FSK modulation techniques with non-coherent detection are employed.
- ***Probability of detection.*** The higher probability of detection depends upon the spread bandwidth of the frequency-hopped signal. Since a single hop period is similar to a narrowband carrier, FHSS signal is simpler to detect than a DSSS signal.

IMPORTANT!

If the hopping rate is relatively fast, it is more difficult to detect the transmitted data without a prior knowledge of the hopping pattern in FHSS.

Application

FHSS is used in the design of wireless networks to provide a reliable transmission in the presence of interfering signals. FHSS finds extensive practical applications in Wireless Local Area Network (WLAN) standard for IEEE 802.11 technology, Wireless Personal Area Network (WPAN) standard for IEEE 802.15 technology, and second generation digital cellular networks such as GSM and CDMA.

6.2.4 Direct Sequence Spread-Spectrum (DSSS)

DSSS Technique

In *direct sequence spread-spectrum* (DSSS) technique, a high-speed PN code sequence is employed along with the slow-speed information data being sent. The information digital signal is multiplied by a pseudorandom sequence whose bandwidth is much greater than that of the information signal itself. It results into considerable increase in the bandwidth of the transmission. The power spectral density is reduced. The resulting spreading signal has a noiselike spectrum to all except the intended spread-spectrum receiver. The received signal is despreaded by correlating it with a local pseudorandom signal which is identical to the PN signal used to spread the data at the transmitting end. The most practical system applications employing direct sequence spread-spectrum techniques use digital modulation formats such as BPSK and QPSK.

IMPORTANT! The most practical system applications employing direct sequence spread-spectrum techniques use digital modulation formats such as BPSK and QPSK.

DSSS Modulator with BPSK

Figure 6.2.7 shows a functional block schematic of a *DSSS modulator* system with binary phase shift keying (BPSK) carrier modulation technique.

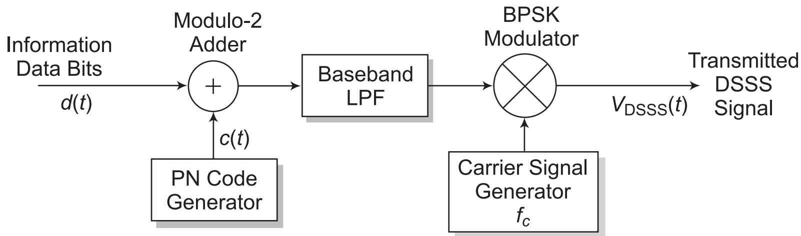


Figure 6.2.7 DSSS Modulator with BPSK Carrier Modulation

Functional Description

A pseudo-noise (PN) sequence is produced by a PN code generator. The baseband information data is spreaded by directly multiplying it with PN sequence. Each bit in the information data is represented by multiple bits of the spreading code. A single pulse of the PN sequence is called a *chip* because it has extremely small time duration.

The spreading code spreads the information signal across a wider frequency band in direct proportion to the number of chips used. A 10-chip spreading code spreads the information signal across a frequency band that is 10 times greater than a 1-chip spreading code. Generally, the information data stream is combined with the spreading code bit stream using an Exclusive-OR (XOR) logical operation.

The information bits or binary coded symbols are added in modulo-2 summer to the chips generated by PN code generator before being phase modulated. The PN code has much higher data rate than the information data rate. The DSSS signal is BPSK modulated using high-frequency carrier signal so as to enable transmission over the communication channel.

DSSS Demodulator with Coherent BPSK Detection

A *coherent BPSK detection* technique is used in the DSSS modulator at the receiver. Figure 6.2.8 shows a functional block schematic of a DSSS demodulator system with coherent BPSK detection technique.

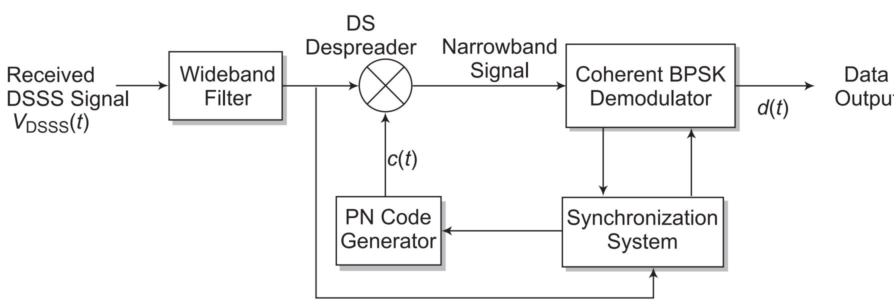


Figure 6.2.8 DSSS Demodulator with Coherent BPSK Detection

Functional Description

The received DSSS signal is first passed through wideband filter. A DSSS with BPSK demodulator uses a locally generated identical PN code generator and a DS despreader (also called correlator).

A *DSSS correlator* is a special matched filter that responds only to signals that are encoded with a PN code that matches its own PN code. It enables to separate the desired coded information from all possible signals. The received DSSS signal is first processed through a DS despreader that despreads the signal. The DS despreaded signal is then demodulated with BPSK demodulator. For proper operation, synchronization system is used so that the locally generated PN sequence be synchronized to the PN sequence used to spread the transmitted signal at the DSSS transmitter. The two PN codes are aligned to within a fraction of the chip in quick time during acquisition phase, followed by tracking using PLL technique. The decoded output data is same as the original information signal.[3]

6.2.5 Comparison of FHSS and DSSS

Table 6.2.1 presents a brief comparison of merits and demerits of two major spread-spectrum techniques—FHSS and DSSS.

Table 6.2.1 Comparison of FHSS and DSSS

Feature	Frequency Hopping Spread-Spectrum (FHSS)	Direct Sequence Spread-Spectrum (DSSS)
Merits	• Can withstand jamming • Less affected by near-far problem • Less affected by multi-access interference	• Simpler to implement • Low probability of interception • Can withstand multi-access interference
Demerits	• Frequency acquisition may be difficult • Needs forward error correction scheme to combat channel noise	• Code acquisition may be difficult • Affected by jamming • Susceptible to near-far problem

6.2.6 Salient Features of Spread-Spectrum Systems

High Processing Gain Processing gain (the ratio of transmitted RF bandwidth to the baseband information bandwidth) of a typical commercial direct sequence spread-spectrum communication system is 11–16 dB, depending on the data rate.

Low Interference Spread-spectrum signals are less likely to interfere with other narrowband radio communications due to spread of energy over a wide band.

Immunity to Jamming Spread-spectrum signals are difficult to detect on narrowband receivers because the signal's energy is spreaded over a bandwidth of may be 100 times the information bandwidth.

Easy Encryption Spread-spectrum systems can be used for encrypting the signals. Only a receiver having the knowledge of spreading code can recover the encoded information.

Greater Security Spread-spectrum signals are difficult to exploit or spoof. Signal exploitation is the ability of undesired interceptor to use information. Spoofing is the act of maliciously sending misleading messages to the network. Spread-spectrum signals are naturally more secure than narrowband radio communications.

---

[3] The BPSK modulation, spectrum spreading and despreading, and coherent BPSK demodulation are linear operations. The incoming data sequence and the PN sequence needs to be synchronized for proper operation.

Multiple Access Several users can independently use the same higher bandwidth with very little interference. This property is used in CDMA cellular communication applications.

Low Probability of Interception The transmitted energy remains the same, but the wideband DSSS signal looks like noise to any receiver that does not know the signal's code sequence. So there is extremely low probability of intercepting this signal.

Increased Tolerance to Interference DSSS processes a narrowband information signal to spread it over a much wider bandwidth. Spreading the information signal desensitizes the original narrowband signal to some potential interference to the channel.

Increased Tolerance to Multipath Increased tolerance to interference also means increased tolerance to multipath interference. In fact, multipath energy may be used to the advantage to improve the system performance.

Increased Ranging Capability Timing error is directly proportional to the range error and inversely proportional to the signal bandwidth. This property enables DSSS signal to measure distance or equipment location, through a method known as *triangulation*.

Self-Assessment Exercise linked to LO 6.2

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/163

Q6.2.1 Wideband frequency spectrum is generated in a frequency-hopping spread-spectrum (FHSS) technique. Implement it depicting its principle of operation. ○●●

Q6.2.2 Frequency hopping spread-spectrum can be broadly classified into fast and slow FHSS systems. Distinguish between them. ○○●

Q6.2.3 Retrieve the most common aspect between direct-sequence and frequency-hopping spread-spectrum techniques. ○○●

Q6.2.4 Predict the processing gain of the frequency-hopping spread-spectrum communication system if the information bit rate is 1500 bps and the bandwidth of spread-spectrum signal is 21.5 MHz. ○●●

Q6.2.5 Analyze the minimum number of frequencies required for a frequency-hopping spread-spectrum communication system if the frequency multiplication factor is 8. ○●●

Q6.2.6 There are two distinct modulation schemes employed in direct-sequence spread-spectrum technique. Organize them properly with the help of functional block schematic diagram. ○●●

Q6.2.7 A direct-sequence spread-spectrum communication system has a binary information data rate of 7.5 kbps, and a PN code rate of 192 Mcps. If QPSK modulation is used instead of BPSK, then check the processing gain. ●●●

Q6.2.8 "DSSS technique is preferred over FHSS technique for bursty data transmission". Evaluate this statement based on criteria of generating the same. ○●●

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 6.2: *Illustrate the concept of frequency hopping spread spectrum (FHSS) and direct sequence spread spectrum (DSSS) with coherent BPSK.*

Mid-Chapter Check

So far you have learnt the following:

- Principles of Spread-Spectrum Modulation
- PN Sequences and their Properties
- Frequency Hopping Spread-Spectrum (FHSS) Technique
- Direct Sequence Spread-Spectrum (DSSS) Technique

Therefore, you are now skilled to complete the following tasks:

LO 6.1

MQ6.1 Find the processing gain if the RF signal bandwidth of frequency-hopping spread-spectrum communication system is 129 MHz, and the information bit rate is 2800 bps.

MQ6.2 A speech signal is bandlimited to 3.3 kHz, and uses 128 quantization levels to convert it into digitized analog information data. This is required to be transmitted by a pseudorandom noise spread-spectrum communication system. Calculate the chip rate needed to obtain a processing gain of 21 dB.

MQ6.3 Create properties of a truly random binary maximum-length sequence.

MQ6.4 "Orthogonal codes play a major role in spread-spectrum techniques and permit a number of signals to be transmitted on the same nominal carrier frequency and occupy the RF bandwidths". Justify the statement.

LO 6.2

MQ6.5 Select the additional hardware needed with frequency-hopping spread-spectrum transmitter as compared to that of direct-sequence spread-spectrum transmitter.

MQ6.6 Outline the necessity of employing error-correction coding scheme in a frequency hopping spread-spectrum system.

MQ6.7 An FH-BPSK system using 32 orthogonal codes serves 40 users. Design the signal-to-interference ratio (SIR) per user if the system bandwidth is 20 MHz, and the system operates at the signaling rate of 32 kbps.

MQ6.8 Plan the clock rate of PN code-generator for a frequency-hopping spread-spectrum communication system utilizing fast-hop technique at the hop rate of 10 hops per information bit, having 512 number of different frequencies, information bit rate of 2800 bps, and final RF multiplication factor of 9.

MQ6.9 Paraphrase the significance of direct-sequence spread-spectrum (DSSS) modulation.

MQ6.10 How is direct-sequence spread-spectrum used in pass band transmission? Mention the nature of spread-spectrum signal at the DSSS receiver.

Sure of what you have learnt so far? For answers, scan the QR code

OR

visit http://qrcode.flipick.com/index.php/42

Be ready for the next sections!

LO 6.3

6.3 APPLICATION OF DSSS IN A MULTIUSER SYSTEM

CDMA—A Form of Direct Sequence Spread-spectrum Modulation

Code Division Multiple Access (CDMA) is a form of direct-sequence spread-spectrum modulation in which multiple users are allowed to use the available RF spectrum. The baseband signals of different users are spreaded with a specific PN code to distinguish it from other signals. In a CDMA system,

- Individual users occupy the complete spectrum whenever they transmit
- Many users can occupy the same spectrum at the same time
- All users transmit information simultaneously by using the same carrier frequency
- Each user has its own PN code, which is orthogonal to PN codes of other users

To detect the information, the receiver should know the exact PN code used by the transmitter and perform a time correlation operation. All other codes appear as noise due to de-correlation

What We Discuss Here .....

Multiuser DS-CDMA system in which each mobile user has its own unique code and is used for generation and detection of spread-spectrum signal.

- Code acquisition and tracking for acquiring the timing information of the transmitted spread-spectrum signal so as to synchronize the transmitter and receiver to within an uncertainty of one chip duration.
- Multiple-access interference at the individual receiver in a cellular CDMA system and near-far interference due to wide range of signal levels received at the cell site from different mobile users located very close or far away from it within its service area.
- Performance of DS-CDMA multiuser system in terms of probability of error, jamming margin, capacity of a single-cell CDMA system and performance improvement factor.

Dear student... *Code Division Multiple Access (CDMA) technique is discussed in Section 6.4.*

6.3.1 Multiuser DS-CDMA System

Spread-Spectrum Modulation—A Basis for CDMA

In fact, *spread-spectrum* is a modulation technique that forms the basis for spread-spectrum multiple access, known as code-division multiple access (CDMA). In its most simplified form, a direct-sequence spread-spectrum (DSSS) transmitter spreads the signal power over a spectrum which is wider than the spectrum of the information signal. In other words, an information bandwidth of R_b occupies a transmission bandwidth of B_c. The DSSS receiver processes the received signal with a processing gain, $G_p = B_c/R_b$.

Typical processing gains for spread-spectrum systems lie between 20 and 60 dB. With a spread-spectrum system, the noise level is determined both by the thermal noise and by interference. For a given user, the interference is processed as noise alone. The input and output signal-to-noise power ratios are related as

$$\boxed{\left(\frac{S}{N}\right)_o = G_p \times \left(\frac{S}{N}\right)_i}$$

But $(S/N)_i$ is related to the E_b/N_o, where E_b is the energy per bit and N_o is the noise power spectral density including both the thermal noise and interference. With spread-spectrum systems, interference is transformed into noise.

$$\left(\frac{S}{N}\right)_i = \frac{(E_b \times R_b)}{(N_o \times B_c)}$$

$$\Rightarrow \qquad \left(\frac{S}{N}\right)_i = \left(\frac{E_b}{N_o}\right) \times \left(\frac{R_b}{B_c}\right)$$

$$\left(\frac{S}{N}\right)_i = \left(\frac{E_b}{N_o}\right) \times \left(\frac{1}{G_p}\right)$$

$$\Rightarrow \qquad \left(\frac{E_b}{N_o}\right) = G_p \times \left(\frac{S}{N}\right)_i$$

$$\Rightarrow \qquad \left(\frac{E_b}{N_o}\right) = \left(\frac{S}{N}\right)_o$$

The minimum E_b/N_o value required for proper system operation can be defined if the performance of the coding methods used on the signals, bit error rate, and the tolerance of the digitized voice signals are known. The best performance of the system can be obtained by maintaining the minimum E_b/N_o required for operation. For a given bit-error probability, the actual E_b/N_o ratio depends on the radio system design and error-correction coding technique used. The measured value of E_b/N_o may be closer, if not equal, to the theoretical value.

What is CDMA Technique?

In CDMA technique, one unique PN code is assigned to each user and distinct PN codes are used for different users. This PN code is employed by a user to mix with each information data bit before it is transmitted. The same PN code is used to decode these encoded bits, and any mismatch in code interprets the received information as noise. In a CDMA system, different spread-spectrum codes are generated by PN code generator and assigned to each user, and multiple users share the same frequency. A CDMA system is based on spread-spectrum technology by spreading the bandwidth of modulated signal substantially, which makes it less susceptible to the noise and interference.

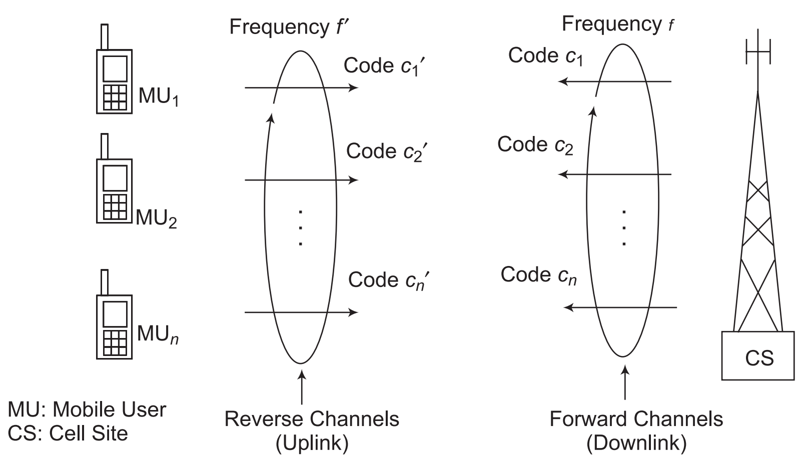


Figure 6.3.1 A Structure of a CDMA System

How it Functions

Each mobile receiver is provided the corresponding PN code so that it can decode the data it is expected to receive. Theoretically, the number of mobile users being served simultaneously is determined by the number of possible orthogonal codes that could be generated. Each active mobile user (MU) is a source of noise to the receiver of other active mobile users because of unique code assignment. If the number of active mobile users is increased beyond a certain limit, the whole CDMA system collapses because the signal received in a receiver will be much less than the noise caused by many other mobile users.

IMPORTANT!

CDMA cellular systems are implemented based on the spread-spectrum technology. The main concern in CDMA system is how many active mobile users can simultaneously use it before the system collapses! It is quite apparent that using a wider bandwidth for a single communication channel may be regarded as disadvantage in terms of effective utilization of available spectrum.

DSSS Form of CDMA

The DSSS form of CDMA is generated by combining each of the baseband signals to be multiplexed with a PN sequence at a much higher data rate. Each of the signals to be multiplexed should use a different PN sequence. If various PN sequences are orthogonal, the individual baseband signals can be recovered exactly without any mutual interference. However, the number of possible orthogonal sequences of codes is limited and depends on the length of the sequence. If the PN sequences are not orthogonal, CDMA is still possible. But there will be some mutual interference between the signals which may result in an increased noise level for all signals. As the number of non-orthogonal signals increases, the signal-to-noise ratio becomes too low and the bit-error rate too high. This may lead to unacceptable operation of the system.

Figure 6.3.2 depicts CDMA configuration as a multiplexing technique used with direct-sequence spread-spectrum.

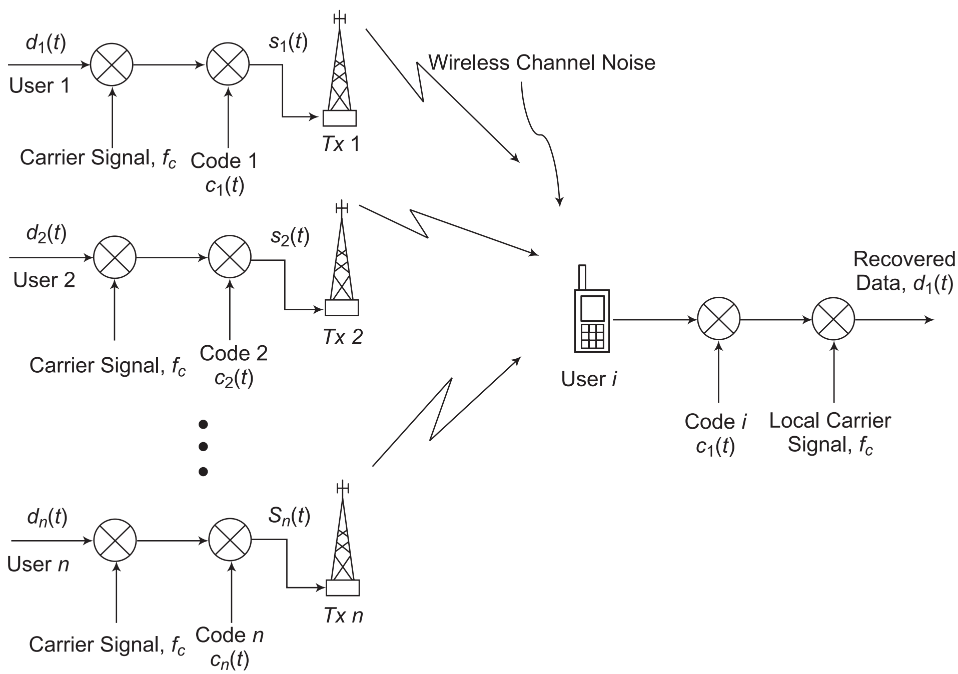


Figure 6.3.2 CDMA in a DSSS Environment

Interpretation

Let there are n users, each transmitting the DSSS signal using its unique orthogonal PN code sequence. For each user, the information data is BPSK modulated and then multiplied by the spreading code for i^{th} user, $c_i(t)$. All these DSSS signals plus channel noise are received at the desired user. Suppose that the CDMA receiver is attempting to recover the data of user 1. The received signal is demodulated and then multiplied by the spreading code of user code 1. The incoming wideband signal is converted to narrow bandwidth of the original information data signal. All other received signals are orthogonal to the spreading code of User 1. Thus, the undesired signal energy remains spreaded over a large bandwidth and the desired signal is concentrated in a narrow bandwidth. The bandpass filter used at the demodulator can therefore recover the desired signal.

Multiuser Detection

In a multiuser DS-CDMA system, each mobile user has its own unique code which is used to spread and despread its information data. The codes assigned to other users produce very small signal levels (like noise). As the number of users increase, the multiuser interference increases for all of the users. If the number of users increase to a point that mutual interference increases among all mobile users then it stops the proper operation for all of them. The spreading codes are selected to be perfectly orthogonal with one another so as to achieve the single-user performance in the multiuser case. Therefore, the design of perfectly orthogonal codes for all users is the most critical system parameter.

Application

Second-generation digital multi-user cellular systems such as IS-95, and most of the third-generation cellular systems use CDMA technique.

SOLVED EXAMPLE 6.3.1 Processing Gain of a DS-CDMA System

A DSSS system has a 10 Mcps code rate and 4.8 kbps information data rate. If the spreading code generation rate is increased to 50 Mcps, how much improvement in the processing gain of this DS-CDMA system will be achieved? Is there any advantage in increasing the spreading code generation rate with a 4.8 kbps information data rate? Comment on the results obtained.

Solution We know that in DSSS system, the RF bandwidth is same as spreading code rate.

For given code rate, R_c = 10 Mcps, RF bandwidth, B_c = 10 MHz

Information data rate, R_b = 4.8 kbps or 4.8×10^3 bps (Given)

Processing gain, $G_p = B_c/R_b = (10 \times 10^6)/4.8 \times 10^3\ )$

Therefore, Processing gain, $G_p = 2.1 \times 10^3$

Expressing it in dB, $G_p = 10 \log (2.1 \times 10^3) = 33.2$ dB

Processing gain, G_p at 50 Mcps code rate = $(50 \times 10^6)/4.8 \times 10^3\ ) = 1.04 \times 10^4$

Expressing it in dB, $G_p\ = 10 \log (1.04 \times 10^4) = 40.2$ dB

Hence, Improvement in processing gain = 40.2 – 33.1 = **7.1 dB** **Ans.**

***Comment on the Results*:** The improvement in processing gain is only 7.1 dB after enhancing the spreading code rate by 5 times (50 Mcps/10 Mcps). The circuit complexity needed to get five times the spreading code rate is too high for an improvement of 7.1 dB in processing gain. So there is not much advantage in this case.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 6.3.2 A DSSS system has a 20 Mcps code rate and 9.6 kbps information data rate. If the spreading code generation rate is increased to 100 Mcps, how much improvement in the processing gain of this DS-CDMA system will be achieved? Is there any advantage in increasing the spreading code generation rate with a 9.6 kbps information data rate? Comment on the results obtained.

6.3.2 Code Acquisition and Tracking

Process of Acquiring Timing Information

The process of acquiring the timing information of the transmitted spread-spectrum signal is achieved by initial code acquisition which synchronizes the transmitter and receiver to within an uncertainty of one chip duration, followed by code tracking which performs and maintains fine synchronization. The timing information of the transmitted signal is required to despread the received signal and demodulate the despread signal.

Initial Code Acquisition

Compared to code tracking, initial code acquisition in a spread-spectrum system is usually very difficult due to many factors such as

- ***Timing Uncertainty.*** It is basically determined by the transmission time and the propagation delay which can be much longer than chip duration. As initial code acquisition is usually achieved by a search through all possible phases of the sequence, a larger timing uncertainty means a larger search area.
- ***Frequency Uncertainty.*** Frequency uncertainty is due to Doppler shift (due to mobile user), and mismatch between the transmitter and receiver oscillators.
- ***Low Signal-to-Noise Ratio Environments.*** Most of the time, the DS or FH spread-spectrum systems are required to operate in low signal-to-noise ratio environments.
- ***Presence of Jammers.*** There may be situations that the DS or FH spread-spectrum systems may operate in presence of jammers.
- ***Channel Fading.*** The received signal levels fluctuate due to mobile radio channel fading.
- ***Multiple Access Interference.*** It is caused by other mobile users operating within the same cell as well as at the mobile user in a cell due to reuse of the same CDMA channel in the neighboring cells.

IMPORTANT!

In many practical systems, additional information such as the time of the day and an additional control channel, is needed to help achieve code acquisition.

Serial Search Code Acquisition

Serial search-code acquisition approach involves the use of matched filter or a single correlator to serially search for the correct phase of DSSS signal or the correct hopping pattern of FHSS signal. Figure 6.3.3 shows a functional block schematic of serial search-code acquisition approach for DSSS signal.

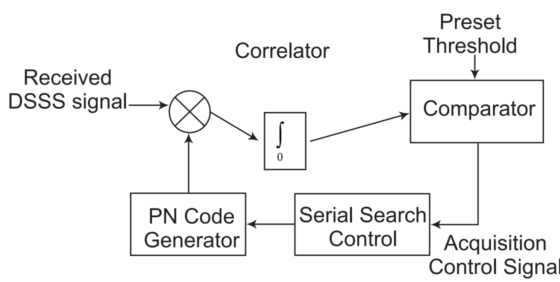


Figure 6.3.3 DSSS Serial Search-Code Acquisition

Functional Description

The locally generated PN code sequence is correlated with the received code signal. The timing epoch of the local PN code is set. At a fixed interval of time, called *search dwell time*, the output of the correlator (or integrator) is compared to a preset threshold in the comparator. If the output is below the threshold, the phase of the locally generated PN code generator is incremented by one-half of a chip in feedback loop, and if it is above the threshold, the phase of the received

PN code is assumed to have been acquired. Serial search-code acquisition approach is usually used because it requires only one single correlator or matched filter (as compared to number of correlators in parallel search approach).

Figure 6.3.4 shows a functional block schematic of serial search-code acquisition approach for FHSS signal.

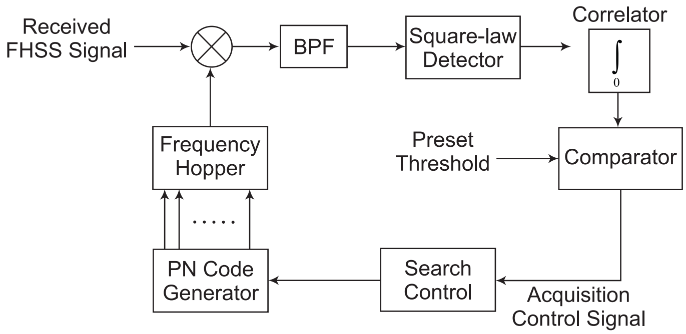


Figure 6.3.4 FHSS Serial Search-Code Acquisition

Functional Description

The locally generated PN code sequence is passed through a frequency hopper and then correlated with the received code signal. The PN code generator controls the frequency hopper. The correlator includes bandpass filter, square-law detector, and integrator. At search dwell time, the output of the integrator is compared to a preset threshold in the comparator. If the output is below the threshold, the frequency-hopping sequence of the frequency hopper is incremented. The process is repeated in a feedback loop system until it is aligned with that of the received signal.

Code Tracking

The purpose of *code tracking* is to perform and maintain fine synchronization. Given the initial code acquisition, code tracking is usually accomplished by a delay lock loop. The *tracking loop* keeps on operating during the entire communication period. The *delay lock loop* loses track of the correct timing if the channel changes abruptly. In that case the initial code acquisition has to be performed again.[4]

6.3.3 Multiple-Access and Near-far Interference

Multiple-Access Interference

Multiple Access Interference (MAI) is defined as the sum of interference caused by other users operating within the same cell (*intracell interference*) and the interference caused at the mobile user in a cell due to reuse of the same channel in the neighboring cells (*intercell interference*).

$$I_{\text{MAI}} = I_{\text{intracell}} + I_{\text{intercellular}}$$

In CDMA systems, the same frequency channel can be reused in the adjacent cell provided MAI has to be kept below a given threshold level necessary to meet the signal quality requirement. Ultimately, the MAI in a cellular CDMA system is more significant at the individual receiver.

---

[4] It is sometimes mandatory to perform initial code acquisition periodically irrespective of whether the tracking loop loses track or not. After the correct code phase is acquired by the code tracking circuitry, a standard Phase Lock Loop (PLL) can be employed to track the carrier frequency and phase.

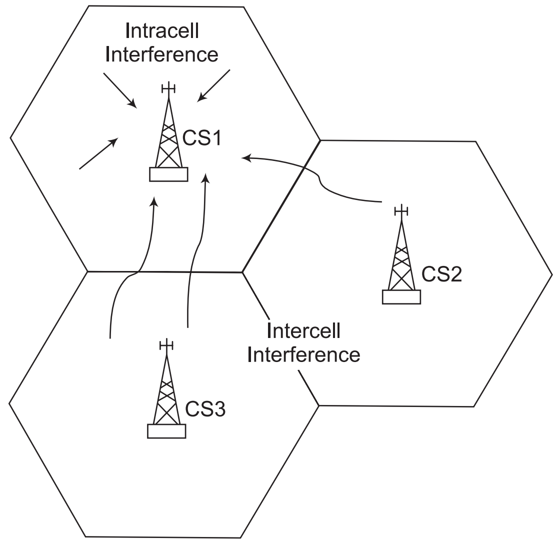


Figure 6.3.5 Intracell and Intercell Interference in Hexagonal Cells

Channel Loading

Mathematically, the intracell interference is given by

$$I_{\text{intracell}} = \left[\frac{(M-l)}{Q}\right] \times E_b$$

where M is the number of simultaneous users, Q is number of chips per time period, and E_b is the common received power level.

Since $M \gg 1$, the expression can be written as

$$I_{\text{intracell}} = \left(\frac{M}{Q}\right) \times E_b$$

where M/Q is termed as *channel loading*.

IMPORTANT!

Most of intercell interference occur from the first and second tiers of the surrounding cells of the serving cell. The interference from more distant cells suffers more propagation attenuation, and hence, can be ignored.

Intercell Interference Factor

The signals causing intercell interference are received at different power levels, because they are power controlled relative to other cell sites. As a consequence, intercell interference depends on the propagation losses from a mobile user to two different cell sites. In general, the relative power from mobile users in other cells will be attenuated relative to the power from the intracell mobile users due to larger distance.

Assuming identical traffic loading in all cells, the relative *intercell interference factor* is defined as the ratio of intercellular interference to intracell interference. That is,

$$\sigma = \frac{I_{\text{intercellular}}}{I_{\text{intracell}}}$$

The value of intercell interference factor σ ranges from 0.5 to 20, depending upon the operating environmental conditions.

Therefore, $I_{\text{MAI}} = I_{\text{intracell}} + \sigma I_{\text{intracell}}$

$$\Rightarrow \quad I_{\text{MAI}} = \left(\frac{M}{Q}\right) \times E_b + \sigma \times \left(\frac{M}{Q}\right) \times E_b$$

$$\Rightarrow \quad \boxed{I_{\text{MAI}} = (1+\sigma)\left[\left(\frac{M}{Q}\right) \times E_b\right]}$$

Thus, the MAI is directly proportional to the channel loading or capacity, M/Q.

Signal-to-Interference-plus-Noise Ratio (SINR)

The *signal-to-interference-plus-noise ratio* (SINR) at the individual receiver is given by

$$\text{SINR} = \frac{E_b}{(N_o + I_{\text{MAI}})}$$

$$\Rightarrow \quad \text{SINR} = \frac{E_b}{I_{\text{MAI}}\left(\dfrac{N_o}{I_{\text{MAI}}} + 1\right)}$$

CDMA cellular systems are often *interference limited*; that is, the operating conditions are such that $I_{\text{MAI}} > N_o$ (typically 6 to 10 dB higher). The I_{MAI}/N_o ratio depends upon the cell size. With large cells and battery-operated mobile phones, most of the transmit power is used to achieve the desired range. Thus, large cells tend to be noise limited. Smaller cells tend to be interference limited, and the interference level at the receiver is typically greater than the noise level of the receiver.

$$\Rightarrow \quad \text{SINR} = \frac{E_b}{(1+\sigma)\left[\left(\dfrac{M}{Q}\right) \times E_b\right]\left(\dfrac{N_o}{I_{\text{MAI}}} + 1\right)}$$

$$\Rightarrow \quad \boxed{\text{SINR} = \frac{1}{(1+\sigma)\left(\dfrac{M}{Q}\right)\left(\dfrac{N_o}{I_{\text{MAI}}} + 1\right)}}$$

This expression shows the three system design factors that affect the SINR at the receiver, and limit the spectral efficiency. The three factors are the following:

- ***Intercell interference, σ:*** It depends on the environment as well as on the handover technique.
- ***Channel loading, M/Q:*** It is a design parameter that needs to be maximized in a commercial cellular system.
- ***Operating I_{MAI}/N_0.*** It is related to cell size.

There is a trade-off between these three system design parameters. For example, for a constant SINR, moving from a noise-limited system ($I_{\text{MAI}}/N_0 = 0$ dB) to an interference-limited system ($I_{\text{MAI}}/N_0 = 10$ dB, say), the permissible channel loading or capacity increases. The channel loading must be significantly decreased to support a noise-limited system at the same SINR. Thus, large cells must have lighter load than small cells. Similarly, for a constant SINR, increasing intercell interference significantly reduces the permissible channel loading.[5]

[5] The required SINR in CDMA systems can be reduced by using RAKE receivers and FEC coding. However, soft hand-offs are the key design aspect to keep intercell interference low. On the other hand, reducing the SINR required by the receiver can significantly improve the permissible channel loading.

SOLVED EXAMPLE 6.3.3 **Number of Simultaneous Users**

A CDMA cellular system is interference limited with a ratio of total intracell-plus-intercell interference to receiver noise of 6 dB. Compute the average number of active users allowed per cell if the relative intercell interference factor is 0.55 and the required SINR at the receiver is 8 dB.

Solution Intracell-plus-intercell interference to receiver noise, $I_{MAI}/N_o = 6$ dB (Given)

Using I_{MAI}/N_o dB) = 10 log [I_{MAI}/N_o (ratio)], I_{MAI}/N_o (ratio) = 3.98

or, N_o/I_{MAI} (ratio) = 1/3.98 = 0.25

The required SINR at the receiver = 8 dB (Given)

Expressing it in ratio, we get SINR = antilog (8/10) = 6.3

The total spreading factor, $Q = 128$ (Standard)

Relative intercell interference factor, $\sigma = 0.55$ (Given)

We know that
$$\text{SINR} = \frac{1}{(1+\sigma)\left(\frac{M}{Q}\right)\left(\frac{N_o}{I_{MAI}}+1\right)}$$

The average number of users per cell, M is given by

$$M = \frac{Q}{(1+\sigma)\left(\frac{N_o}{I_{MAI}}+1\right)(\text{SINR})}$$

$$M = \frac{128}{(1+0.55)(0.25+1)(6.3)} = 10.5 \text{ users per cell}$$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 6.3.4 A CDMA cellular system is interference limited with a ratio of total intracell-plus-intercell interference to receiver noise of 9 dB. Compute the average number of active users allowed per cell if the relative intercell interference factor is 0.55 and the required SINR at the receiver is 12 dB.

RAKE Receiver—A Solution to Multipath Interference

The RAKE receivers used in CDMA mobile receivers attempt to recover the signals from multiple paths and then combine them with suitable delays to provide a robust signal reception in a hostile mobile radio environment. The multiple versions of a signal arrive more than one chip interval apart from each other. The mobile receiver can recover the signal by correlating the chip sequence with the dominant received signal. The remaining signals are treated as noise. The effect of multipath interference can be reduced by combining direct and reflected signals in the receiver.

Figure 6.3.6 illustrates the principle of the RAKE receiver.

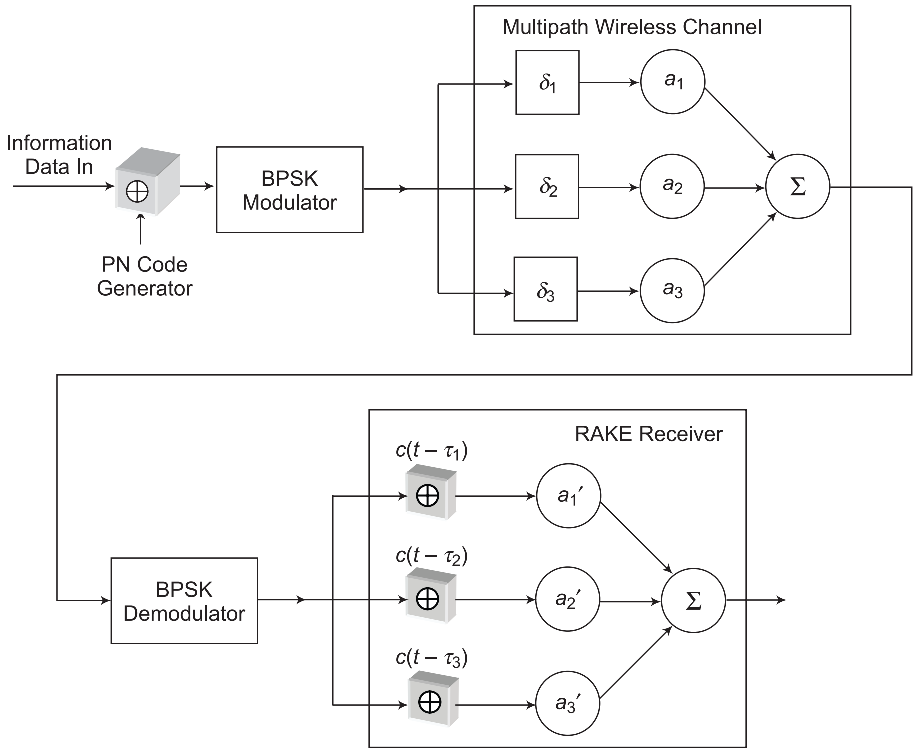


Figure 6.3.6 Principle of RAKE Receiver

Interpretation

The original information signal in the binary form to be transmitted is spreaded by the Exclusive-OR (XOR) operation with the PN spreading code. The spread bit sequence is then modulated for transmission over the wireless channel. The wireless channel generates multiple copies of the same signal due to multipath effects. Each one has a different amount of time delay (δ_1, δ_2, etc.), and attenuation factors (a_1, a_2, etc.). At the receiver, the combined signal is demodulated. The demodulated chip stream is then fed into multiple correlators, each delayed by a different amount. These signals are then combined using weighting attenuator factors (a_1', a_2', etc.) estimated from the channel characteristics.

A typical RAKE receiver structure for a DS-CDMA multiuser system is shown in Figure 6.3.7.

Operation of the RAKE Receiver

The received signal is passed through a tapped-delay line. The signal at each tap is passed through a correlator. The outputs of the correlators are then brought together in a diversity combiner. Its output is the estimate of the transmitted information symbol. In RAKE receiver, the delays between the consecutive fingers of the RAKE receiver are fixed at half of the chip duration. This provides two samples of the overall correlation function for each chip period. For a rectangular shaped chip pulse with triangular correlation function, there will be four samples. It is not possible to capture all the major peaks of the correlation function because the peaks are not aligned precisely at multiples of the sampling rate. But a RAKE receiver implemented with a sufficiently large number of fingers will provide a good approximation of all major peaks. An algorithm is used in digitally implemented RAKE receivers having few fingers to search for some major peaks of the correlation function and then adjust the finger locations accordingly.

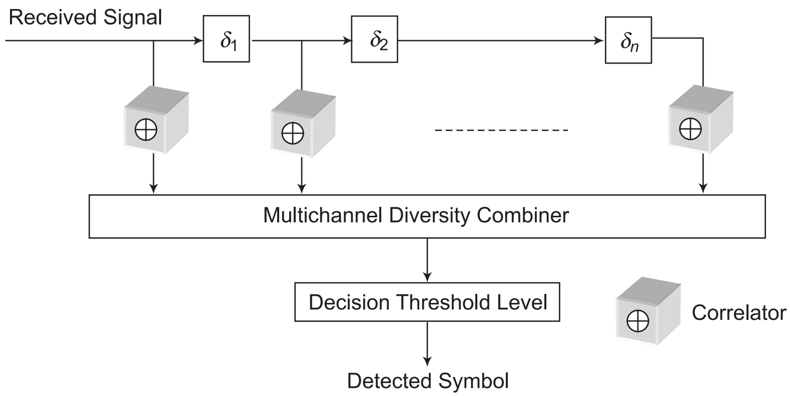


Figure 6.3.7 The RAKE Receiver Structure

Practical Aspects

The mobile user receives the signal transmitted from the serving cell site through several paths with different propagation delays in a mobile radio environment. The mobile unit has the capability of combining up to three RF signals—one signal which is received directly from the serving cell site, the other two signals may be copies of the same signals received after reflections from the structures between the cell-site transmitter and mobile receiver, or may also be received from neighboring cell sites operating in the same frequency. The cell-site receiver can combine up to four signals—the direct signal from the mobile user and three copies of the signals received after reflection from nearby buildings.[6]

SOLVED EXAMPLE 6.3.5 **Delay Calculations in Rake Receiver**

A rake receiver in a CDMA mobile phone receives a direct signal from a cell site A located 1 km away and a reflected signal from a building 0.5 km behind the mobile. It also receives a direct signal from another cell site B located 3 km away, as shown in Figure 6.3.8.

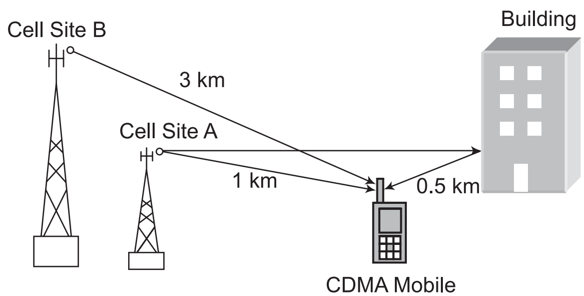


Figure 6.3.8 CDMA Mobile Rake Receiver Operation

Calculate the amount of time delay each "finger" of the CDMA mobile receiver needs to be applied.

[6] A RAKE receiver is capable of combining the received signal paths using any standard diversity combiner technique such as a selective, equal-gain, square-law, or maximal-ratio combiner. A maximal-ratio combining RAKE receiver does not introduce intersymbol interference. It provides optimum system performance in the presence of time diversity. The maximal-ratio combiner weights the received signal from each branch by the signal-to-noise ratio at that branch.

Solution Distance traveled by direct signal from cell site A, $d_1 = 1$ km (Given)

Distance between cell-site A and building, $d_{11} = 1.5$ km (From the figure)

Distance between mobile and building, $d_{12} = 0.5$ km (Given)

Therefore, distance traveled by the reflected signal, $d_2 = d_{11} + d_{12}$

Hence, distance traveled by the reflected signal, $d_2 = 1.5 + 0.5 = 2$ km

Distance traveled by the direct signal from cell site B, $d_3 = 3$ km (Given)

The rake receiver finger receiving signal from cell site B which is 3 km away need not delay the signal. But the two signals (direct as well as reflected) received from cell-site A needs to be delayed enough to allow all the three signals to be synchronized.

Case 1. To calculate time delay for direct signal from cell site A, t_1

The direct signal from the cell site A has to be delayed a time equal to the propagation time for a distance of (3 km – 1 km =) 2 km. That is,

Time delay for direct signal from cell site A, $t_1 = (2 \times 10^3)/(3 \times 10^8) = 6.67$ μs **Ans.**

Case 2. To calculate time delay for reflected signal from cell site A, t_2

The reflected signal from the cell site A has to be delayed a time equal to the propagation time for a distance of (3 km – 2 km =) 1 km. That is,

Time delay for reflected signal from cell site A, $t_2 = (1 \times 10^3)/(3 \times 10^8) = 3.33$ μs **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 6.3.6 A rake receiver in a CDMA mobile phone receives a direct signal from a cell site A located 2 km away and a reflected signal from a building 1 km behind the mobile. It also receives a direct signal from another cell site B located 6 km away, as shown in Figure 6.3.9.

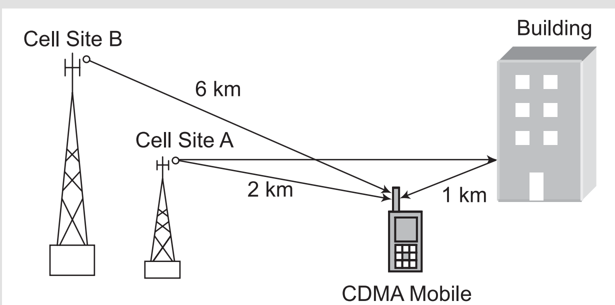


Figure 6.3.9 CDMA Mobile Rake Receiver Operation

Calculate the amount of time delay each "finger" of the CDMA mobile receiver needs to be applied.

Near-Far Interference

Near-far interference is a problem in CDMA cellular systems which are due to wide range of signal levels received at the cell site from different mobile users located very close or far away from it within the service area of the cell. In a DS-CDMA system, all traffic channels within one cell share the same radio channel simultaneously. Practically, some mobile users are near to the cell site while others are far away from it. A strong signal received at the cell site from a near-in mobile users masks the weak-signal received from a far-end mobile users. Figure 6.3.10 illustrates a situation in operation of a cellular system in which two mobile users (MUs) are communicating with the same cell site (CS).

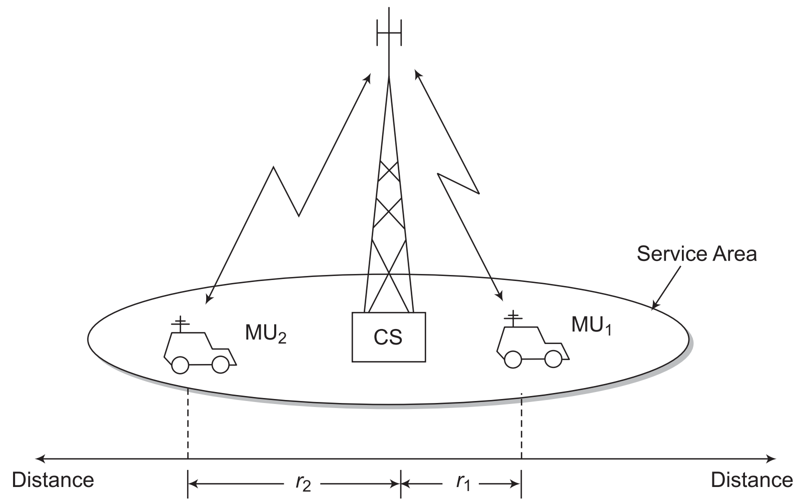


Figure 6.3.10 Illustration of Near-far Interference Problem

Interpretation

Assume that the transmitted power of each MU is the same and they are operating at adjacent channels. The received signal levels at the CS from the MU_1 and MU_2 are quite different due to the difference in the propagation signal path-lengths. If $r_1 < r_2$, then the received signal level from MU_1 will be much larger that the received signal level from MU_2 at the cell site. Out-of-band radiation of the signal from the MU_1 may interfere with the signal from the MU_2 in the adjacent channel. This effect, called *adjacent channel interference*, becomes serious when the difference in the received signal strength is large. This situation may lead to near-far interference.[7]

SOLVED EXAMPLE 6.3.7 Near-Far Interference Problem

In a given cellular system, the distance of a mobile user from the base station may range from 100 m to 10 km. Given a fourth-power propagation path loss in a mobile radio environment, what is the expected difference in received power levels at the base station if both mobile users transmit at the same power level? Comment on the results obtained.

Solution Let both mobile users transmit at identical power level, P_{tm}.

The received power level at the base station, $P_{rm} = \beta P_{tm}/d^{\gamma}$

where β is proportionality constant, d is the distance between transmitter and receiver, and γ is the propagation path-loss constant ($\gamma = 4$ … given).

The distance of the mobile 1 from base station, $d_1 = 100$ m (Given)

[7] The near-far interference problem leads to significant degradation in the quality performance of the system especially where spread-spectrum signals are multiplexed on the same frequency using low cross-correlation PN codes. The possible solution is to implement effective power control at mobile transmitter end.

Therefore, the received power level at mobile 1, $P_{rm1} = \beta P_{tm}/(100)^4$

The distance of the mobile 2 from base station, $d_2 = 10$ km or 10,000 m (Given)

Therefore, the received power level at the mobile 2, $P_{rm2} = \beta P_{tm}/(10000)^4$

Ratio of received power levels at the base station, $\dfrac{P_{rm1}}{P_{rm2}} = \dfrac{(100)^4}{(10000)^4} = (0.01)^4$

Expressed it in dB, $\dfrac{P_{rm1}}{P_{rm2}}(\text{dB}) = 10\log(0.01)^4 = -80$ dB **Ans.**

Comment on the Results: The difference in received power levels at the base station will be as large as 80 dB if both mobile users transmit at the same power level. This means that even in a CDMA system with a code spreading rate of 512, which is equivalent to a processing gain of 27 dB, the stronger mobile user (located nearer to cell–site) would prevent detection of the weaker mobile user (located far away from the same cell-site). This is near-far interference problem.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 6.3.8 In a given cellular system, the distance of a mobile user from the base station may range from 1 km to 10 km. Given a fourth-power propagation path loss in a mobile radio environment, calculate the estimated difference in received power levels at the base station if both mobile users transmit at the same power level? Comment on the results obtained.

Solution to Near-Far Interference Problem

Multiple-access as well as near-far interference is required to be minimized to achieve improvement in spectral efficiency and increase system capacity. This is directly related to controlling the transmitted RF power of each mobile user at all time of its operation. *Adaptive RF power control* schemes reduce the *near-far interference* problem, and optimize the system capacity and spectral efficiency.

RF Power Control in Multiuser System

RF power control is the technique of controlling the mobile transmit power so as to affect the cell-site received power, and hence the overall *carrier-to-interference* (C/I) value. An ideal power-control scheme ensures that the signals received at the cell site from all mobile users within the cell remain at the same power level. It is irrespective of propagation path loss, location and/or movement of the mobile users. It is desirable that the power received at the cell site from all mobile users served by it must be nearly equal. If the received signal power is too low, there is a high probability of bit errors and if the received signal power is too high, interference increases. In fact, power control is applied at both the mobile users as well as the cell site.

RF Power Control Mechanisms

There are several RF power control mechanisms that can be based on the signal strength received by the cell site or can depend on other system parameters. Accordingly, the cell site or the mobile user can either initiate the power control. There are mainly two types of RF power-control mechanisms: an *open-loop RF power control* and a *closed-loop RF power control*. Because all the traffic channels occupy the same frequency and time, the received signals from multiple mobile users located anywhere within the serving cell must all have the same received signal strength at the cell site for proper detection. A mobile user that transmits unnecessarily at a large power may jam or interfere with the received signals of all the other mobile users.

Open-loop RF Power Control

Open-loop RF power control refers to the procedure whereby the mobile user measures its received signal level and adjusts its transmit power accordingly. The cell sites are not involved in open-loop power-control mechanism. A mobile user closer to the cell site should transmit less power because of small path loss. Mobile users that are far away from a cell site should transmit at a larger transmit power to overcome the greater path loss in signal strength.

An *adaptive open-loop RF power control* is based on the measured signal strength of the received pilot signal by the mobile user. The mobile user then sets its transmit power after estimating the path-loss between the cell site transmitter and mobile receiver. It is assumed that the forward and reverse links have the same propagation path loss. However, this arrangement may not be precise and accurate.

Operation Mechanism of Open-loop RF Power Control

The operation of open-loop power control mechanism is quite simple. At the time of switching on the mobile user, it adjusts its transmit power based on the total received power from all cell sites on the pilot channel. If the received signal strength of the pilot channel is very strong, the mobile user transmits a weak signal to the cell site. Otherwise, it transmits a strong signal to the cell site. As a first approximation, the transmitted power of the mobile, P_{tm} (dBm) is set as

$$P_{tm}\text{ (dBm)} = -76\text{ dB} - P_{rm}\text{(dBm)}$$

where P_{rm} is the received power by the mobile. The mobile user begins by transmitting at the power level determined by the above expression and increases its transmit power if it does not receive acknowledgement from the cell site. This could happen if a substantial amount of the received power at the mobile user is actually from adjacent cells. Once a link with the nearest cell site is established, the open-loop power control setting is adjusted in 1 dB increments after every 1.25 ms by commands from the cell site. This ensures that the received power from all mobile users is at the same level. It may be noted that the power received at the cell site from all mobile users must be nearly equal, say within 1 dB, for the system to work properly.

Closed-loop RF Power Control

Closed-loop power control refers to the procedure whereby the cell site measures its received signal level and then sends corresponding message to the mobile user to adjust its transmit power to the desired level. Power-control message indicates either an increment or decrement in the transmit power of the mobile user.

Typically, power-control adjustments are sent as a single bit command.

- A logical 1 means that the transmit power be decreased by about 1 dB.
- A logical 0 means that the transmit power be increased by about 1 dB.

The power-control bits are sent at the rate of 800 Hz to 1500 Hz. In a practical multiuser communication system, the combination of the open- and closed-loop power control techniques must be carefully considered.

Disadvantages of RF Power Control Schemes in Multiuser Systems

- It may not be desirable to set transmission powers to higher values because battery power at the mobile user is a limited resource that needs to be conserved.
- Increasing the transmitted power on one channel, irrespective of the power levels used on other channels, can cause inequality of transmission over other channels.
- Power-control techniques are restricted by the physical limitations on the transmitter power levels not to exceed the unwanted radiations.

SOLVED EXAMPLE 6.3.9 Open-loop Power Control at CDMA Mobile Phone

A CDMA mobile phone measures the received signal power level from its serving cell site as –85 dBm.

(a) What should the mobile transmitter power be set to as a first approximation?

(b) Once the mobile transmitter power is set as computed in Part (a), its serving cell-site needs the mobile user to change its transmit power level to +5 dBm. How long will it take to make this change?

Solution

(a) Received signal strength by mobile, P_{rm} = –85 dBm (Given)

As a first approximation, the transmitter power of the mobile is given by the expression

$$P_{tm}\text{ (dBm)} = -76\text{ dB} - P_{rm}\text{(dBm)}$$

Therefore, P_{tm} (dBm) = –76 dB – (–85 dBm) = +9 dBm **Ans.**

(b) Required transmitter power of the mobile = +5 dBm

Difference in mobile transmitter levels = +9 dBm – (+5 dBm) = 4 dB

The mobile transmitter power level is adjusted by 1 dB step after every 1.25 ms. That is, time taken to adjust 1 dB power level = 1.25 ms

Number of steps required to adjust mobile transmitter level to +5 dBm = 4

Time needed to adjust mobile transmitter level to +5 dBm = 4 × 1.25 ms

Hence, time needed to adjust mobile transmitter level = 5 ms **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 6.3.10 A CDMA mobile phone measures the received signal power level from its serving cell site as –95 dBm.

(a) Find the mobile transmitter power to be set as a first approximation.

(b) Once the mobile transmitter power is set as computed in Part (a), its serving cell site needs the mobile user to change its transmit power level to +8 dBm. How long will it take to make this change?

6.3.4 Performance of DS-CDMA Multiuser System

A Model of DSSS BPSK System

Consider a model of direct-sequence spread-spectrum binary PSK (BPSK) system which involves both the spectrum spreading and the BPSK modulation.

Figure 6.3.11 shows such a model with input and output signals marked at each functional block.

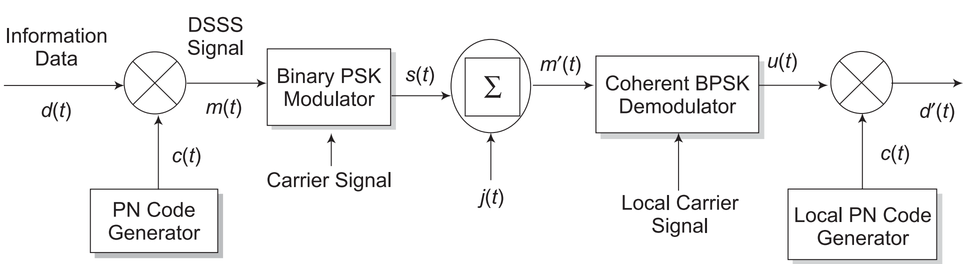


Figure 6.3.11 A Model of DS Spread-spectrum BPSK System

Mathematical Analysis

The information data signal, $d(t)$ modulates the carrier signal to produce BPSK signal, $s(t)$. The DSSS signal $m(t)$ is simply modulo-2 addition of digital binary phase modulated signal $s(t)$ and the PN sequence signal $c(t)$.

That is,

$$m(t) = s(t)\, c(t);\text{ where } s(t) = \pm\sqrt{\frac{2E_b}{T_b}}\cos(2\pi f_c t)$$

Note that the ± sign corresponds to information bits 1 and 0 respectively; E_b is the signal energy per bit; T_b is the bit duration; f_c is the carrier frequency which is assumed to be an integer multiple of $\frac{1}{T_b}$.

$$\therefore \qquad m(t) = \pm\sqrt{\frac{2E_b}{T_b}}\cos(2\pi f_c t)c(t)$$

$$\Rightarrow \qquad m(t) = \pm\sqrt{\frac{E_b}{N}}\sum_{k=0}^{N-1} c_k \phi_k(t); \quad 0 \le t \le T_b$$

where c_k is the PN code sequence $\{c_0, c_1, ..., c_{N-1}\}$ for the N-dimensional transmitted signal $m(t)$. It requires a minimum of N orthonormal functions for its representation; $\phi_k(t)$ is one of the set of orthonormal basis function, defined as

$$\phi_k(t) = \begin{cases} \sqrt{\frac{2}{T_c}}\cos(2\pi f_c t); & kT_c \le t \le (k+1)T_c \\ 0; & \text{otherwise} \end{cases}$$

The other set of orthonormal basis function is defined as

$$\widehat{\phi_k(t)} = \begin{cases} \sqrt{\frac{2}{T_c}}\sin(2\pi f_c t); & kT_c \le t \le (k+1)T_c \\ 0; & \text{otherwise} \end{cases}$$

where T_c is the chip duration; and $k = 0, 1, 2, ..., N - 1$

Let the effect of the channel noise be insignificant as compared to that due to interference signal introduced in the channel such that the system is interference limited only. Ignoring the effect of channel noise, the output of the channel is given by

$$m'(t) = m(t) + n_j(t)$$

$$\Rightarrow \qquad m'(t) = s(t)\, c(t) + n_j(t)$$

where $m'(t)$ is the output of the channel or the input to the receiver; $n_j(t)$ represents the interference signal (jammer) which may be represented in terms of orthonormal basis functions as

$$n_j(t) = \sum_{k=0}^{N-1} n_{j_k}\phi_k(t) + \sum_{k=0}^{N-1}\widehat{n_{j_k}}\,\widehat{\phi_k(t)}; \quad 0 \le t \le T_b$$

where $n_{j_k} = \int_0^{T_b} n_j(t)\phi_k(t)\,dt$; and $\widehat{n_{j_k}} = \int_0^{T_b} n_j(t)\widehat{\phi_k(t)}\,dt$; $k = 0, 1, 2, ..., N - 1$.

The interference signal $n_j(t)$ is $2N$-dimensional because $\sum_{k=0}^{N-1} n_{j_k}^2 \approx \sum_{k=0}^{N-1}\widehat{n_{j_k}^2}$.

In the receiver, the received signal $m'(t)$ is first multiplied by the local PN sequence signal $c(t)$ to produce the despread signal $u(t)$, that is,

$$u(t) = m'(t)\, c(t)$$

$$\Rightarrow \qquad u(t) = \left[s(t)c(t) + n_j(t)\right]c(t)$$

$$\Rightarrow \qquad u(t) = s(t)\,c(t)\,c(t) + c(t)\,n_j(t) = s(t)\,c^2(t) + c(t)\,n_j(t)$$

Since the PN sequence signal $c(t)$ alternates between –1 and +1, therefore $c^2(t) = 1$.

$$\therefore \qquad u(t) = s(t) + c(t)\,n_j(t)$$

This is the input to the coherent BPSK demodulator which consists of a binary PSK signal $s(t)$ plus the code-modulated interference signal $c(t)\,n_j(t)$. This implies that the spectrum of the interference signal is spreaded. The output of coherent BPSK demodulator $d'(t)$ is the detected receiver output bits which is an estimate of the information data bits. It can be represented as

$$d'(t) = \sqrt{\frac{2}{T_b}}\int_0^{T_b} u(t)\cos(2\pi f_c t)\,dt$$

$$\Rightarrow \qquad d'(t) = \sqrt{\frac{2}{T_b}}\int_0^{T_b} \left[s(t) + c(t)n_j(t)\right]\cos(2\pi f_c t)\,dt$$

$$\Rightarrow \qquad d'(t) = \sqrt{\frac{2}{T_b}}\int_0^{T_b} s(t)\cos(2\pi f_c t)\,dt + \sqrt{\frac{2}{T_b}}\int_0^{T_b} c(t)n_j(t)\cos(2\pi f_c t)\,dt$$

The first term is due to the despread BPSK signal $s(t)$, which is given as

$$s(t) = \pm\sqrt{\frac{2E_b}{T_b}}\cos(2\pi f_c t); \quad 0 \le t \le T_b$$

The second term is due to the spread interference $c(t)n_j(t)$.

Let the first term be denoted by v_s, and the second term by v_{cj}.

$$\therefore \qquad v_s = \sqrt{\frac{2}{T_b}}\int_0^{T_b}\left[\pm\sqrt{\frac{2E_b}{T_b}}\cos(2\pi f_c t)\right]\cos(2\pi f_c t)\,dt$$

Similarly, the second term due to spread interference is given by

$$v_{cj} = \sqrt{\frac{2}{T_b}}\int_0^{T_b} c(t)n_j(t)\cos(2\pi f_c t)\,dt = \sqrt{\frac{T_c}{T_b}}\sum_{k=0}^{N-1} c_k n_{j_k}$$

$$\Rightarrow \qquad v_s = \pm\sqrt{\frac{2}{T_b}}\sqrt{\frac{2E_b}{T_b}}\int_0^{T_b}\left[\cos^2(2\pi f_c t)\right]dt$$

$$\Rightarrow \qquad v_s = \pm\frac{2}{T_b}\sqrt{E_b}\int_0^{T_b}\left[\frac{1+\cos(4\pi f_c t)}{2}\right]dt$$

$$\Rightarrow \qquad v_s = \pm\frac{2}{T_b}\sqrt{E_b}\left[\frac{T_b}{2} + 0\right]; \text{ (since } f_c \text{ is an integral multiple of } \frac{1}{T_b}\text{)}$$

$$\Rightarrow \qquad v_s = \pm\sqrt{E_b}$$

where the + sign corresponds to the information data bit 1, and the – sign corresponds to the information data bit 0; and E_b is the signal energy per bit.

Probability of Error for DS/BPSK System

The peak instantaneous power of the signal component is E_b. The equivalent noise component contained in the coherent DSSS demodulator output may be approximated as a Gaussian random variable with zero mean and variance.

It is given as $P_{if}\frac{T_c}{2}$; where P_{if} is the *average interference power* and T_c is the chip duration. If E_b denote the bit-energy level, then the *probability of error* for DS/BPSK binary system with large spread factor N can be expressed as

$$\boxed{P_e \approx \frac{1}{2} erfc\left(\sqrt{\frac{E_b}{P_{if}T_c}}\right)}$$

For the purpose of comparison, the probability of error for a coherent binary PSK system is given as

$$P_e \approx \frac{1}{2} erfc\left(\sqrt{\frac{E_b}{N_o}}\right)$$

The interference in DS/BPSK binary system may be treated as wideband noise of power spectral density, that is, $\frac{N_o}{2} = \frac{P_{if}T_c}{2}$. Using $E_b = P_{\text{av}}T_b$ where P_{av} is the average signal power and T_b is the bit duration, the signal energy per bit-to-noise spectral density ratio can be expressed as

$$\frac{E_b}{N_o} = \frac{P_{\text{av}}}{P_{\text{if}}} \times \frac{T_b}{T_c}$$

$$\Rightarrow \qquad \frac{P_{\text{av}}}{P_{\text{if}}} = \frac{E_b/N_o}{T_b/T_c}$$

$$\Rightarrow \qquad \frac{P_{\text{if}}}{P_{\text{av}}} = \frac{T_b/T_c}{E_b/N_0} = \frac{G_p}{E_b/N_0}$$

where $\boxed{G_p = T_b/T_c}$ is the processing gain which represents the gain achieved by processing a spread-spectrum signal over an unspreaded signal.

Jamming Margin

By definition, the ratio of average interference power and average signal power, $\frac{P_{\text{if}}}{P_{\text{av}}}$ is termed the *jamming margin, J_m*. Therefore,

$$\boxed{J_{\text{m}} = \frac{G_p}{E_b/N_0}}$$

Expressing various terms in decibel form, we can write

$$\boxed{J_{\text{m}}\Big|_{\text{dB}} = G_p\Big|_{\text{dB}} - 10 \times \log_{10}\left(\frac{E_b}{N_o}\right)_{\text{min}}}$$

where $\left(\dfrac{E_b}{N_o}\right)_{\min}$ represents the minimum value of $\dfrac{E_b}{N_o}$ needed to support the prescribed value of the average probability of error under given operating conditions.

Capacity of a Single-Cell CDMA System

Let there be M simultaneous users on the reverse channel of a CDMA network. It is assumed that there is an ideal power control employed on the reverse channel so that the received power of signals from all mobile users has the same value P_r. Then, the received power from the target mobile user after processing at the cell-site receiver is $N \times P_r$. The received interference from $(M - 1)$ other mobile users is $(M - 1) \times P_r$.

Assume that a cellular system is interference-limited and the background noise is dominated by the interference noise from other mobile users. The received signal-to-interference ratio, S_r for the target mobile receiver will be

$$S_r \approx \frac{(N \times P)}{[(M - l) \times P]}$$

$$\Rightarrow \qquad S_r \approx \frac{N}{(M - l)}$$

All mobile users always have a requirement for the acceptable error rate of the received data stream. For a given modulation scheme and coding technique used in the system, that error-rate requirement will be supported by a minimum S_r requirement that can be used to determine the number of simultaneous users or capacity of the system.

Using $N = \dfrac{B_c}{R_b}$, we have

$$S_r = \left(\frac{B_c}{R_b}\right) \times \frac{1}{(M - 1)}$$

$$\Rightarrow \qquad M = \left(\frac{B_c}{R_b}\right) \times \left(\frac{1}{S_r}\right) + 1$$

$$\Rightarrow \qquad M = \left(\frac{B_c}{R_b}\right) \times \left(\frac{1}{S_r}\right)$$

The number of simultaneous users or capacity of a single-cell CDMA system is inversely proportional to the acceptable signal-to-interference ratio in the system. The relationship between the number of mobile users, M, the processing gain, G_p and the E_b/N_o ratio is, therefore, given as

$$\boxed{M = G_p \times \frac{1}{E_b/N_0}}$$

Performance Improvement Factor

In the practical design of DS-CDMA digital cellular systems, there are three system parameters that affect the capacity as well as the bandwidth efficiency of the system. These are the following:

1. ***Multi-user interference factor, ρ.*** It accounts for mobile users in other cells in the system. Because all neighboring cells in a CDMA cellular network operate at the same frequency, they will cause additional interference. The interference increase factor is relatively small due to the processing gain of the system and the distances involved. A value of $\rho = 1.6$ (2 dB) is commonly used in the practical CDMA cellular system.

2. ***Voice activity interference reduction factor, G_v.*** It is the ratio of the total connection time to the active talk time. Voice activity refers to the fact that the active user speaks only 40% of the time on an average. If a mobile phone transmits only when the user speaks then it will contribute to the interference just 40% of the total connection time. By considering voice activity, the system capacity increases by a factor of approximately 2.5 per cell, that is G_v is 2.5 (4 dB).
3. ***Cell sectorization, G_A.*** Cell sectorization refers to the use of 120°directional antennas in a 3-sector cellular system configuration. Cell sectorization reduces the overall interference, increasing the allowable number of simultaneous users by a sectorization gain factor, G_A= 2.5 (4 dB).

Incorporating these three factors, the number of simultaneous users that can be supported in a practical *multicell CDMA cellular system* can be approximated by

$$\boxed{M = \left(\frac{B_c}{R_b}\right) \times \left(\frac{1}{S_r}\right) \times \left(\frac{G_V \times G_A}{\rho}\right)}$$

where $\left(\frac{G_V \times G_A}{\rho}\right)$ is termed the *performance improvement factor* P_f in a CDMA digital cellular system due to multi-user interference, voice activation, and cell-sectorization factors in a cell.

Hence,
$$\boxed{M = \left(\frac{B_c}{R_b}\right) \times \left(\frac{1}{S_r}\right) \times P_f}$$

LET'S RECONFIRM OUR UNDERSTANDING!!
- What is meant by near-far interference?
- How is jamming margin related with processing gain?

SOLVED EXAMPLE 6.3.11 **CDMA Capacity per Carrier**

Given that the CDMA digital cellular systems require 3 dB < S_r < 9 dB which employs QPSK modulation scheme and convolutional coding technique. The bandwidth of the channel is 1.25 MHz, and the transmission data rate is R_b = 9600 bps. Determine the capacity of a single cell.

Solution Channel bandwidth, B_c = 1.25 MHz or 1250 kHz (Given)
The transmission data rate, R_b = 9600 bps or 9.6 kbps (Given)
To determine maximum number of simultaneous users, M_{max}
The minimum acceptable, $S_{r(\min)}$ = 3 dB (Given)
Converting $S_{r(\min)}$ = 3 dB in $S_{r(\min)\,\text{ratio}}$ by using the expression

$$S_{r(\min)\text{dB}} = 10 \log (S_{r(\min)\,\text{ratio}})$$
$$\Rightarrow \quad 3\text{dB} = 10 \log (S_{r(\min)\,\text{ratio}})$$
$$\Rightarrow \quad S_{r(\min)\,\text{ratio}} = 10^{\,3/10} = 10^{0.3} = 2$$

The maximum number of simultaneous users can be determined by using the expression

$$M_{\max} = \left(\frac{B_c}{R_b}\right) \times \left(\frac{1}{S_{r(\min)}}\right) = \left(\frac{1250 \text{ kHz}}{9.6 \text{ kbps}}\right) \times \left(\frac{1}{2}\right) \approx 65 \text{ users}$$

To determine minimum number of simultaneous users, M_{min}.

The maximum acceptable, $S_{r(\max)} = 9$ dB (Given)

Converting $S_{r(\max)} = 9$ dB in $S_{r(\max)\text{ ratio}}$ by using the expression

$$S_{r(\max)} \text{ dB} = 10 \log (S_{r(\max)\text{ ratio}})$$

or, $$9 \text{ dB} = 10 \log (S_{r(\max)\text{ ratio}})$$

or, $$(S_{r(\max)\text{ ratio}}) = 10^{9/10} = 10^{0.9} = 7.94$$

The minimum number of simultaneous users can be determined by using the expression

$$M_{\min} = \left(\frac{B_c}{R_b}\right) \times \left(\frac{1}{S_{r(\max)}}\right) \simeq \left(\frac{1250 \text{ kHz}}{9.6 \text{ kbps}}\right) \times \left(\frac{1}{7.94}\right) \approx 16 \text{ users}$$

Hence, a single cell IS-95 CDMA digital cellular systems system can support from **16 users** to **65 users.** **Ans.**

SOLVED EXAMPLE 6.3.12 Compute Performance Improvement Factor

Compute the performance improvement factor P_f in a CDMA digital cellular system, considering typical values of voice activation, cell-sectorization, and multi-user interference in a cell.

Solution The voice activity interference reduction factor, $G_V = 2.5$ (Typical)

The antenna sectorization gain factor, $G_A = 2.5$ (Typical)

The interference increase factor, $\rho = 1.6$ (Typical)

The performance improvement factor, P_f in CDMA cellular system is given by

$$P_f = \left(\frac{G_V \times G_A}{\rho}\right) = (2.5 \times 2.5)/1.6 = 3.9$$

Expressing it in dB, $\mathbf{P_f}$ **(dB)** = 10 log 3.9 **= 5.9 dB** **Ans.**

SOLVED EXAMPLE 6.3.13 Multicell CDMA Capacity Range

Using QPSK modulation and convolutional coding, the CDMA digital cellular systems require 3 dB < S_r < 9 dB. Determine the multicell CDMA capacity range if the performance improvement factor due to antenna sectorization, voice activity, and interference increase parameter is approximately 6 dB.

Solution Channel bandwidth, $B_c = 1.25$ MHz or 1250 kHz (Standard)

Transmission data rate, $R_b = 9600$ bps (Standard)

Performance improvement factor, $P_f = 6$ dB (Given)

Expressing it in ratio, we get $P_f = 10^{6/10}$ dB $= 10^{0.6} = 4$

The minimum acceptable, $S_{r(\min)} = 3$ dB (Given)

Expressing it in ratio, we get $S_{r(\min)} = 10^{3/10} = 10^{0.3} = 2$

The maximum number of simultaneous users,

$$M_{\max} = \left(\frac{B_c}{R_b}\right) \times \left(\frac{1}{S_{r(\min)}}\right) \times P_f$$

$$\Rightarrow \qquad M_{max} = \left(\frac{1250 \text{ kHz}}{9.6 \text{ kbps}}\right) \times \left(\frac{1}{2}\right) \times 4 = 260 \text{ users}$$

The maximum acceptable, $S_{r(max)} = 9$ dB (Given)

Expressing it in ratio, we get $S_{r(max)} = 10^{9/10} = 10^{0.9} = 7.94$

The minimum number of simultaneous users,

$$M_{min} = \left(\frac{B_c}{R_b}\right) \times \left(\frac{1}{S_{r(max)}}\right) \times P_f$$

$$\Rightarrow \qquad M_{min} = \left(\frac{1250 \text{ kHz}}{9.6 \text{ kbps}}\right) \times \left(\frac{1}{7.94}\right) \times 4 = 64 \text{ users}$$

Hence, a multicell CDMA digital cellular systems system can support from 64 users to 260 users, that is, $64 < M < 260$ **Ans.**

SOLVED EXAMPLE 6.3.14 Number of Mobile Users in CDMA System

Show that the number of mobile users that can be supported by a CDMA system using an RF bandwidth of 1.25 MHz to transmit data at 9.6 kbps is 33 mobile users per sector. Assume $E_b/N_o = 6$ dB; the interference from neighboring cells = 60%; the voice activity factor = 50%; the power control accuracy factor = 0.8.

Solution CDMA channel bandwidth, $B_c = 1.25$ MHz or 1250 kHz (Given)

Baseband data rate, $R_b = 9.6$ kbps (Given)

We know that processing gain, $G_p = \dfrac{B_c}{R_b} = \dfrac{1250 \text{ kHz}}{9.6 \text{ kbps}} = 130.2$

$E_b/N_o = 6$ dB or 3.98 (Given)

Interference factor, $\rho = 60\%$ or 0.6 (Given)

Power control accuracy factor, $\alpha = 0.8$ (Given)

Voice activity factor, $G_v = 50\%$ or 0.5 (Given)

Assuming omnidirectional system, the number of mobile users per cell is given by the expression

$$M = \frac{G_p}{E_b/N_o} \times \frac{1}{1+\rho} \times \alpha \times \frac{1}{G_v} = \frac{130.2}{3.98} \times \frac{1}{1+0.6} \times 0.8 \times \frac{1}{0.5} = 33 \text{ mobile users} \quad \textbf{Ans.}$$

SOLVED EXAMPLE 6.3.15 Processing Gain in CDMA System

A total of 36 equal-power mobile terminals share a frequency band through a CDMA system. Each mobile terminal transmits information at 9.6 kbps with a DSSS BPSK modulated signal which has E_b/N_o of 6.8 dB. Calculate the processing gain. Assume the interference factor from neighboring cells = 60%; the voice activity factor = 50%; the power control accuracy factor = 0.8.

Solution Number of mobile users, $M = 36$ users (Given)

$E_b/N_o = 6.8$ dB or 4.8 (Given)

Interference factor from neighbouring cell, $\rho = 60\%$ or 0.6 (Given)
Power control accuracy factor, $\alpha = 0.8$ (Given)
Voice activity factor, $G_v = 50\%$ or 0.5 (Given)
Assuming omnidirectional antenna at base stations ($G_A = 1$)

The number of mobile users per cell, $$M = \frac{G_p}{E_b/N_o} \times \frac{1}{1+\rho} \times \alpha \times \frac{1}{G_v} \times G_A$$

Therefore, $$36 = \frac{G_p}{4.8} \times \frac{1}{1+0.6} \times 0.8 \times \frac{1}{0.5} \times 1$$

Processing gain, $G_p = 172.8$ **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 6.3.16 Determine the maximum number of mobile users that can be supported in a single cell CDMA system using omnidirectional cell-site antenna and no voice activity detection. The system uses an RF bandwidth of 1.25 MHz to transmit data @ 9.6 kbps, and a minimum acceptable E_b/N_o is found to be 10 dB. Use $\rho = 0.5$, $G_v = 0.85$, $G_A = 1$. Assume the CDMA system is interference limited.

Ex 6.3.17 The CDMA system uses an RF bandwidth of 1.25 MHz to transmit data @ 9.6 kbps, and a minimum acceptable E_b/N_o is 10 dB. Determine the maximum number of mobile users that can be supported in a single cell CDMA system using three sectors at the cell-site and voice activity detection factor, $G_v = 0.75$. Use typical values for other factors.

Self-Assessment Exercise linked to LO 6.3

Q6.3.1 If the PN sequences used for spreading and despreading the baseband signal are not orthogonal, is CDMA still possible?

Q6.3.2 Define intracell interference and intercell interference. How do these affect multiuser wireless communications?

Q6.3.3 Describe the purpose of code acquisition and tracking in spread-spectrum systems. List important factors that make initial code acquisition to be very difficult process.

Q6.3.4 Give a brief account of advantages and disadvantages of RAKE receiver concept used in CDMA phone receiver.

Q6.3.5 Distinguish between open-loop and closed-loop power control mechanisms employed in DS-CDMA multiuser communications.

Q6.3.6 Define jamming margin. Specify the significance of jamming margin equal to 18 dB.

Q6.3.7 What is meant by voice activity? How can it contribute in increasing the system capacity of DS-CDMA multiuser system?

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/157

Q6.3.8 A total of 36 equal-power mobile terminals share a frequency band through a CDMA system. Each mobile terminal transmits information at 9.6 kbps with a DSSS BPSK modulated signal which has E_b/N_o of 6.8 dB corresponding to bit-error probability of 0.001. Analyze the minimum chip rate of the spreading PN code in order to maintain the specified E_b/N_o value. Assume the interference factor from neighboring cells = 60%; the voice activity factor = 50%; the power control accuracy factor $\alpha = 0.8$.

Q6.3.9 In an omnidirectional (single-cell, single-sector) CDMA cellular system, $E_b/N_o = 20$ dB is required for each user. If 100 users, each with a baseband data rate of 13 kbps, are to be accommodated. Calculate the minimum channel bit rate of the spread-spectrum chip sequence. Consider voice activity factor as 40%.

Q6.3.9 For DS-CDMA multiuser system, a chip rate of 1.2288 Mcps is specified for the data rate of 9.6 kbps. E_b/N_o is taken as 6.8 dB. Estimate the average number of mobile users that can be supported by the system in a sector of the 3-sector cell. Assume the interference from neighboring cells = 50%; the voice activity factor = 60%; the power control accuracy factor $\alpha = 0.85$; and the improvement factor from sectorization = 2.55.

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 6.3: *Implement DSSS in code division multiple access (CDMA) multiuser communication systems.*

LO 6.4

6.4 MULTIPLE ACCESS TECHNIQUES

Recall

The fundamental physical resource in wireless communications is the available radio frequency (RF) spectrum. In a radio communication system, only a finite amount of radio spectrum (or number of channels) is available to provide simultaneous communication links to many users in a given service area.

Objectives of Multiple Access

The chief objective of multiple access techniques is to maximize the spectrum utilization. *Multiple access techniques* refers to sharing the available limited spectrum among many users simultaneously while maintaining the desired quality of communications, to achieve high user capacity. The multiple access techniques describe the general approach to sharing the physical resources of a wireless medium by a large number of mobile users at the same time in different locations. Multiple access techniques are used to achieve high user capacity by sharing the available limited spectrum among many users simultaneously, while maintaining the desired quality of communications. The choice of an access method will have a great impact on the capacity and quality of service provided by a wireless network. In practice, most wireless communication systems are a combination of different multiple access strategies.

The concept of Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA), Code Division Multiple Access (CDMA), and orthogonal FDMA techniques.

What We Discuss Here .....

6.4.1 Frequency Division Multiple Access (FDMA)

Frequency Division Multiple Access (FDMA) technique refers to sharing the available radio spectrum by assigning specific frequency channels to users on permanent or temporary basis. The base station dynamically assigns a different carrier frequency to each active mobile user for transmission. The available radio spectrum is divided into a set of continuous frequency channels labeled 1 through N. The frequency channels are assigned to individual mobile users on a continuous time basis for the duration of a call.

Define

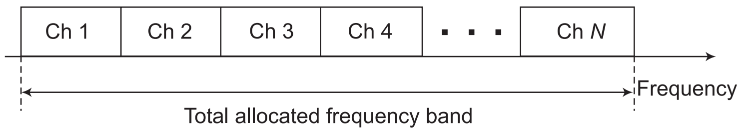


Figure 6.4.1 *Basic Concept of FDMA*

A duplex spacing is used between the forward channels (downlink–base station to user transmission) and reverse channels (uplink–user to base station transmission). The combination of forward and reverse channels is known as *frequency division duplexing* (FDD).

Frequency Division Duplexing (FDD)

The structure of forward and reverse channels in FDMA is shown in Figure 6.4.2.

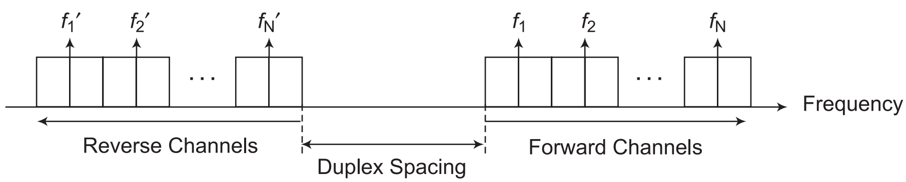


Figure 6.4.2 The Channel Structure in FDMA

In both forward and reverse channels, the signal transmitted must be kept confined within its assigned channel bandwidth, and the out-of-band signal energy causes negligible interference to the users using adjacent channels.

Practical Considerations

- In order to minimize adjacent channel interference, two design measures are usually considered.
  - First, the power spectral density of the modulated signal is controlled so that the power radiated into the adjacent band is at least 60 to 80 dB below that in the desired band. This requirement can be achieved with the use of highly selective filters in the system design.
  - Second, usually it is extremely difficult to achieve the desired filter characteristic so as not to cause adjacent channel interference.
- A *guard band* B_g is used to minimize adjacent channel interference between two adjacent channels. In other words, guard bands are inserted as buffer frequency zones in adjacent channels.
- To ensure acceptable signal quality performance, it is important that each frequency channel signal be kept confined to the assigned channel bandwidth. Otherwise there may be adjacent channel interference which can degrade signal quality.

IMPORTANT! The FDMA channel carries only one dedicated communication link (forward and reverse channel) at a time. After the assignment of a voice channel, the base station and the mobile user transmit simultaneously and continuously.

Figure 6.4.3 shows the individual FDMA channels with guard band.

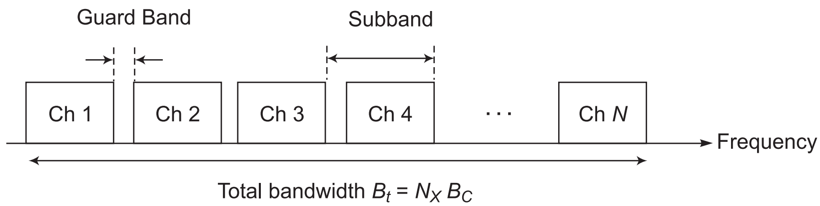


Figure 6.4.3 Guard Bands in FDMA Channels

Expression for Number of Channels in FDMA

The frequency bandwidth allocated to each mobile user is called the sub-band B_c. If there are N channels in a FDMA system, the total bandwidth B_t is equal to $N \times B_c$. In other words, each user is assigned only a fraction of the channel bandwidth, that is, $B_c = B_t/N$. Each user accesses the assigned channel on a continuous-time basis. Hence, the number of channels N that can be simultaneously supported in a FDMA system is given by

$$N = \frac{B_t - 2B_g}{B_c}$$

where B_t is the total spectrum allocation, B_g is the guard band allocated at the edge of the allocated spectrum band, and B_c is the channel bandwidth. B_t and B_c may be specified in terms of simplex bandwidths where it is understood that there are symmetric frequency allocations for the forward band and reverse band.

Good to Know!

FDMA has been widely adopted in all *first-generation analog cellular systems* for handheld and vehicle-installed mobile users. The FDMA channel carries only one dedicated communication link (forward and reverse channel) at a time. After the assignment of a voice channel, the base station and the mobile user transmit simultaneously and continuously. The disadvantage of FDMA is obvious that if the assigned channel is not in use then it remains idle and cannot be used by other mobile users. This is clearly wastage of spectrum resource. However, the utilization of channel during free time is essential to increase system capacity.

Application

The first-generation analog cellular communication systems used FDMA technique, with speech signals being transmitted over the forward or reverse channels using frequency modulation scheme. Cable television channels are transmitted using FDMA over coaxial cable. Each analog television signal utilizes 6 MHz of the 500 MHz bandwidth of the cable.

SOLVED EXAMPLE 6.4.1 Number of Channels in FDMA System

A US AMPS analog cellular system is allocated 12.5 MHz for each simplex band. If the guard band at either end of the allocated spectrum is 10 kHz, and the channel bandwidth is 30 kHz, find the number of channels available in an FDMA system.

Solution Allocated spectrum, $B_t = 12.5$ MHz (Given)

Allocated guard band, $B_g = 10$ kHz (Given)

Channel bandwidth, $B_c = 30$ kHz (Given)

The number of channels available in the FDMA system is given as

$$N = \frac{B_t - 2B_g}{B_c}$$

$$\Rightarrow \quad N = \frac{12.5 \times 10^6 - 2 \times 10 \times 10^3}{30 \times 10^3} = 416 \qquad \textbf{Ans.}$$

SOLVED EXAMPLE 6.4.2 **Number of Simultaneous Links in FDMA System**

A cellular system operator is allocated total spectrum of 5 MHz for deployment of Analog Cellular System based on FDMA technique, with each simplex channel occupying 25 kHz bandwidth. Compute the number of simultaneous calls possible in the system.

Solution Total spectrum allocated = 5 MHz (Given)

Channel bandwidth = 25 kHz (Given)

Number of simplex channels = Total spectrum allocated/Channel bandwidth

Number of simplex channels = 5 MHz / 25 kHz = 200

Number of simplex channels in a duplex channel = 2

Therefore, number of duplex channels = 200/2 = 100

Hence, in a given analog cellular FDMA system, 100 full-duplex communication links can be established simultaneously as each link requires two simplex channels (one for uplink and another for down link) or one duplex channel.

Therefore, the number of simultaneous calls = **100 calls** **Ans.**

SOLVED EXAMPLE 6.4.3 **FDMA/FDD in 1G Cellular System**

Illustrate the concept of FDMA/FDD system commonly used in First Generation (1G) analog cellular communications standard.

Solution In FDMA/FDD systems, forward and reverse channels use different carrier frequencies, and a fixed sub-channel pair is assigned to a user during the communication session. Figure 6.4.4 shows the FDMA/FDD based system commonly used in first-generation analog cellular systems.

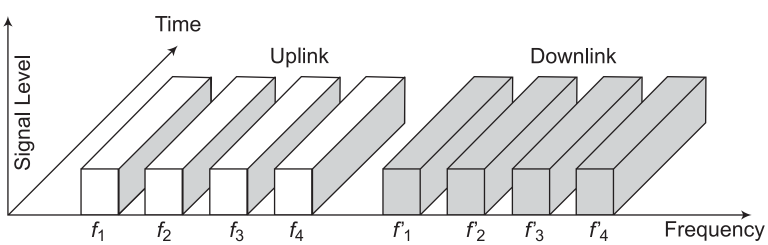


Figure 6.4.4 The Concept of FDMA/FDD

At the receiving end, the mobile unit filters the designated channel out of the composite signal received. The 1G analog cellular standard 'Advanced Mobile Phone System (AMPS)' is based on FDMA/FDD. As shown in Figure 6.4.5, the AMPS system allocates 30 kHz of channel bandwidth for each uplink (824 MHz – 849 MHz) and downlink (869 MHz – 894 MHz) frequency band.

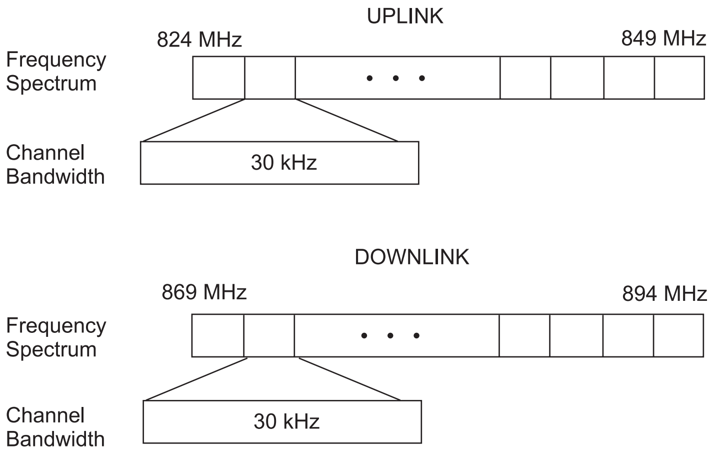


Figure 6.4.5 FDMA/FDD in AMPS

Some of the salient features of FDMA/FDD system concept are the following:

- During the call, a mobile user occupies two simplex channels, one each on the uplink and downlink, for full duplex communication.
- The two simplex channels are spaced by fixed duplex spacing. For example, duplex spacing in AMPS is (869 MHz – 824 MHz =) 45 MHz.
- When a call is terminated, or when hand-off occurs, the occupied channels are released which can be used by other mobile users in the system.
- Multiple or simultaneous mobile users are accommodated in AMPS by allocating each calling or called mobile user a dedicated channel.
- Voice signals are sent on the forward channel from the base station to mobile user, and on the reverse channel from the mobile user to the base station.
- In AMPS, analog narrowband frequency modulation technique is used to modulate the carrier.

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEM

Ex 6.4.4 If 20 MHz of total spectrum is allocated for a duplex wireless cellular system and each duplex channel has 25 kHz RF bandwidth, find the number of duplex channels. Assume the system uses FDMA technique.

6.4.2 Time Division Multiple Access (TDMA)

Define

Time division multiple access (TDMA) technique refers to allowing a number of users to access a specified channel bandwidth on a time-shared basis. In each time slot, only one user is allowed to either transmit or receive. A number of users share the same frequency band by transmitting or receiving in their assigned time slot. Let there are *N* number of time slots in a TDMA frame. Each user occupies a repeating time slot which reoccurs in every frame periodically. Figure 6.4.6 depicts the splitting of a single carrier channel into several time slots and distribution of time slots among multiple users.

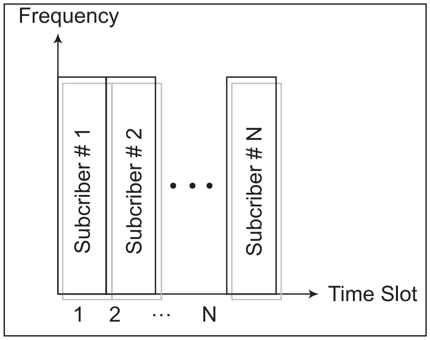


Figure 6.4.6 Basic Concept of TDMA

TDMA/FDD Mode

Usually, a TDMA system operate in TDMA/FDD mode, i.e., the forward and reverse channel frequencies differ. This means that the carrier frequencies are different but frame structures are same for the forward and reverse channels. In general, TDMA/FDD systems intentionally induce delay of several time slots between the forward and reverse time slots for a particular user. This avoids the need of duplexers in the user unit.

Time Slots and Frames

In TDMA, a carrier channel is divided into *N* time slots. These time slots are allocated for each user to transmit and receive information. The number of distinct consecutive time slots is called a *frame* and these time slots are repeated. Each frame of the TDMA structure contains *N* number of time slots of equal duration. Information data is transferred and received in the form of TDMA frames. The transmission rate for a digital TDMA channel is typically *N* times higher than that required for a single channel.

The illustration of forward and reverse channels in a TDMA/FDD system employing the similar frame and time slot structure is given in Figure 6.4.7.

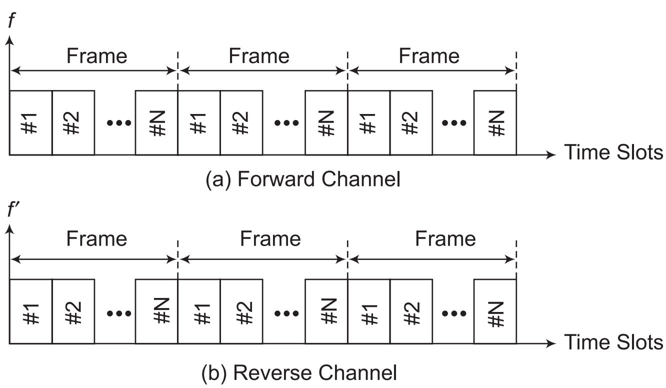


Figure 6.4.7 Structure of TDMA/FDD System

Bitwise Structure of Time Slot

The bitwise structure of each time slot is different in different types of TDMA systems. Typically, the bits contained in each time slot of a TDMA frame are divided into two major functional groups: Signaling and control data bits, and traffic data bits.

- Signaling and control data bits perform the functions which assist the receiver in performing some auxiliary functions such as synchronization and frame error rate.

- Specifically, the synchronization bits in a time slot enable the receiver to recover sinusoidal carrier essential for coherent detection.
- Traffic data bits represent digitized speech bits or any other forms of information-bearing data bits.
- The guard bits or guard time between the time slots helps in minimizing the interference due to propagation delays along different radio paths in the wireless channel.

Figure 6.4.8 shows a TDMA frame consists of a preamble, an information data field, and tail bits. The information data field of a frame consists of number of time slots.

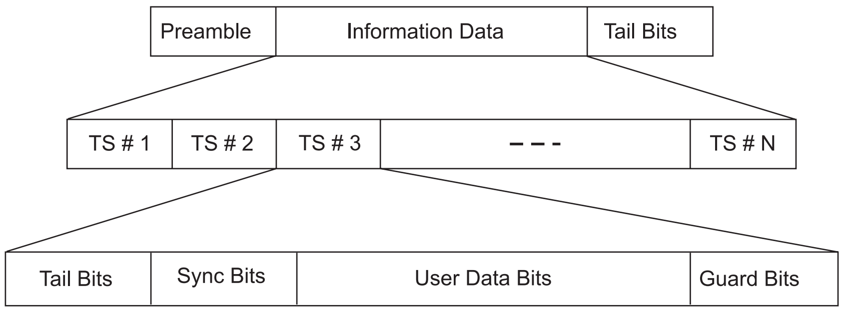


Figure 6.4.8 Typical Frame Structure and Time Slot of TDMA

Expression for Number of Channels in TDMA

In a TDMA system, the communication channels essentially consist of many time slots, which make it possible for one frequency carrier channel to be efficiently utilized by many users. Each user has access to the total bandwidth B_t of the carrier channel. Each user accesses the channel for only a fraction of the time that it is in use and on a periodic regular and orderly basis. The overall channel transmission data rate is N times the user's required data rate. The total number of TDMA time slots is determined by multiplying the number of time slots per carrier channel by the number of channels available. It is given by

$$\boxed{N = \frac{m \times (B_t - 2B_g)}{B_c}}$$

where m is the number of time slots per carrier channel, B_t is the total allocated spectrum bandwidth in Hz, B_c is the carrier channel bandwidth in Hz, and B_g is the guard bandwidth in Hz.

Two guard bands, one at the lower end and another at the higher end of the allocated frequency spectrum, are required to ensure that users operating at the edges of the allocated frequency band do not interfere with other systems operating in adjacent frequency band.

Frame Efficiency

The *frame efficiency* of a TDMA system is defined as the number of bits representing digitized speech, expressed as a percentage of the total number of bits including the control overhead bits that are transmitted in a frame. For example, in a TDMA cellular system based on IS-136 cellular standards, the forward channel contains 260 traffic data bits out of total 322 bits in a TDMA frame (ignoring 2 bits used as reserved bits). The frame efficiency in this case is $(260/322 \times 100) = 80.7\%$.

Salient Features of TDMA

The major feature of the TDMA is the flexibility of its digital format which can be buffered and multiplexed efficiently, and assignments of time-slots among multiple users which are readily adaptable to provide different access rates. With TDMA, a cell-site controller assigns time slots to users for the requested service, and an assigned time slot is held by a user until it releases it. The receiver synchronizes to the incoming TDMA signal frame, and extracts the time slot designated for that user. Therefore, the most critical feature of TDMA operation is *time synchronization*.

Application

Digital data encoding and digital modulation schemes are used with TDMA. The transmission from various users is interlaced into a uniformly repeating TDMA frame structure. Various TDMA based cellular standards such as USDC (United States Digital Cellular), GSM (Global System for Mobile Communication) have different TDMA frame structures.

SOLVED EXAMPLE 6.4.5 Number of Simultaneous Users in TDMA System

Consider a TDMA/FDD based GSM system, which uses 25 MHz band for the forward link and is divided into RF channels of 200 kHz each. If 8 time slots are supported on a single RF channel, find the number of simultaneous users that can be accommodated in GSM, assuming no guard band.

Solution The allocated spectrum, B_t = 25 MHz = 25×10^6 Hz (Given)

The channel bandwidth, B_c = 200 kHz = 200×10^3 Hz (Given)

Number of time slots, m = 8 per RF channel

The guard bandwidth, $B_g = 0$ (specified in the problem statement)

Hence, the number of simultaneous users that can be accommodated in GSM system is given as

$$N = \frac{m \times (B_t - 2B_g)}{B_c} = \frac{8 \times (25 \times 10^6 - 2 \times 0)}{200 \times 10^3} = 1000 \text{ users}$$

Hence, GSM system can accommodate 1000 simultaneous users. **Ans.**

SOLVED EXAMPLE 6.4.6 Frame Efficiency of TDMA-based GSM System

The basic TDMA frame structure of GSM cellular system comprises of 156.25 bits in a time slot, of which 40.25 bits are overhead (ignoring the 2 flag bits), compute the frame efficiency.

Solution Total bits in a TDMA frame = 156.25 bits (Given)

Number of overhead bits = 40.25 bits (Given)

The frame efficiency = [1 – (overhead bits / total bits)] × 100

The frame efficiency = [1 – (40.25/156.25)] × 100

Hence, the frame efficiency = **74.2 %** **Ans.**

YOU ARE NOW READY TO ATTEMPT THE FOLLOWING PROBLEMS

Ex 6.4.7 Consider a digital cellular system based on TDMA/FDD scheme that is allocated 5 MHz band each for the uplink and downlink. The system used digital modulation technique which requires 200 kHz channel bandwidth. If 8 time slots are supported on a single radio channel, find the number of simultaneous communication links that can be served by the system, assuming 10 kHz guard band on each end of the allocated spectrum.

Ex 6.4.8 A normal TDMA-GSM time slot consists of six trailing bits, 8.25 guard bits, 26 training bits, and two traffic bursts of 58 bits of data. Compute the frame efficiency.

6.4.3 Code Division Multiple Access (CDMA)

Define

The *code division multiple access (CDMA)* technique refers to a multiple-access technique in which the individual users occupy the complete frequency spectrum whenever they transmit and different spread-spectrum codes are generated by PN code generator and assigned to each user. This implies that multiple users share the same frequency.

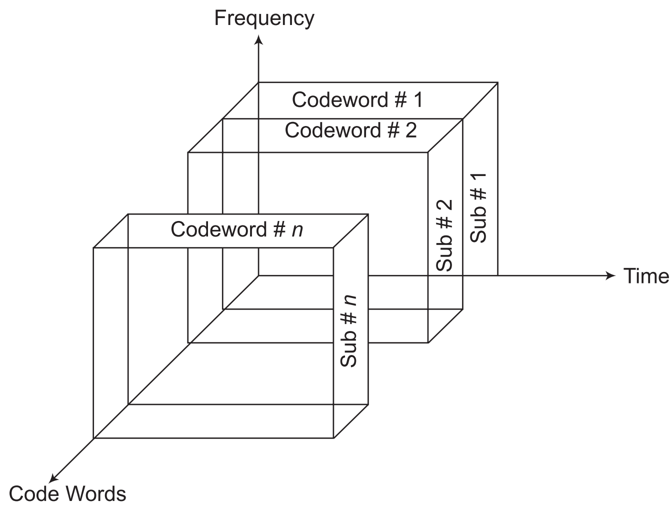


Figure 6.4.9 Basic Concept of CDMA

CDMA and Spread-spectrum

A CDMA system is based on spectrum-spread technology by spreading the bandwidth of modulated signal substantially, which makes it less susceptible to the noise and interference. A CDMA system is usually quantified by the chip rate of the *orthogonal PN codes*, which is defined as the number of bits changed per second.

The orthogonality of the PN codes enables simultaneous data transmission from many mobile users using the complete frequency band assigned for a cell site. Each mobile receiver is provided the corresponding PN code so that it can decode the data it is expected to receive. The encoding in the transmitter and the corresponding decoding at the receiver make the system design robust but quite complex.

CDMA/FDD Concept

Consider that the available bandwidth and time as resources needed to be shared among multiple mobile users. In a CDMA environment, multiple users use the same frequency band at the same time, and the user is distinguished by a unique code that acts as the key to identify that user. These unique codes are selected so that when they are used at the same time in the same frequency band, a receiver can detect that user among all the received signals with the help of known code of that user.

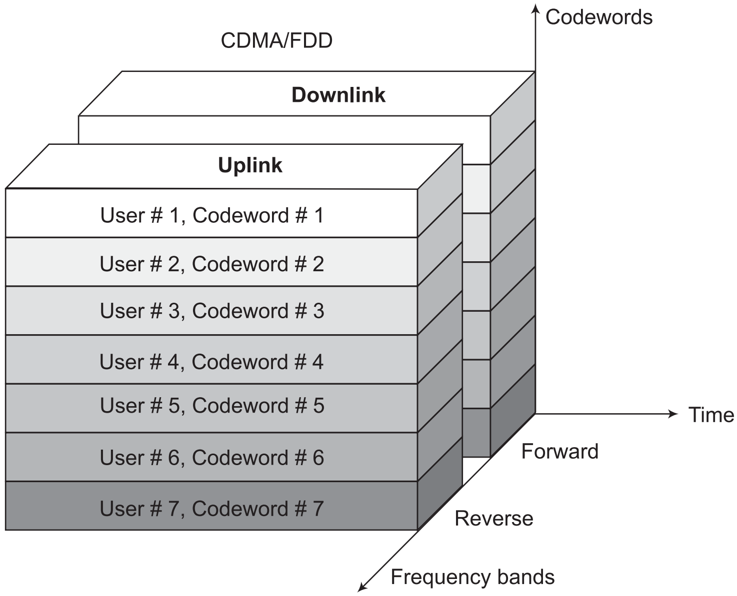


Figure 6.4.10 CDMA/FDD Concept

Application

Some second-generation digital cellular systems such as IS-95 and most of the third-generation IMT-2000 digital cellular systems use CDMA/FDD technique.

Table 6.4.1 shows the different multiple access techniques being used in various analog and digital cellular communications systems.

Table 6.4.1 Multiple Access Techniques in Cellular Systems

S. No.	Type of Cellular System	Standard	Multiplexing Technique	Multiple Access Technique
1.	1G Analog Cellular	AMPS	FDD	FDMA
2.	US Digital Cellular	USDC	FDD	TDMA
3.	2G Digital Cellular	GSM	FDD	TDMA
4.	Pacific Digital Cellular	PDC	FDD	TDMA
5.	US Narrowband Spread-spectrum Digital Cellular	IS-95	FDD	CDMA
6.	3G Digital Cellular	W-CDMA	FDD	CDMA
7.	3G Digital Cellular	Cdma2000	FDD	CDMA

Table 6.4.2 gives comparative study of FDMA, TDMA, and CDMA techniques.

Table 6.4.2 Comparison of FDMA, TDMA, and CDMA

S. No.	Parameter	FDMA	TDMA	CDMA
1.	*Basic concept*	Divides the allocated frequency band into a number of sub-bands	Divides the time into non-overlapping time slots	Spreads the signal with orthogonal codes
2.	*Signal separation*	Frequency filtering	Time synchronization	PN codes
3.	*Bandwidth efficiency*	Higher for single-cell system	higher for single-cell system	higher for multiple-cells
4.	*Diversity*	By using multiple receivers	By using multiple receivers for TDMA/ FDMA	By using RAKE receiver
5.	*Forward Error-Correction (FEC) coding*	Requires higher bandwidth	Requires higher bandwidth	Does not require more bandwidth
6.	*Modulation type*	FM and FSK	Higher order PSK	BPSK or QPSK
7.	*Active mobile users*	On their assigned frequency channels	In their specified time slot on the same frequency	On the same frequency
8.	*Hand-off or handover*	Hard-decision handoff algorithms	Hard-decision hand-off algorithms	Soft handover capability

OFDMA

FDMA is simple and robust, and has been widely used in analog cellular systems. TDMA and CDMA are flexible and used in 2G/2.5G/3G digital cellular systems.

6.4.4 Multi-Carrier Multiple Access

OFDMA—A Multi-Carrier Multiple Access Technique

Orthogonal frequency division multiple access (OFDMA) is one of multi-carrier multiple access techniques which use multiple carrier signals at different frequencies, sending some of the bits on each channel. All subchannels are dedicated to a single data source. In OFDMA technique, data is distributed over multiple carriers at precise frequencies. The precise relationship among the subcarriers is referred to as *orthogonality*. The peaks of the power spectral density of each subcarrier occur at a point at which the power of other subcarriers is zero. The subcarriers can be packed tightly together because there is minimal interference between adjacent subcarriers.

Figure 6.4.11 illustrates the basic concept of OFDMA.

Functional Description

A data stream R bps is split by a serial-to-parallel converter into N substreams. Each substream has a data rate of R/N bps. It is transmitted on a separate subcarrier with a spacing between adjacent subcarriers of f_b. The base frequency, f_b is the lowest-frequency subcarrier. All of the other subcarriers are integer multiples of the base frequency, namely $2f_b$, $3f_b$, and so on.

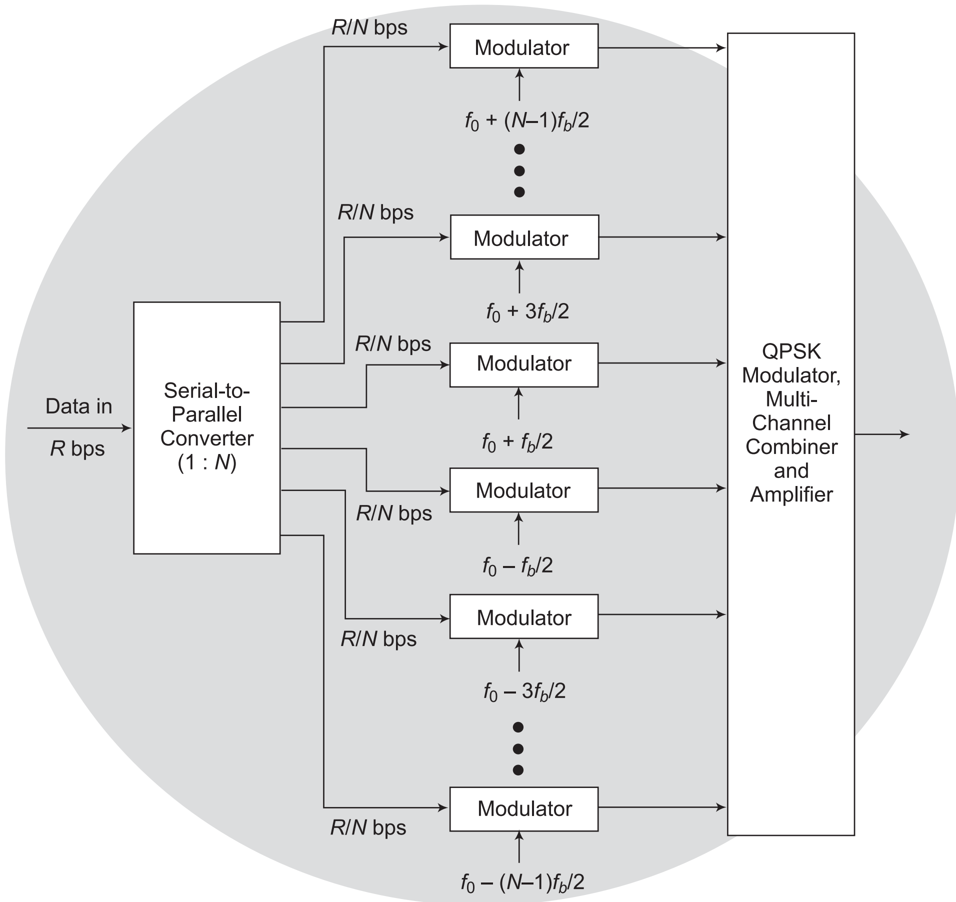


Figure 6.4.11 Basic Concept of OFDMA

The set of OFDMA subcarriers is further modulated using digital modulation scheme QPSK to a higher frequency band. Multiple access in OFDMA is achieved by assigning subsets of subcarriers to individual users, thus allowing simultaneous low data rate transmission from several users.

Multi-Carrier CDMA

The *Multi-Carrier Code Division Multiple Access (MC-CDMA)* scheme is a combination of OFDMA and DS-CDMA. MC-CDMA maintains the original signaling interval while it spreads the signal over wide bandwidth like DS-CDMA. As MC-CDMA spreads an information bit over many subcarriers, it can make use of information contained in some subcarriers to recover the original symbol. MC-CDMA gathers nearly all the scattered powers effectively using cyclic prefix insertion technique. As the received signals are sampled at the original symbol rate in MC-CDMA, the sampling points may not be optimum.

Figure 6.4.12 depicts the pictorial representation of relationship among SC-FDMA, OFDMA, and DS-CDMA/FDE.

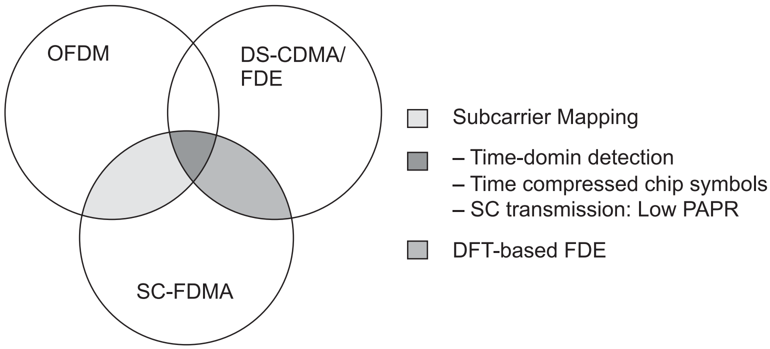


Figure 6.4.12 Relationship among SC-FDMA, OFDMA, and DS-CDMA/FDE

Salient Features of MC-CDMA

In general, the performance of MC-CDMA is equivalent to *m*-finger rake receiver in DS-CDMA, where *m* is the number of symbols in cyclic prefix of MC-CDMA. Multi-carrier direct sequence code division multiple access (MC-DS-CDMA) scheme is a combination of time-domain spreading and OFDMA. MC-CDMA is a combination of frequency-domain spreading and OFDMA. In MC-CDMA, a good bit error rate (BER) performance can be achieved by using frequency-domain equalization (FDE) since the frequency diversity gain is obtained. MC-DS-CDMA can also obtain the frequency diversity gain by applying FDE to a block of a number of OFDMA symbols. Equalization can be done in the frequency domain using discrete Fourier transform (DFT).

Benefits of Spread-spectrum Communications

The effect of multipath fading as well as interference can be reduced by a factor, known as the processing gain, which is the ratio of the spreaded bandwidth to the original information bandwidth.

- The spectral density of the DSSS transmitted signal is reduced by a factor equal to the processing gain.
- Under ideal conditions, there is no difference in bit error rate performance between spreaded and non-spreaded forms of BPSK or QPSK digital modulation schemes.
- Through proper receiver design, multipath can be used to advantage to improve receiver performance by capturing the energy in paths having different transmission delays.
- By using RAKE receiver concept, a spread-spectrum receiver can obtain an important advantage in diversity in fading channels.
- The choice of spreading codes is critical to reducing multipath self-interference and multiple-access interference.
- Spread-spectrum signals can be overlaid onto frequency bands where other systems are already operating, with minimal performance impact to both systems.
- Spread-spectrum is a wideband signal that has a superior performance over conventional communication systems on frequency selective fading multipath channel.
- Spread-spectrum provides a robust and reliable transmission in urban and indoor environments where wireless transmission suffers from heavy multipath conditions.
- Cellular systems designed with CDMA spread-spectrum technology offer greater system capacity and operational flexibility.

OFDMA is considered highly suitable for broadband wireless networks IEEE 802.16 standard WiMAX. In spectrum sensing cognitive radio, OFDMA is a possible approach to filling free radio frequency bands adaptively.

Application

Self-Assessment Exercise linked to LO 6.4

Q6.4.1 Why is FDMA technique usually implemented in narrowband wireless communication systems? Give a specific reason for not employing equalization in narrowband FDMA systems.

Q6.4.2 Suggest some measures which can be adopted in FDMA cellular systems to overcome the problem of near-far interference.

Q6.4.3 Analyze the advantages of digital TDMA cellular systems over analog FDMA cellular systems. How can desired signal quality be maintained in FDMA systems?

Q6.4.4 Code division multiple access (CDMA) technique is based on direct-sequence spread-spectrum (DSSS) modulation scheme. Outline the distinct features of CDMA technique.

Q6.4.5 Evaluate the usage of multicarrier communication. How does OFDMA minimize the impact of frequency selective fading?

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/159

✓ **If you have been able to solve the above exercises then you have successfully mastered**

LO 6-4: *Outline multiple access techniques including FDMA, TDMA, CDMA and Multi-Carrier OFDMA.*

Key Concepts

- chip rate
- code division multiple access
- direct sequence spread-spectrum
- fast frequency hopping
- frequency division duplexing
- frequency division multiple access
- frequency-hopping spread-spectrum
- Gold sequence
- jamming margin
- Kasami sequence
- *m*-sequence
- multiple access interference
- multiple access technique
- multiuser communication
- narrow-band FM transmissions
- orthogonal FDMA
- probability of error
- processing gain
- pseudo-noise (PN) sequence
- signal-to-interference ratio
- slow frequency hopping
- spread-spectrum communications
- spreading code
- time division multiple access

Learning Outcomes

LO 6.1

- Spread-spectrum communication is a means of transmitting information data by occupying much wider bandwidth than necessary.
- Spread-spectrum techniques have a processing gain which allows reduced signal-to-noise, or low transmitted power levels, to deliver satisfactory performance in a hostile operating environment.
- Spreading of spectrum is achieved through the use of a pseudorandom noise (PN) code sequence at the transmitter (for spreading) as well as at the receiver (for despreading), operating in synchronization.
- The PN code is independent of the information data sequence.
- An important attribute of spread-spectrum communication is that the transmitted signal assumes a noiselike appearance.

LO 6.2

- Direct-sequence bi- (or quad-) phase shift keying and frequency-hop M-ary frequency-shift keying are the two most popular spread-spectrum techniques, both of them rely on generation of PN sequences.
- In FH/MFSK system, the PN sequence makes the carrier signal hop over a number of frequencies in a pseudo-random fashion, the spectrum being spread in a sequential manner.
- For a DS/QPSK system, the PN sequence makes the transmitted signal assume a noiselike appearance, and data are transmitted using a pair of carriers in quadrature.

LO 6.3

- In a multiuser DS-CDMA communication system, each mobile user has its own unique code which is used to spread and despread its information data.
- The DSSS form of CDMA is generated by combining each of the baseband signals to be multiplexed with a PN sequence at a much higher data rate.
- In DS-CDMA systems, the same frequency channel can be reused in the adjacent cell provided MAI is below a given threshold level necessary to meet the signal quality requirement.
- RF power control is simply the technique of controlling the mobile transmit power so as to affect the cell-site received power, and hence the overall carrier-to-interference (C/I) value.
- RAKE receivers are commonly used in DSSS receivers in CDMA cellular mobile phones which enables to provide a robust signal reception in a hostile mobile radio environment.

LO 6.4

- The efficient ways to access the limited radio spectrum by the number of potential users simultaneously in a wireless environment span through division of frequency, time, and code.
- For an efficient use of resources by multiple number of users simultaneously, the multiple access technique such as FDMA, TDMA and CDMA is used in a wireless cellular system.
- Code division multiple access technique is based on spread-spectrum in which different users transmit on the same carrier frequency but use different spreading codes.
- Orthogonal frequency division multiple access (OFDMA) is one of multicarrier multiple access schemes which use multiple carrier signals at different frequencies, sending some of the bits on each channel.

Objective-Type Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/84

6.1 The effect of spread-spectrum modulation is that the bandwidth of the spreaded signal ______________________.
(a) remains constant
(b) increases significantly
(c) increases marginally
(d) decreases

6.2 A DSSS system has a 48 Mcps code rate and 4.8 kbps information data rate. The processing gain is computed to be
(a) 4.8 dB
(b) 40 dB
(c) 48 dB
(d) 60 dB

6.3 For baseband data rate of 9.6 kbps, the user information data is spreaded by a factor of _________ to a channel chip rate of 1.2288 Mcps.
(a) 64
(b) 128
(c) 256
(d) 1024

6.4 In a frequency-hopped spread-spectrum signal,
(a) the frequency is constant in each time chip, but changes from chip to chip
(b) the frequency changes in each time chip, but constant from chip to chip
(c) the frequency changes in each time chip as well as from chip to chip
(d) the frequency remains constant in each time chip as well as from chip to chip

6.5 *Statement I*: Spread-spectrum communication systems operate in two steps specified as information data sequence modulates a PN-sequence, followed by a BPSK or QPSK modulation process.
Statement II: Spread-spectrum communication systems operate in two steps specified as direct-sequence spread-spectrum modulation, followed by frequency hopping spread-spectrum modulation.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

6.6 Code-division multiple access scheme can support many mobile users over the same communication channel, provided that _________________ exists between i^{th} and j^{th} receiver.
(a) good cross-correlation
(b) poor cross-correlation
(c) good auto-correlation
(d) poor auto-correlation

6.7 If the information signal is narrowband and the PN signal is wideband, the modulated signal will have a spectrum that is
(a) nearly the same as the narrowband information signal

(b) nearly the same as the wideband PN signal
(c) much smaller than the wideband PN signal
(d) much larger than the wideband PN signal

6.8 The number of simultaneous users that can be supported in a practical CDMA multicell system depends on the performance improvement factor given as
(a) $(G_V \times G_A) / \sigma$
(b) $(G_V \times \sigma) / G_A$
(c) $(G_A \times \sigma) / G_V$
(d) $\sigma / (G_V \times G_A)$

6.9 _________ technique allows multiple users to simultaneously occupy the same frequency spectrum at the same time.
(a) FDMA
(b) TDMA
(c) CDMA
(d) OFDMA

6.10 *Statement I:* Multiple access in OFDMA is achieved by assigning subsets of subcarriers to individual users, thus allowing simultaneous high data rate transmission from several users.
Statement II: As MC-CDMA spreads an information bit over many subcarriers, it can make use of information contained in some subcarriers to recover the original symbol.
(a) Statement I is correct; Statement II is incorrect.
(b) Statement I is incorrect; Statement II is correct.
(c) Both statements are correct.
(d) Both statements are incorrect.

Short-Answer-Type Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/51

6.1 Map spread-spectrum approach in frequency domain and time domain with the help of suitable illustrations.

6.2 Spread-spectrum techniques require PN sequences for spreading, synchronization and despreading. Identify typical problems encountered with spread-spectrum modulation?

6.3 List important characteristics of PN sequences. State the 'run property' of maximum length sequences.

6.4 What is the basic concept of frequency-hopping spread-spectrum approach? Illustrate the basic process of frequency hopping for specified sequence s_6, s_4, s_1, s_8, s_3, s_5, s_2, s_7 corresponding to frequency sequence $f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8$ respectively.

6.5 Which type of carrier-modulation technique is generally used with FHSS signals? Draw the functional block schematic of FHSS modulator.

6.6 Tabulate the merits and demerits of FHSS and DSSS techniques.

6.7 With spread-spectrum systems, interference is transformed into noise. Establish the relationship between the input and output signal-to-noise power ratios.

6.8 How does the multiuser interference increase with the increase in the number of users in DS-CDMA system? Mention the most critical system parameter in the design of an efficient multiuser DS-CDMA system. ○○●

6.9 What is the purpose of code tracking in spread-spectrum signal? How is it achieved? ○○●

6.10 Specify the significance of signal-to-interference-plus-noise ratio (SINR) in DS-CDMA cellular systems. List three system design factors that affect SINR. ○○●

6.11 "Near-far interference results into adjacent channel interference in multiuser system". Justify this statement with an appropriate example. ●●●

6.12 Describe the step-by-step simplified procedure of open-loop RF power control in DS-CDMA system. ○○●

6.13 Why does CDMA require perfect synchronization among all the subscribers? ○○●

6.14 Compare and contrast frequency division multiple access (FDMA) and frequency-hopping multiple access (FHMA) techniques. ○●●

6.15 The multi-carrier code division multiple access (MC-CDMA) scheme is a combination of OFDMA and DS-CDMA. Depict the pictorial representation of the relationship among SC-FDMA, OFDMA, and DS-CDMA/FDE. ●●●

Discussion Questions

For answers, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/155

6.1 The processing gain is given as the gain in signal-to-noise ratio (*S/N*) obtained by the use of spread-spectrum modulation technique. Represent it graphically as well as mathematically for spread-spectrum approach illustrated in frequency domain and time domain for DSSS as well as for FHSS techniques. **[LO 6.1]**

6.2 In frequency-hopping spread-spectrum (FHSS) technique, the spectrum of the transmitted signal is spreaded sequentially (pseudo-random-ordered sequence of the frequency hops) rather than instantaneously. Compare and contrast two distinct methods of frequency hopping—Fast FHSS and Slow FHSS. Give the mathematical expression for signal-to-interference ratio (SIR) per user in FHSS system. **[LO 6.2]**

6.3 There are several RF power control mechanisms that can be based on the signal strength received by the cell site or can depend on other system parameters. Discuss two main types of RF power control mechanisms—an open loop and a closed loop. Which one is preferred for a specific situation? **[LO 6.3]**

6.4 The major feature of the TDMA is the flexibility of its digital format which can be buffered and multiplexed efficiently. By specifying a basic TDMA frame, how can frame efficiency be improved? **[LO 6.4]**

Problems

For answer keys, scan the QR code given here

OR

visit http://qrcode.flipick.com/index.php/161

6.1 Two *m*-sequences are generated in two different chain of shift registers (SRs), and then these two sequences are bit-by-bit XORed. Predict the number of shift registers required to obtain a processing gain of 45.15 dB.

6.2 If the PN generator is driven by a clock signal at the rate of 10 MHz, and the feedback register has 37 stages, Solve the total length of sequence in hours.

6.3 A pseudo-random (PN) sequence is generated using a feedback-shift register with four number of memory elements. What is the PN sequence length, N?

6.4 A spread-spectrum communication system has information bit duration of 4.095 ms, and PN chip duration of 1.0 μs. Design the value of shift-register length, $N = 2^m - 1$.

6.5 A frequency-hopping spread-spectrum communication system utilizes fast-hop technique at the hop rate of 16 hops per information bit, information bit rate of 2400 bps, frequency multiplication factor of 8. Calculate the minimum of different frequencies required to be generated by the frequency synthesizer, assuming this number to be a power of 2.

6.6 A frequency-hopping spread-spectrum communication system utilizes fast-hop technique at the hop rate of 10 hops per information bit. If the information bit rate is 2500 bps, what would be the frequency separation?

6.7 Hypothesize the clock rate of PN code-generator for a frequency-hopping spread-spectrum communication system utilizing fast-hop technique at the hop rate of 10 hops per information bit, having 512 number of different frequencies, information bit rate of 2800 bps, and final RF multiplication factor of 9.

6.8 A hybrid FHSS/DSSS system uses a PN code rate of 150 kbps. The RF bandwidth of spread-spectrum signal is 256 MHz centered at 1.0 GHz. Paraphrase the total number of frequencies which the frequency synthesizer must produce for the system to cover the entire RF bandwidth.

6.9 A frequency-hopping spread-spectrum communication system utilizes fast-hop technique at the hop rate of 10 hops per information bit, total number of different frequencies of 512, information bit rate of 2800 bps, and final RF multiplication factor of 9. Determine the RF signal bandwidth of spreaded signal.

6.10 In an omnidirectional (single-cell, single-sector) CDMA cellular system, $E_b/N_o = 20$ dB is required for each user. If 100 users, each with a baseband data rate of 13 kbps, are to be accommodated. Construct the minimum channel bit rate of the spread-spectrum chip sequence. Ignore voice activity considerations.

6.11 Evaluate the maximum number of mobile users that can be supported in a single cell CDMA system using omnidirectional cell-site antenna and no voice activity detection. Assume the CDMA system is interference limited. The system uses an RF bandwidth of 1.25 MHz to transmit data @ 9.6 kbps, and a minimum acceptable E_b/N_o is found to be 10 dB. Use $\rho = 0.5$, $G_v = 0.85$, $G_A = 1$.

6.12 The CDMA system uses an RF bandwidth of 1.25 MHz to transmit data @ 9.6 kbps, and a minimum acceptable E_b/N_o is 10 dB. Monitor the maximum number of mobile users that can be supported in a single cell CDMA system using three-sectors at the cell-site and voice activity detection factor, $G_v = 0.75$. Assume use typical values for other factors.

6.13 A total of 36 equal-power mobile terminals share a frequency band through a CDMA system. Each mobile terminal transmits information at 9.6 kbps with a DSSS BPSK modulated signal which has E_b/N_o of 6.8 dB corresponding to bit error probability of 0.001. Calculate the minimum chip rate of the spreading PN code in order to maintain the specified E_b/N_o value. Assume the interference factor from neighboring cells = 60%; the voice activity factor = 50%; the power control accuracy factor $\alpha = 0.8$. ○●●

6.14 If RF spectrum of 20 MHz is allocated for a duplex wireless system and each duplex channel has 25 kHz RF bandwidth, structure the number of duplex channels. Assume the system uses FDMA technique. ●●●

6.15 A normal TDMA-GSM time slot consists of six trailing bits, 8.25 guard bits, 26 training bits, and two encoded-data bursts of 58 bits each. Review the number of bits in a time slot and in a frame containing 8 time slots. ○○●

Critical Thinking Questions

6.1 Spreading of narrow bandwidth of baseband signal is accomplished through the use of a spreading code, called pseudo-noise (PN) code. State the two most important properties and three criteria which can be used to validate that a sequence of numbers is practically random. Draw a simple arrangement to generate PN sequences and explain the procedure for the same. **[LO 6.1]**

6.2 The most practical system applications employing direct sequence spread-spectrum (DSSS) techniques use digital carrier modulation formats such as BPSK and QPSK. Why is it necessary to employ coherent detection scheme at the DSSS receiver? Discuss various aspects of DSSS demodulator with coherent BPSK detection with reference to suitable functional block schematic diagram. **[LO 6.2]**

6.3 In the practical design of DS-CDMA digital cellular systems, there are three system parameters that affect the capacity as well as the bandwidth efficiency of the system. Recite them and write an expression for approximating the number of simultaneous users that can be supported in a practical multicell CDMA system, incorporating these three performance improving factors. **[LO 6.3]**

6.4 In orthogonal frequency division multiple access (OFDMA) technique, data is distributed over multiple carriers at precise frequencies. Illustrate the basic concept of OFDMA that uses QPSK modulator, multi-channel combiner and amplifier and describe its application in multi-carrier CDMA. **[LO 6.4]**

References for Further Reading

[GA00] Garg VK; *IS-95 CDMA and cdma2000.* Prentice Hall, Upper Saddle River, NJ, 2000.

[KI00] Kim KI; *Handbook of CDMA System Design, Engineering, and Optimization.* Prentice Hall, Upper Saddle River, NJ, 2000.

[SI10] Singal, TL; *Wireless Communications.* Tata McGraw-Hill, 2010.

[SI12] Singal, TL; *Analog and Digital Communications.* Tata McGraw-Hill, 2012.

APPENDIX A

Random Signal Theory—An Overview

A.1 BASIC CONCEPT OF PROBABILITY THEORY

The probability theory can be modeled by an experiment, repeated over a large number of times, with an unpredictable outcome which exhibit statistical averages. The probability of an event is intended to represent the likelihood that an outcome will result in the occurrence of that event. The concept of probability occurs naturally because the possible outcomes of most of the experiments are not always the same. Thus, the possibility of occurrence of an event may be defined as the ratio of the number of possible favorable outcomes to the total number of possible equally likely outcomes. That is, $0 \leq P \leq 1$ ($P = 0$ means an event is not possible while $P = 1$ means that event is certain).

A.2 RANDOM VARIABLES

The term '*random variable*' is used to signify the functional relationship by which a real number is assigned to each possible outcome of an event. Let a random variable X represent the functional relationship between a random event and a real number. The random variable X may be discrete or continuous.

- If a random variable X assumes only a finite number of discrete values in any finite interval, then it is called *discrete random variable*.
- If a random variable X can assume any value within an interval, then it is called *continuous random variable*.

The *cumulative distribution function*, denoted by $F_x(x)$, of the random variable X is given by $F_x(x) = P(X \leq x)$; where $P(X \leq x)$ is the probability that the value of the random variable X is less than or equal to a real number x. The cumulative distribution function is a probability, i.e., $0 \leq F_x(x) \leq 1$. It may include all possible events, i.e., $F_x(+\infty) = 1$ and no possible events, i.e., $F_x(+\infty) = 0$. For $x_1 \leq x_2$; $F_x(x_1) \leq F_x(x_2)$.

The *probability density function* (pdf), of the random variable X is simply the derivative of the cumulative distribution function $F_x(x)$ as $\boxed{p_x(x) = \frac{d}{dx}[F_x(x)]}$. The *pdf* is a function of a real number x. The probability that the outcome lies in the range $x_1 \leq X \leq x_2$ for the case where $f(x)$ has no impulses can be written as $P(x_1 \leq X \leq x_2) = \int_{x_1}^{x_2} f(x)dx$, i.e., $P(x_1 \leq X \leq x + dx) = f(x)dx$. As x increases, $F_x(x)$ increases monotonically (i.e., more outcomes are included in the probability of occurrence), and therefore, $p_x(x) \geq 0$ for all values of x. Also, $P(x_1 \leq X \leq x_2) = \int_{x_1}^{x_2} f(x)dx$. A pdf is always a non-negative function with a total area of unity, i.e., $\int_{-\infty}^{\infty} p_x(x)dx = F_x(+\infty) - F_x(-\infty) = 1 - 0 = 1$. The probability that a random variable X has a value in very narrow range from x to $x + \Delta x$ can be approximated as $P(x \le X \le x + \Delta x) = p_x(x)\,\Delta x$. As $\Delta x \to 0$, $P(X = x) = p_x(x)dx$. And

$$P(x_1 \le X \le x_2) = P(X \le x_2) - P(X \le x_1)$$

$$\Rightarrow \qquad P(x \le X \le x_2) = F_x(x_1)$$

$$\Rightarrow \qquad P(x_1 \le X \le x_2) = \int_{x_1}^{x_2} p_x(x)dx$$

The concept a of single random variable can easily be extended to two or more random variables, either by all discrete or all continuous, or a mix of discrete and continuous.

- If X and Y are two *discrete random variables* then the joint probability density function of X and Y is given by $p(x, y) = P(X = x, Y = y)$, where $p(x, y) \ge 0$, and $\sum_x \sum_y p(x,y) = 1$.
- If X and Y are two *continuous random variables* then the joint probability density function of X and Y is given by $p(x, y) = P(X = x, Y = y)$, where $p(x, y) \ge 0$, and $\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p(x,y)dxdy = 1$.
- For two *independent random variables* X and Y, the relationship between joint cumulative distribution and probability density function can be expressed as

$$P(x_1 \le X \le x_2, y_1 \le Y \le y_2) = \left[\int_{x_1}^{x_2} f(x)dx\right]\left[\int_{y_1}^{y_2} f(y)dy\right]$$

The mean, variance, and higher order averages of a random variable can be mathematically expressed as *moments of a random variable*. Consider a random variable X, characterized by its probability density function $p(X)$. The first moment of a random variable X, denoted by $E\{X\}$, is defined as $\boxed{E\{X\} = \int_{-\infty}^{\infty} xp_x(x)dx}$. This is the *mean or expected value* of X. Similarly, the second moment of the random variable X, is defined as $E\{X^2\} = \int_{-\infty}^{\infty} x^2 p_x(x)dx$. It is also known as *mean square value*. In general, the average of X^n can be represented by the n^{th} moment of the random variable X, given by $E\{X^n\} = \int_{-\infty}^{\infty} x^n p_x(x)dx$. The *central moments* of a probability distribution of a random variable X are the moments of the difference between X and m_x. The second central moment is called the *variance* of X, as $\sigma_x{}^2 = E\{(X - m_x)^2\} = \int_{-\infty}^{\infty} (x - m_x)^2 p_x(x)dx$. The *variance* is a measure of the randomness of the random variable X, thereby constraining the width of its probability density function. The variance is related to the mean-square value as $\boxed{\sigma_x{}^2 = E\{X^2\} - m_x{}^2}$. The square root of variance of X is called the *standard deviation*, denoted by σ_x.

Mean of a Sum of Random Variables

Let X, Y, and Z be random variables with respective means m_x, m_y, and m_z. Let Z be the sum of two independent or dependent random variables X and Y, that is $Z = X + Y$. Then, by definitions of mean values, we know that

$$m_x = \int_{x=-\infty}^{\infty} \int_{y=-\infty}^{\infty} x[f(x, y)]dxdy$$

$$m_y = \int_{x=-\infty}^{\infty} \int_{y=-\infty}^{\infty} y[f(x, y)]dxdy$$

$$m_z = \int_{x=-\infty}^{\infty} \int_{y=-\infty}^{\infty} (x+y)[f(x, y)]dxdy$$

$$\Rightarrow \qquad m_z = \int_{x=-\infty}^{\infty} \int_{y=-\infty}^{\infty} x[f(x, y)]dxdy + \int_{x=-\infty}^{\infty} \int_{y=-\infty}^{\infty} y[f(x, y)]dxdy$$

$$\Rightarrow \qquad m_z = m_x + m_y$$

Hence, the mean of the sum of random variables is equal to the sum of the means of the individual random variables.

Variance of a Sum of Random Variables

Let X and Y be two independent random variables, and Z is sum of these two independent random variables, that is $Z = X + Y$. Then, the square root of the average value, also called the second moment, or the variance, or the root mean square (rms) value of Z is defined as $\overline{Z^2} = \overline{(X+Y)^2}$.

$$\Rightarrow \qquad \overline{Z^2} = \int_{x=-\infty}^{\infty} \int_{y=-\infty}^{\infty} (x+y)^2[f(x, y)]dxdy$$

$$\Rightarrow \qquad \overline{Z^2} = \int_{x=-\infty}^{\infty} x^2 f(x)dx \int_{y=-\infty}^{\infty} f(y)dy + \int_{y=-\infty}^{\infty} y^2 f(y)dy \int_{x=-\infty}^{\infty} f(x)dx + 2\int_{x=-\infty}^{\infty} xf(x)dx \int_{y=-\infty}^{\infty} yf(y)dy$$

But $\displaystyle\int_{x=-\infty}^{\infty} f(x)dx = \int_{y=-\infty}^{\infty} f(y)dy = 1$ (by probability definitions)

$$\therefore \qquad \overline{Z^2} = \overline{X^2} + \overline{Y^2} + 2\overline{XY}$$

If either $\overline{X} = 0$, or $\overline{Y} = 0$, or both $\overline{X}$ and $\overline{Y}$ are zero then $\overline{Z^2} = \overline{X^2} + \overline{Y^2}$

Since $\sigma_z{}^2 = E\left[X^2\right]$ or $\overline{X^2}$, $\sigma_z{}^2 = \sigma_x{}^2 + \sigma_y{}^2$

Hence, the variance of the sum of random variables is equal to the sum of the variance of the individual random variables.

A.3 CONVOLUTION AND CORRELATION

Convolution of two time-domain signals is equivalent to multiplying the frequency spectra of those two signals. The process of convolution can be described in simple steps as firstly one signal is flipped back to front, then the flipped signal is shifted in time. The extent of the shift is the position of convolution function point to be calculated. Each element of one signal is then multiplied by the correspondent element of the other signal. The area under the resulting curve is integrated.

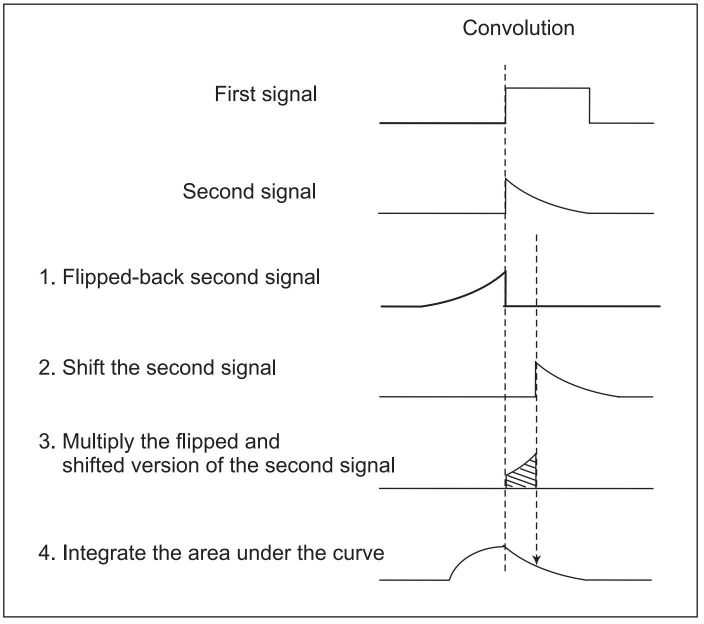


Fig. A.1 Process of Convolution Function

Convolution is used to find the output of the system for a given input, although it requires a lot of computations. It can also be used to smooth a noisy sinusoidal signal by convolving it with a rectangular function. The smoothing property can be considered equivalent to digital filtering. Sending the signal over a channel is a convolution process. By the output of this process, a system can be termed as the narrowband or wideband system.

Correlation is a measure of the similarities between two signals as a function of time shift between them. Correlation is primarily a time-domain process. It can be defined for continuous-time as well as discrete-time signals. The procedure for carrying out correlation function can be described in simple steps as firstly, one signal is shifted with respect to the other. The amount of shift is the position of the correlation function point to be calculated. Each element of one signal is multiplied by the corresponding element of the other. The area under the resulting curve is integrated.

When two signals are similar in shape and are in phase, the correlation function is maximum because the product of two signals is positive. Then the area under the curve gives the value of the correlation function at zero point, and this is a large value. When one signal is shifted with respect to the other, the signals go out of phase, the correlation function is minimum because the product of two signals is negative. Then the area under the curve gives the value of the correlation function at the point of shift, and this is a smaller value. The width of the correlation function, where it has significant value, indicates how long the signals remain similar. Correlation is mainly required at the receiver side where signal matching is to be done for acquisition of the system. Correlation requires a lot of computation.

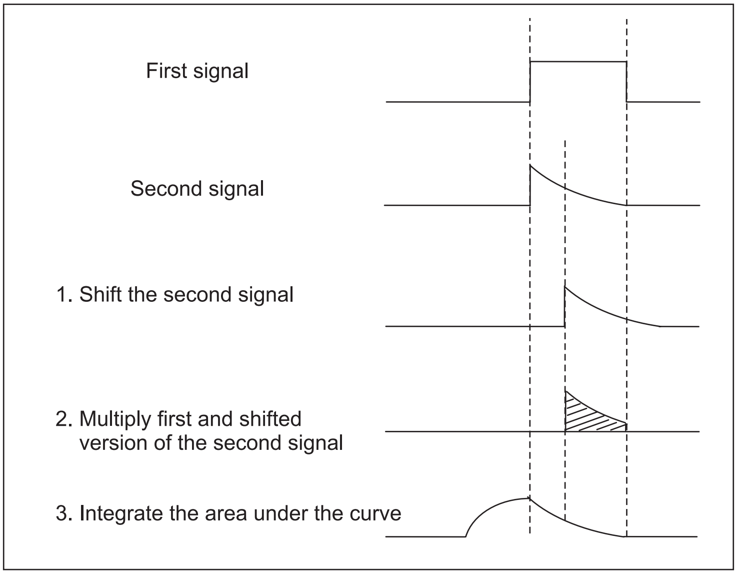


Fig. A.2 Process of Correlation Function

Autocorrelation of Signals

Autocorrelation is defined as the degree of correspondence between the data signal and the phase-shifted replica of itself. Autocorrelation is basically correlation of a signal with itself. The digital realization of the auto-correlation requires some means to perform the 'multiplication and shift' of the samples of the signal with the stored reference signal. The autocorrelation function of a real power signal $s(t)$ is defined as

$$\boxed{R_a(\tau) = \lim_{T\to\infty} \frac{1}{T}\int_{-T/2}^{+T/2} s(t)s(t+\tau)\,d\tau} \text{ for } -\infty < \tau < \infty$$

$R_a(\tau)$ is a function of the time difference τ between original waveform of the signal and its shifted version. The autocorrelation function $R_a(\tau)$ provides a measure of close approximation of the signal with its version shifted τ units in time. If the power signal $s(t)$ is periodic, its autocorrelation function is symmetrical in τ about zero, i.e., $R_a(\tau) = R_a(-\tau)$. Its maximum value occurs at the origin, i.e., $R_a(\tau) \le R_a(0)$ for all values of τ. Its value at the origin is equal to the average power of the signal, i.e., $R_a(0) = \frac{1}{T}\int_{-T/2}^{+T/2} s^2(t)\,dt$. Autocorrelation and Power Spectral Density (PSD) form a Fourier transform pair, i.e., $R_a(\tau) \leftrightarrow G_a(f)$.

The autocorrelation function of a real-valued energy signal $s(t)$ is defined as $\boxed{R_s(\tau) = \int_{-\infty}^{\infty} s(t)\, s(t+\tau)\, d\tau}$ for $-\infty < \tau < \infty$. The autocorrelation function $R_s(\tau)$ is a function of the time difference τ between the original waveform of the energy signal and its time-shifted version. It is symmetrical in τ about zero, i.e., $R_s(\tau) = R_s(-\tau)$. Its maximum value occurs at the origin, i.e., $R_s(\tau) \le R_s(0)$ for all values of τ. Its value at the origin is equal to the energy of the signal, i.e., $R_s(0) = \int_{-\infty}^{\infty} s^2(t)\, dt$. Autocorrelation and Energy Spectral Density (ESD) form a Fourier transform pair, i.e., $R_s(\tau) \leftrightarrow E_s(f)$.

Random noise is only similar to itself with no shift at all. The correlation function of random noise with itself is a single sharp spike at zero shifts. The autocorrelation function of a periodic signal is itself a periodic signal, with the same period as that of the original signal. Periodic signals go in and out of phase as one is shifted with respect to the other. So they will show strong correlation at any shift where the peaks coincide. Short signals can only be similar to themselves for small values of shift and so their autocorrelation functions are short.

Cross-correlation of Signals

Correlation of a signal with another signal is called cross-correlation. The cross-correlation between two different signal waveforms signifies the measure of similarity between one signal and time-delayed version of another signal. The cross-correlation function for two periodic or power signals $s_1(t)$ and $s_2(t)$ is

$$\boxed{R_c(\tau) = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{+T/2} s_1(t) s_2^*(t-\tau)\, dt}$$, where T is the time period.

The cross-correlation for two finite energy signals is given by

$$\boxed{R_c(\tau) = \int_{-\infty}^{\infty} s_1(t) s_2^*(t-\tau)\, dt}$$

When the cross-correlation of two power or energy signals is taken at origin ($\tau = 0$), or $R_c(0) = 0$; then these two signals are called as orthogonal over the complete intervals of time. The cross-correlation function exhibits conjugate symmetry property. It does not exhibit commutative property. If a copy of the known reference signal is correlated with an unknown signal then the correlation will be high when the reference signal is similar to the unknown signal. The unknown signal is correlated with a number of known reference signals. A large correlation value shows the degree of confidence that the reference signal is detected. It also indicates when the reference signal occurs in time. The largest value for correlation is the most likely to match.

The preference of convolution over correlation for filtering function depends on how the frequency spectra of two signals interact with each other. If one signal is symmetric, the process of correlation and convolution are identical and flipping that signal back to front does not change it. Convolution by multiplying frequency spectra can take advantage of the fast Fourier transform, which is a computationally efficient algorithm. Convolution in frequency domain can be faster than convolution in the time domain and is called *fast convolution*.

A.4 PROBABILITY DISTRIBUTION FUNCTIONS

There are various types of probability distribution functions for discrete and continuous random variables. The most important probability distribution functions for discrete random variables are binomial and Poisson probability distributions, and for continuous random variables, Gaussian and Rayleigh probability distributions.

Binomial Probability Distribution Function

Consider a discrete random variable that can take only two values such as True or False, as in case of a binary logic system. If probability of True outcome is p each time it is checked then the probability of False outcome will be $q = (1 - p)$. The probability of x true outcomes in n number of checks is given by a probability distribution function known as *Binomial probability distribution.* That is,

$$\boxed{p(x) = P(X = x) = np^x\, q^{n-x} + xp^x\, q^{n-x}}$$

The mean $\overline{x} = np$, and variance, $\sigma^2 = npq$ of binomial probability distribution can be computed. The binomial probability distribution is quite useful in the context of a digital transmission. For example, to determine whether an information or data transmitted has reached correctly or not at receiver through a communication channel.

Poisson Probability Distribution Function

If n is large and p is small (close to zero), then calculation of a desired probability using Binomial probability distribution method is very difficult. Generally, for $n \geq 50$ and $p \leq 0.1$, Poisson probability distribution give simpler and satisfactory results. According to Poisson probability distribution, the probability function of a discrete random variable X that can assume values 0, 1, 2, … is given as

$$\boxed{p(x) = P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!},\ x = 0, 1, 2, \ldots}$$, where λ is a positive constant.

Poisson probability distribution can be related to the arrival of telephone calls at the telephone exchange or data transmission when error rate is low, or even to emission of electrons in electronic components/devices resulting into shot noise.

Gaussian Probability Distribution Function

Gaussian probability distribution, also known as normal probability distribution, is defined for a continuous random variable. The probability distribution function for a Gaussian random variable X, is expressed as

$$\boxed{p(x) = f_x(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-m_x)^2}{2\sigma^2}};\ -\infty < x < \infty}$$, where m_x is the mean value, and σ is standard deviation (σ^2 is variance) of the random variable.

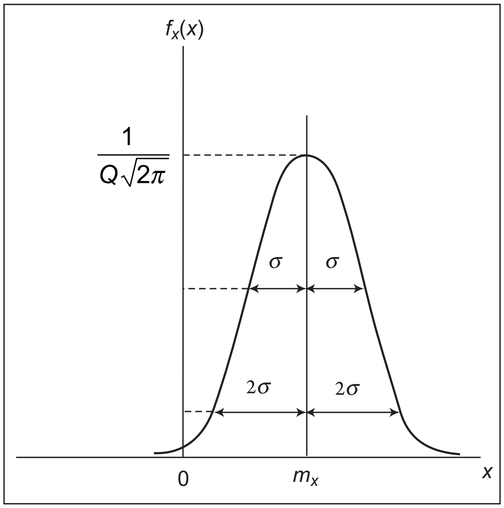


Fig. A.3 Plot of Gaussian Probability Distribution Function

- The peak value of Gaussian probability density spectrum (Gaussian PDS) occurs at $x = m_x$, and is given by $f_x(x)\big|_{\max} = \frac{1}{\sigma\sqrt{2\pi}}$
- The plot of Gaussian PDS exhibit even symmetry around mean value, i.e.,

  $$f_x(m_x - \sigma) = f_x(m_x + \sigma)$$
- The area under the Gaussian PDS curve is one-half each for all values of x below m_x and above m_x respectively. That is, $P(X \le m_x) = P(X > m_x) = \frac{1}{2}$.
- As σ approaches zero, the Gaussian function approaches to impulse function located at $x = m_x$

The corresponding Gaussian probability distribution function is given by

$$\boxed{f(x) = P(X \le x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{\frac{-(x-m_x)^2}{2\sigma^2}} dx}$$

If $m_x = 0$, and $\sigma = 1$ then $f(x) = P(X \le x) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}$, where x is the standardized random variable whose mean value is zero and standard deviation (and hence variance) is unity. $f(x)$ is known as *standard Gaussian density function*. To evaluate Gaussian probability distribution function, the *error function* of x is defined as

$$erf(X) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-x^2} dx; \; 0 < erf(X) < 1$$

$\therefore$ $F(x) = 1 - \frac{1}{2} erfc\left(\frac{x}{\sqrt{2}}\right)$, where *erfc* is complementary error function of x.

Probability Density of Sum of Random Variables

Given two independent Gaussian random variables, the sum of these random variables is itself a Gaussian random variable. Even in the case when the Gaussian random variables are dependent, a linear combination of such random variables is still Gaussian. In general, the probability density function of a sum of random variables of like probability density does not preserve the form of the probability density of individual random variables. However, it tends to approach Gaussian if a large number of random variables are summed, regardless of their probability density. The probability density $f(z)$ of $Z = X + Y$ in terms of the joint probability density $f(x, y)$ can be given as $\boxed{F(z) = P(Z \le z) = P(X \le \infty, Y \le \{z - X\})}$, where z is an arbitrary value of Z. Thus, the probability that $(Z \le z)$ is the same as the probability that $(Y \le \{z - X\})$ independently of the value of X, that is, for $-\infty \le X \le +\infty$. The cumulative distribution function is given as

$$F_{XY}(x, y) = P(X \le x, Y \le y) = \int_{-\infty}^{y}\int_{-\infty}^{x} f_{XY}(x, y)\,dx\,dy$$

$$\therefore \qquad F(z) = \int_{-\infty}^{\infty} dx \int_{-\infty}^{z-x} f(x, y)\,dy$$

Since $f(z) = \int_{-\infty}^{\infty} f(x) f(z - x)\,dx$, the probability density of Z can then be determined as $f(z) = \dfrac{dF(z)}{dz} = \int_{-\infty}^{\infty} f(x, z - x)\,dx.$

This shows that $f(z)$ is the convolution of $f(x)$ and $f(y)$. If $f(x)$ and $f(y)$ are Gaussian probability density functions with $m_x = m_y = 0$, and $\sigma^2 = \sigma_x^2 + \sigma_y^2$ then

$$\boxed{f(z) = \frac{e^{-\frac{z^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}}}$$ which is also a Gaussian probability density function.

- Thus, the sum of two independent Gaussian random variables is itself a Gaussian random variable.
- The variance σ^2 of the sum of random variables is the sum $(\sigma^2_x + \sigma^2_y)$ of the individual independent random variables, not just Gaussian random variables.

If $m_z = m_x + m_y$ then $f(z) = \dfrac{e^{-\frac{(z-m_z)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}}$

Rayleigh Probability Distribution Function

The Rayleigh probability distribution is defined for continuous random variables. It is created from two independent Gaussian random variables having zero mean value and same variance value. Thus, the Rayleigh probability distribution is strongly related to the Gaussian probability distribution. Under properly normalized conditions, the Rayleigh probability distribution function is expressed as $\boxed{f_R(x) = \begin{Bmatrix} \frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}}; 0 \le x \le \infty \\ 0; x < 0 \end{Bmatrix}}$.

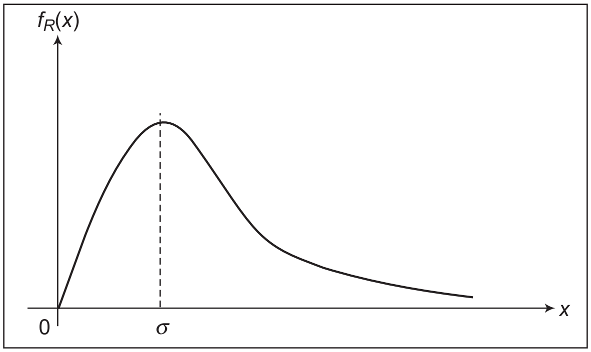


Fig. A.4 Plot of Rayleigh Probability Distribution Function

Its maximum value occurs at $x = \sigma$. Average value, $m_x = 1.25\sigma$ and variance, $\sigma^2 = 2x^2$. The Rayleigh probability distribution is always used for modeling of statistics of signals transmitted through a narrowband noisy wireless communication channel and fading in mobile radio that occurs due to multipath propagation.

Rician Probability Distribution Function

It is expressed as $\boxed{f_{Ri}(x) = \begin{Bmatrix} \frac{x}{\sigma^2} e^{-\frac{x^2+a^2}{2\sigma^2}} I_0\left(\frac{ax}{\sigma^2}\right); 0 \le x \le \infty \\ 0; \qquad x < 0 \end{Bmatrix}}$, where σ^2 is the variance of the Gaussian process, a is the amplitude of the sinusoid, a^2 is sum of square of means of two independent Gaussian processes, and I_0 is modified Bessel function of the first order.

At $a = 0$, it is same as the Rayleigh probability distribution.

As a increases, it approaches Gaussian probability distribution.

The plot of $f_R(x)$ versus x appears to be similar to that of Rayleigh probability distribution curve except that it is shifted towards right, the extent of shift depends on the value of a, i.e., $E[x^2] = 2\sigma^2 + a^2$

The Rician probability distribution is useful in analysis of RF carrier sinusoidal signal passing through narrowband noisy channel or in a signal fading model in line-of-sight wireless communications.

Uniform Probability Distribution Function

A random variable X is said to have a uniform distribution provided its density function is described by $\boxed{f_u(x) = \begin{Bmatrix} \dfrac{1}{b-a}; a \le x \le b \\ 0; \quad \text{otherwise} \end{Bmatrix}}$. Its average value, $\overline{x} = \dfrac{a+b}{2}$ and variance, $\sigma^2 = \dfrac{(b-a)^2}{12}$.

A.5 RANDOM PROCESSES

The collection of an infinite number of information and/or noise signals (random variables that is a function of time) is known as *random process* or a *stochastic process*. The notion of a random process is a natural extension of the random variable. A collection of signal or noise waveforms is called an *ensemble*, and the individual waveforms are called *sample functions*. Random processes are classes of signals whose fluctuations in time are partially or completely random. A *statistical average* may be determined from the measurements made at some fixed time $t = t_1$ on all the sample functions of the ensemble. The *ensemble average* can be determined by squaring and adding each value, and then dividing by the number of sources in the ensemble. Generally, ensemble averages are not same as time averages because the statistical characteristics of the sample functions in the ensemble change with time.

Stationary Random Process

When the statistical characteristics of the sample functions do not change with time, the random process is described as stationary random process. A random process is said to be *Wide-Sense Stationary (WSS)* if its mean and autocorrelation function do not change with a shift in time in the time origin. From a practical point of view, it is not necessary for a random process to be stationary for all time. However, a random process may be stationary for some observation time intervals under consideration.

Ergodic Random Process

A random process is said to be an ergodic random process if its time averages equals its ensemble averages, and the statistical properties can be determined by time averaging over a single sample function of the process. An ergodic random process must be stationary in the strict sense, i.e., $\boxed{m_x = \lim_{T\to\infty} \dfrac{1}{T}\int_{-T/2}^{+T/2} X(t)dt}$. A random process $X(t)$ can be generally classified as a power signal having its power spectral density $G_x(f)$ defined as $G_x(f) = \lim_{T\to\infty} \dfrac{1}{T}\left|X_T(f)\right|^2$, where $X_T(f)$ is a Fourier transform of a truncated version $x_T(t)$ of the nonperiodic power signal $x(t)$ during the interval $(-T/2, +T/2)$. $G_x(f)$ describes the distribution of a signal's power in the frequency domain, so it is particularly useful in communication systems. The signal power that will pass through a communication system having known frequency characteristics can be evaluated by $G_x(f)$.

Gaussian Random Process

Let a random variable Y be a linear function of $X(t)$ and finite, i.e., $Y = \int_0^T X(t)f(t)dt$, where $X(t)$ is a random process, and $f(t)$ is any function which weighs the random process $X(t)$ over the observation time interval $(0 \leq t \leq T)$. If the random variable Y is described for each value of function $f(t)$ then the process $X(t)$ is known as a Gaussian random process. The probability density function of random variable Y is given by $\boxed{f_Y(y) = \frac{1}{\sigma_y\sqrt{2\pi}} e^{\frac{-(y-m_y)^2}{2\sigma_y^2}}}$, where m_y is the mean and σ_y^2 is the variance of the random variable Y.

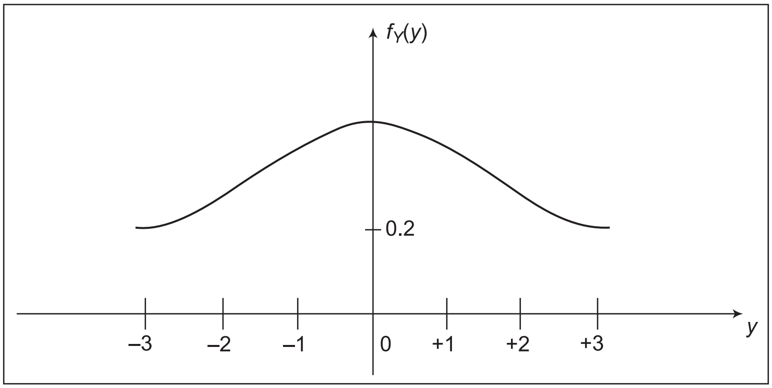


Fig. A.5 Gaussian pdf for $m_y = 0$ and $\sigma^2_y = 0$

If a Gaussian random process $X(t)$ is applied to input of a stable linear filter then the output of the filter is also a Gaussian random process $Y(t)$. If a Gaussian random process is wide-sense stationary then the process is also stationary in the strict sense. If the set of the random variables obtained by sampling a Gaussian random process are uncorrelated, then these are statistically independent. The power spectral density of the sum of uncorrelated WSS random processes is equal to the sum of their individual power spectral densities.

Covariance of Random Process

The *auto-covariance function* of a strictly stationary random process $X(t)$ depends only on the difference between the observation times $(t_2 - t_1)$. It is given by

$$C_x(t_1, t_2) = E[\{X(t_1) - m_x\}\{X(t_2) - m_x\}]$$

$$\Rightarrow \quad C_x(t_1, t_2) = E[X(t_1)\, X(t_2)\} - m_x^2]$$

$$\Rightarrow \quad C_x(t_1, t_2) = R_x(t_2 - t_1) - m_x^2$$

where $R_x(t_2 - t_1)$ is autocorrelation function of a strictly stationary random process $X(t)$ for all t_1 and t_2 and m_x is the mean of a strictly stationary random process which is a constant for all values of t. The auto-covariance function can be uniquely determined if the autocorrelation function and the mean of the random process are known. Also, it is sufficient to describe the first two moments of the random process.

Transmission of Random Processes

Let a random process $X(t)$ is applied at the input of a linear time-invariant system. Let the transfer function of this system be $H(\omega)$. Then the power spectral density of the output random process $Y(t)$ is given by $S_y(\omega) = |H(\omega)|^2\, S_x(\omega)$. The autocorrelation function of this system is given by $R_y(\tau) = h(\tau) * h(-t) * R_x(\tau)$. If two wide-sense stationary random processes $X_1(t)$ and $X_2(t)$ are added to form a random process $Z(t) = X_1(t) + X_2(t)$ then the statistics of $Z(t)$ can be determined in terms of $X_1(t)$ and $X_2(t)$. For example, the auto correlation function of sum of two random processes is given by $R_z(\tau) = \overline{Z(t)Z(t+\tau)}$

$$\Rightarrow \qquad R_z(\tau) = \overline{\left[X_1(t) + X_2(t)\right]\left[X_1(t+\tau) + X_2(t+\tau)\right]}$$

$$\Rightarrow \qquad R_z(\tau) = R_x(\tau) + R_y(\tau) + R_{xy}(\tau) + R_{yx}(\tau)$$

If $X_1(t)$ and $X_2(t)$ are orthogonal (that is, uncorrelated with either m_x or $m_y = 0$) then $R_z(\tau) = R_x(\tau) + Ry(\tau)$, and $\overline{Z^2} = \overline{X_1{}^2} + \overline{X_2{}^2}$. The mean square of a sum of orthogonal random processes is equal to the sum of the mean squares of individual random processes. Similarly, the PSD of sum of two random processes is the sum of the PSDs of individual random processes. That is, $S_z(\omega) = S_x(\omega) + S_y(\omega)$.

Relationship between PSDF and Autocorrelation

The power spectral density function $S_s(f)$ and the autocorrelation function $R_a(\tau)$ of a stationary random process $X(t)$ form a Fourier-transform pair. These are related to each other by the following expressions:

$$\boxed{S_s(f) = \int_{-\infty}^{\infty} R_a(\tau) e^{-j2\pi f\tau} d\tau}$$

$$\boxed{R_a(\tau) = \int_{-\infty}^{\infty} S_s(f) e^{j2\pi f\tau} df}$$

These relations are usually called the *Einstein–Wiener–Khintchine* relations, and are used in the theory of spectral analysis of random processes. Einstein-Wiener-Khintchine relations indicate that if either the PSD or the autocorrelation function of a random process is known, the other can be determined precisely. At $f = 0$, $S_s(0) = \int_{-\infty}^{\infty} R_a(\tau) d\tau$. This implies that the zero-frequency value of the PSD of a stationary random process equals the total area under the autocorrelation function curve. Also at $f = 0$, $R_a(0) = E\left[X^2(t)\right] = \int_{-\infty}^{\infty} S_s(f) df$. The mean-square value of a stationary random process equals the total area under the PSD curve. $S_s(f) \geq 0$ for all values of f, that is, the PSD of a stationary random process is always zero or positive. $S_s(f) = S_s(-f)$ because $R_a(\tau) = R_a(-\tau)$. The PSD of a real-valued random process is an even function of frequency. $p_x(f) = \dfrac{S_x(f)}{\int_{-\infty}^{\infty} S_x(f) df} \geq 0$ for all values of f. This clearly indicates that the normalized

PSD has the properties usually associated with a probability density function. The total area under $p_x(f)$ curve is also unity.

A.6 PROBABILITY OF ERROR AND BIT ERROR RATE (BER)

In general, Probability of Error, P_e is defined as $\boxed{P_e \equiv \lim_{N\to\infty} \frac{N_e}{N}}$, where N_e is the number of instances in which errors were made, and N is the number of messages (usually N is a very large number). For finite value of N, $P_e \neq \frac{N_e}{N}$ and the estimate p of P_e is $\frac{N_e}{N}$ which is a random variable. If average estimate $\overline{p}$ of P_e is determined on the basis of observation of a limitless number of messages, then $\overline{p} = P_e$.

Tchebycheff's inequality yields an upper or lower limit for probability distribution of a random variable in terms of σ^2 and ε, but not an exact value. According to Tchebycheff's inequality, $\boxed{P\left[\left|X-b\right| \geq \varepsilon\right] \leq \frac{E\left[(X-b)^2\right]}{\varepsilon^2}}$, where X is a random variable, b is any real number, ε is any positive number, and $E[(X-b)^2]$ is finite. If $b = m_x$ (mean value of X) and $\varepsilon = k\sigma$, then $P\left[\left|X-m_x\right| \geq k\sigma\right] \leq \frac{1}{k^2}$. The variance of a random variable is closely related to the width of the probability density function. It is more precisely related to the deviation of the distribution about the mean value. Small variance signifies that it is more probable to have small deviations from the mean. Large deviations from the mean are almost improbable. If the probability distribution of a random variable X is known then its mean and variance can be computed. If the mean and variance of a random variable are known, then its probability distribution cannot be known. This implies that $P\left[\left|X-b\right| \geq \varepsilon\right]$ or $P\left[\left|X-b\right| < \varepsilon\right]$ cannot be known. But with the help of Tchebycheff's inequality, an upper or lower bounds to such probabilities can be obtained. Applying Tchebycheff's inequality to the random variable $x \equiv \left|p-P_e\right|$, we have $\sigma_x^{\,2} = \overline{\left|p-P_e\right|^2}$

$$\Rightarrow \qquad \sigma_x^{\,2} = \overline{p^2 + P_e^{\,2} - 2 \times p \times P_e}$$

$$\Rightarrow \qquad \sigma_x^{\,2} = \overline{p^2} + P_e^{\,2} - 2 \times \overline{p} \times P \text{, where } \overline{p^2} = \frac{1}{N^2}(N)P_e + \frac{1}{N^2}(N)(N-1)P_e^2$$

$$\Rightarrow \qquad \sigma_x^{\,2} = \frac{P_e - P_e^{\,2}}{N}$$

For $P_e \ll 1$ (as in any communications system), $\sigma_x^{\,2} = \frac{P_e}{N}$

$$\therefore \qquad \boxed{P\left[\left|(p-P_e)\right| > \varepsilon\right] \leq \frac{P_e}{N\varepsilon^2}}$$

Generally, the typical values of $\varepsilon = \frac{P_e}{2}$ and $P(|(p - P_e)| > \varepsilon) \le 10\%$ are acceptable. In that case, $0.01 \simeq \frac{P_e}{N\left(\frac{P_e}{2}\right)^2} \simeq \frac{4}{NP_e}$; $\Rightarrow N \approx \frac{40}{P_e}$. Thus, $N = \frac{40}{P_e}$ should be considered for all practical purposes.

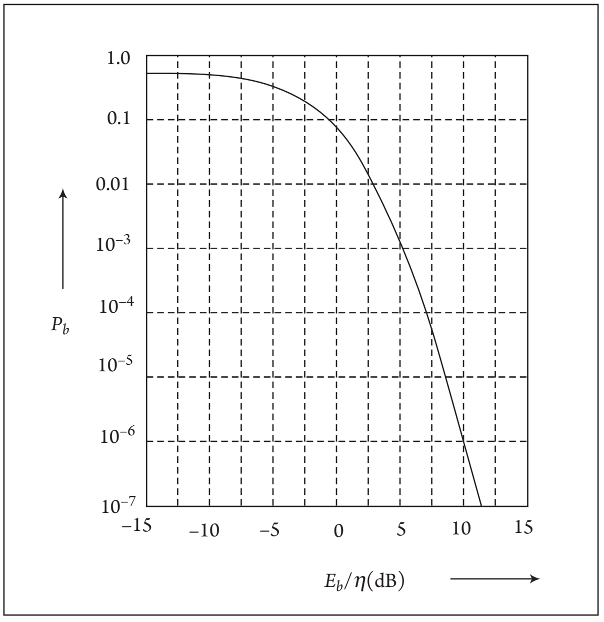


Fig. A.6 *Plot of P_b versus Eb/η*

In the binary case, the probability of error, or error probability is called the bit error probability or *Bit Error Rate (BER)*, and is denoted by P_b. The parameter $\frac{E_b}{\eta}$ is the normalized energy per bit, and is used as a figure of merit in digital communication. Because the signal power is equal to E_b times the bit rate, a given E_b is equivalent to a given signal power for a specified bit rate. As expected, P_b decreases as $\frac{E_b}{\eta}$ increases.

APPENDIX B The Fourier Transform

B.1 REVIEW OF THE FOURIER SERIES

Signals that are periodic functions of time with finite energy within each time period T can be represented by an infinite series called the *Fourier series*. The Fourier series provides a frequency domain model for periodic signals. It is useful to analyze the frequency components of periodic signals. It is useful to determine the steady-state response of systems for periodic input signals, rather than transient analysis. The Fourier series may be used to represent either functions of time or functions of space co-ordinates. The Fourier series can be used to represent a periodic power signal over the time interval $(-\infty, +\infty)$ as well as to represent an energy signal that is strictly time limited.

Trigonometric Form of the Fourier Series

Let $s(t)$ be a periodic function of period T_0, and $s(t) = s(t + T_0)$ for all values of t. A periodic function $s(t)$ can be expressed in the trigonometric form of the Fourier series, which can then be represented as

$$\boxed{s(t) = A_0 + \sum_{n=1}^{n=\infty} A_n \cos\left(\frac{2\pi nt}{T_0}\right) + \sum_{n=1}^{n=\infty} B_n \sin\left(\frac{2\pi nt}{T_0}\right)}$$

where A_0 is the average value of the signal $s(t)$, and is given by $A_0 = \frac{1}{T_0}\int_{-T_0/2}^{+T_0/2} s(t)dt$.

The coefficient A_n is given by $A_n = \frac{2}{T_0}\int_{-T_0/2}^{+T_0/2} s(t)\cos\left(\frac{2\pi nt}{T_0}\right)dt$.

The coefficient B_n is given by $B_n = \frac{2}{T_0}\int_{-T_0/2}^{+T_0/2} s(t)\sin\left(\frac{2\pi nt}{T_0}\right)dt$.

Dirichlet's Conditions for Fourier Series

The Fourier series exists only when the periodic function, $s(t)$ satisfies the *Dirichlet's conditions* as stated below:

1. It has a finite average value over the period T_0.
2. It is well defined and single-valued, except possibly at a finite number of points.
3. It must posses only a finite number of discontinuities in the period T_0.
4. It must have a finite number of positive and negative maxima in the period T_0.

For an even waveform, the trigonometric Fourier series has only cosine terms such as $\boxed{s(t) = C_0 + \sum_{n=1}^{n=\infty} C_n \cos\left(\frac{2\pi nt}{T_0} - \phi_n\right)}$, where $C_0 = A_0 = \frac{1}{T_0}\int_{-T_0/2}^{+T_0/2} s(t)dt$, $C_n = \sqrt{(A_n^2 + B_n^2)}$,

and $\phi_n = \tan^{-1}\frac{B_n}{A_n}$. The coefficients C_n are called *spectral amplitudes*. Thus, the Fourier series of a periodic function $s(t)$ consists of sum of harmonics of a fundamental frequency $f_0 = \frac{1}{T_0}$.

Exponential Form of the Fourier Series

The Fourier series of the periodic function $s(t)$ can also be represented in an exponential form with complex terms as

$$s(t) = \sum_{n=-\infty}^{n=\infty} A_n e^{j\frac{2\pi nt}{T_0}}\text{; where } A_n = \frac{1}{T_0}\int_{-T_0/2}^{+T_0/2} s(t)e^{-j\frac{2\pi nt}{T_0}}dt$$

A nonperiodic signal can be viewed as a limiting case of a periodic signal in which the signal period approaches infinity. It can be represented by the Fourier series in order to develop the frequency domain representation.

$$\boxed{s(t) = \frac{1}{T}\left[\sum_{n=-\infty}^{n=\infty} S_n e^{+jn\omega_o t}\right]}\text{; where } S_n = \int_{-T/2}^{+T/2} s(t)e^{-jn\omega_o t}dt$$

The spacing between any two successive harmonics is given by

$$\omega_o = \frac{2\pi}{T} = \Delta\omega \text{ (say);} \quad \Rightarrow \quad \frac{1}{T} = \frac{\Delta\omega}{2\pi}$$

Therefore, $s(t) = \frac{1}{2\pi}\left[\sum_{n=-\infty}^{n=\infty} S_n e^{+jn\omega_o t}(\Delta\omega)\right]$

As $T \to \infty$, then $\Delta\omega \to 0$, S_n becomes a continuous function $S(\omega)$ given as

$$\boxed{S(\omega) = \lim_{T\to\infty} S_n = \int_{-\infty}^{+\infty} s(t)e^{-j\omega_o t}dt}$$

The Fourier series for a periodic signal becomes the Fourier integral representation for a non-periodic signal over $-\infty$ to $+\infty$, given as

$$s(t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} S(\omega)e^{+j\omega_o t}d\omega$$

$$s(\omega) = \int_{-\infty}^{+\infty} s(t)e^{-j\omega_o t}dt$$

Thus, $s(t)$ and $S(\omega)$ constitute a Fourier transform pair, written as

$$S(\omega) = \Im[s(t)]\text{, and } s(t) = \Im^{-1}[S(\omega)]$$

Normalized Power in a Fourier Expansion

The normalized power of a periodic function of time $s(t)$ in the time domain can be given as $\boxed{S = A_0^2 + \sum_{n=1}^{\infty} \frac{A_n^2}{2} + \sum_{n=1}^{\infty} \frac{B_n^2}{2}}$. The total normalized power is the sum of the normalized power due to each term in the series separately associated with a real waveform. This is due to orthogonality of the sinusoids used in a Fourier expansion. For a periodic function, the power in a waveform $s(t)$ can be evaluated by adding together the powers contained in each frequency component of the signal $s(t)$.

B.2 THE FOURIER TRANSFORM

The Fourier spectrum of a signal indicates the relative amplitudes and phases of the sinusoids that are required to synthesize that signal. The periodic signal can be expressed as a sum of discrete exponentials with finite amplitudes. It can also be expressed as a sum of discrete spectral components each having finite amplitude and separated by finite frequency intervals, $f_0 = 1/T_0$ where T_0 is its time period. The normalized power as well as the energy (in an interval T_0) of the signal is finite. Aperiodic waveforms may be expressed by Fourier transforms. Consider a single-pulse (centred on $t = 0$) aperiodic waveform. As T_0 approaches infinity, the spacing between spectral components becomes infinitesimal and the frequency of the spectral components becomes a continuous variable. The normalized energy of the aperiodic waveform remains finite but its normalized power becomes infinitesimal and so also the spectral amplitudes.

The Fourier spectrum of a periodic signal $s(t)$ with period T_0 can be expressed in terms of complex Fourier series as

$$s(t) = \sum_{n=-\infty}^{n=\infty} S_n e^{jn\omega_o t}; \text{ where } \omega_o = \frac{2\pi}{T_0}$$

$\boxed{\sum_{n=-\infty}^{n=\infty} S_n e^{jn\omega_o t} = \int_{-\infty}^{\infty} S(f) e^{jn\omega_o t} df}$; where $S(f)$ is called the Fourier transform of $s(t)$.

Therefore, $S(f) = \int_{-\infty}^{\infty} s(t) e^{-jn\omega_o t} dt$; and $S_n = \int_{-\infty}^{\infty} s(t) e^{-jn\omega_o t} dt$.

Thus, the spectrum is discrete, and $s(t)$ is expressed as a sum of discrete exponentials with finite amplitudes. In communication systems, aperiodic energy signals may be strictly time limited, or asymptotically time limited. For an aperiodic signal, the spectrum exists for every value of frequency but the amplitude of each component in the spectrum is zero. The spectrum of the aperiodic signal becomes continuous. The spectral density per unit bandwidth is more meaningful instead of the amplitude of a component of some frequency. The signal should follow Dirichlet's conditions for the Fourier transform, as stated in the Fourier series.

Expressing Fourier transform in angular frequency ω_o, where $\omega_o = 2\pi f_0$, we have

$$s(\omega) = \Im[s(t)] = \Im\left[\sum_{n=-\infty}^{n=\infty} S_n e^{jn\omega_o t}\right] = \int_{-\infty}^{\infty} s(t) e^{-j\omega_o t} dt$$

$$s(t) = \Im^{-1}[S(\omega)] = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\, e^{j\omega_o t}\, dt$$

Also, $\Im[s(t)] = \sum_{n=-\infty}^{n=\infty} S_n \Im\left[1.e^{jn\omega_o t}\right]$

Using $\Im\left[1.e^{jn\omega_o t}\right] = 2\pi\delta(\omega - n\omega_o)$, we have

$$\Im[s(t)] = 2\pi \sum_{n=-\infty}^{n=\infty} S_n \delta(\omega - n\omega_o)$$

The Fourier transform of a periodic function consists of a series of equally spaced impulses. The plot of amplitudes at different frequency components (harmonics) for a periodic signal $s(t)$ is known as *discrete frequency spectrum*. As the repetition period approaches infinity, the periodic signal $s(t)$ will become non-periodic. Therefore, the discrete spectrum will become a continuous spectrum.

B.3 PROPERTIES OF THE FOURIER TRANSFORM

It is quite obvious that compressing the signal in time domain results in expanding it in frequency domain and vice versa. The notation $s(t) \leftrightarrow S(\omega)$ is used to represent Fourier Transform pairs to describe the properties of Fourier transform. Fourier transform has the following major properties:

Time-Scaling Property

If $s(t) \leftrightarrow S(\omega)$ then $s(bt) \leftrightarrow \frac{1}{|b|} S\left(\frac{\omega}{b}\right)$; where b is a real constant. The function $s(bt)$ represents compression of $s(t)$ in time-domain. $S\left(\frac{\omega}{b}\right)$ represents expansion of $S(\omega)$ in frequency domain by the same factor. Time expansion of a signal results in its spectral compression, and time compression of a signal results in its spectral expansion. That means, if the signal has wide duration, its spectrum is narrower, and vice versa. Doubling the signal duration halves its bandwidth, and vice versa.

Time-Shifting Property

If $s(t) \leftrightarrow S(\omega)$, then $s(t - b) \leftrightarrow S(\omega)e^{-jwb}$; where b is a real constant. A shift in the time domain by a real constant factor b is equivalent to multiplication by $e^{-j\omega b}$ in the frequency domain. The magnitude spectrum remains unchanged. Phase spectrum is changed by $-\omega b$.

Frequency-Shifting Property

If $s(t) \leftrightarrow S(\omega)$, then $e^{j\omega_o t} s(t) \leftrightarrow S(\omega - \omega_o)$. The multiplication of a function $s(t)$ by $e^{j\omega_o t}$ is equivalent to shifting its Fourier transform $S(\omega)$ by an amount ω_o.

Convolution Property

The convolution function $s(t)$ of two time functions $s_1(t)$ and $s_2(t)$ is given as

$$s(t) = \int_{-\infty}^{+\infty} s_1(\tau)s_2(t-\tau)d\tau$$

$$\Rightarrow \qquad s(t) = s_1(t) \otimes s_2(t)$$

Convolution is a mathematical operation used for describing the input-output relationship in a linear time-invariant system.

Time-Convolution Theorem

It states that convolution of two functions in time domain is equivalent to multiplication of their spectra in frequency domain. That is,

If $s_1(t) \leftrightarrow S_1(\omega)$, and $s_2(t) \leftrightarrow S_2(\omega)$ then

$$s_1(t) \leftrightarrow s_2(t) \leftrightarrow S_1(\omega) \times S_2(\omega)$$

Frequency-Convolution Theorem

It states that multiplication of two functions in time domain is equivalent to convolution of their spectra in frequency domain. That is,

If $s_1(t) \leftrightarrow S_1(\omega)$, and $s_2(t) \leftrightarrow S_2(\omega)$ then

$$s_1(t) \times s_2(t) \leftrightarrow S_1(\omega) \leftrightarrow S_2(\omega)$$

Table B.1 Properties of Fourier Transform

S. No.	Property	$s(t)$	$S(\omega)$
1.	Time scaling	$s(bt)$	$\frac{1}{\lvert b\rvert}S\left(\frac{\omega}{b}\right)$
2.	Time shifting	$s(t-T)$	$S(\omega)e^{-j\omega T}$
3.	Frequency shifting	$s(t)e^{j\omega_0 T}$	$s(\omega-\omega_0)$
4.	Time convolution	$s_1(t) \leftrightarrow s_2(t)$	$S_1(\omega)\, S_2(\omega)$
5.	Frequency convolution	$s_1(t)\, s_2(t)$	$S_1(\omega) \leftrightarrow S_2(\omega)$
6.	Time differentiation	$\frac{d^n s(t)}{dt}$	$(j\omega)^n\, S(\omega)$
7.	Time integration	$\int_{-\infty}^{t} s(x)dx$	$\frac{S(\omega)}{j\omega}+\frac{1}{2}S(0)\delta(\omega)$
8.	Duality	$S(t)$	$s(-\omega)$
9.	Superposition	$s_1(t)+ s_2(t)$	$S_1(\omega) + S_2(\omega)$
10.	Scalar product	$ks(t)$	$kS(\omega)$

Table B.2 Fourier Transform Pairs

S. No.	Time Function	Fourier Transform
1.	$\delta(t)$	1
2.	1	$\delta(f)$
3.	$\delta(t-\tau)$	$e^{-j2\pi f\tau}$
4.	$u(t)$	$\frac{1}{2}\delta(f)+\frac{1}{j2\pi f}$
5.	$\sum_{i=-\infty}^{\infty}\delta(t-i\tau)$	$\frac{1}{\tau}\sum_{n=-\infty}^{\infty}\delta\left(f-\frac{n}{\tau}\right)$
6.	$e^{j2\pi f_c t}$	$\delta(f-f_c)$
7.	$\sin(2\pi f_c t)$	$\frac{1}{2j}[\delta(f-f_c)-\delta(f+f_c)]$
8.	$\cos(2\pi f_c t)$	$\frac{1}{2}[\delta(f-f_c)-\delta(f+f_c)]$
9.	$sgn(t)$	$\frac{1}{i\pi f}$
10.	$\frac{1}{\pi t}$	$-jsgn(f)$
11.	$sinc(2Bt)$	$\frac{1}{2B}rect\left(\frac{f}{2B}\right)$
12.	$rect\left(\frac{t}{T}\right)$	$T\,sinc(fT)$
13.	$e^{-at}u(t),\ a>0$	$\frac{1}{a+j2\pi f}$
14.	$e^{-a\lvert t\rvert},\ a>0$	$\frac{2a}{a^2+(2\pi f)^2}$
15.	$e^{-\pi t^2}$	$e^{-\pi f^2}$

APPENDIX C

Power Spectra of Discrete PAM Signals

The power spectra of discrete PAM signals or various line codes depends on the signaling pulse waveform. It also depends on the statistical properties of the sequence of transmitted bits. It is desirable that the spectrum of the transmitted discrete PAM signals should have small power content at low frequencies. This is necessary for transmission over a channel that has a poor amplitude response in the low-frequency range. In a synchronous digital transmission systems, digital data is transmitted serially over a communication channel.

Consider a discrete PAM signal, $X(t)$ ranging from $k = -\infty$ to $k = +\infty$

$$X(t) = \sum_{k=-\infty}^{\infty} A_k v(t - kT_b)$$

The coefficient A_k is a discrete random variable, and $v(t)$ is a basic pulse. It is described in Table C.1 for different signaling data formats.

Table C.1 Value of the coefficient A_k for various data formats

<table>
<tr><th>S. No.</th><th>Type of Data Format</th><th>Symbol</th><th>A_k=</th><th>Basic pulse $v(t)$ is ……</th></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Unipolar NRZ</td><td>1</td><td>$+V$</td><td rowspan="7">Rectangular pulse of unit magnitude, centered at the origin ($t = 0$) and normalized such that $v(0) = 1$. The duration of the pulse is T_b seconds.</td></tr>
<tr><td>0</td><td>0</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Polar NRZ</td><td>1</td><td>$+V$</td></tr>
<tr><td>0</td><td>$-V$</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2">Bipolar NRZ</td><td>1</td><td>$+V$, $-V$ for alternating 1s</td></tr>
<tr><td>0</td><td>0</td></tr>
<tr style="display:none"></tr>
<tr><td rowspan="2">4</td><td rowspan="2">Manchester</td><td>1</td><td>+V</td><td rowspan="2">Doublet plus of magnitude ±1 , and total duration of the pulse is T_b seconds.</td></tr>
<tr><td>0</td><td>–V</td></tr>
</table>

The Power Spectral Density (PSD) of the discrete PAM signal $X(t)$ is given by

$$P_X(f) = \frac{1}{T_b}\left|V(f)\right|^2 \sum_{n=-\infty}^{\infty} R_A(n) e^{(-j2\pi n f T_b)}$$

where $V(f)$ is the Fourier transform of the basic pulse $v(t)$; and $R_A(n) = E[A_k A_{k-n}]$ is the ensemble-averaged autocorrelation function.

The values of the function $V(f)$ and $R_A(n)$ depend on the type of discrete PAM signal data format.

C.1 PSD OF UNIPOLAR NRZ DATA FORMAT

Assume that the occurrence of binary 0s and 1s of a random binary sequence is equally probable. For unipolar Non-Return-to-Zero (NRZ) format of the digital data representation, we have $p\left[A_k = 0\right] = p\left[A_k = +V\right] = \frac{1}{2}$

For $n = 0$, $\qquad R_A(0) = E[A_k\, A_{k}–0] = E[A_k]^2 \qquad$ (E is the expectation operator)

$$\Rightarrow \qquad E[A_k]^2 = (0)^2\, p[A_k = 0] + (+V)^2\, p[A_k = +V]$$

$$\Rightarrow \qquad E[A_k]^2 = 0 \times \frac{1}{2} + V^2 \times \frac{1}{2} = \frac{V^2}{2}$$

For $n \neq 0$; the dibit represented by the product $A_k\, A_{k-n}$ can assume only the following four possible values

$(A_k\, A_{k-n})$;	(00);	$(0 \times 0) = 0$
$(A_k\, A_{k-n})$;	(01);	$(0 \times V) = 0$
$(A_k\, A_{k-n})$;	(10);	$(V \times 0) = 0$
$(A_k\, A_{k-n})$;	(11);	$(V \times V) = V^2$

Assuming that the successive symbols in the binary sequence are statistically independent, these four values occur with equal probability, that is, ¼ each. Hence,

For $n \neq 0$; $\qquad R_A(n) = E\left[A_k A_{k-n}\right] = 0 \times \frac{1}{4} + 0 \times \frac{1}{4} + 0 \times \frac{1}{4} + V^2 \times \frac{1}{4} = \frac{V^2}{4}$

Now, the Fourier transform of the basic pulse, $v(t)$ can be given as

$$V(f) = T_b \sin c(fT_b);\ \text{where } \sin c(fT_b) = \frac{\sin(\pi f T_b)}{\pi f T_b}$$

The power spectral density of the unipolar NRZ format can be derived as follows:

$$P_X(f) = \frac{1}{T_b}\left|T_b \sin c(fT_b)\right|^2 \left(\sum_{n=0} R_A(n) e^{(-j2\pi n f T_b)} + \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} R_A(n) e^{(-j2\pi n f T_b)} \right)$$

$$\Rightarrow \qquad P_X(f) = \frac{1}{T_b} T_b^{\,2} \sin c^2(fT_b) \left(\frac{V^2}{2} + \frac{V^2}{4} \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} e^{(-j2\pi n f T_b)} \right)$$

$$\Rightarrow \qquad P_X(f) = \frac{V^2 T_b}{2} \sin c^2(fT_b) + \frac{V^2 T_b}{4} \sin c^2(fT_b) \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} e^{(-j2\pi n f T_b)}$$

By Poisson's formula,

$$\sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} e^{\left(-j2\pi n f T_b\right)} = \frac{1}{T_b} \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} \delta\left(f - \frac{n}{T_b}\right);\ \text{where } \delta(t) \text{ denotes a Dirac delta function at } f = 0$$

$$\therefore \qquad P_X(f) = \frac{V^2 T_b}{2} \sin c^2(fT_b) + \frac{V^2 T_b}{4} \sin c^2(fT_b) \left[\frac{1}{T_b} \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} \delta\left(f - \frac{n}{T_b}\right) \right]$$

$$\Rightarrow \qquad P_X(f) = \frac{V^2 T_b}{2}\sin c^2(fT_b) + \frac{V^2}{4}\sin c^2(fT_b)\left[\sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} \delta\left(f - \frac{n}{T_b}\right)\right]$$

Since sin $c(fT_b)$ function has nulls at $f = \pm\frac{1}{T_b}, \pm\frac{1}{2T_b}, \ldots$, therefore

$$\boxed{P_X(f) = \frac{V^2 T_b}{2}\sin c^2(fT_b) + \frac{V^2}{4}\delta(f)}$$

This is the expression for PSD of unipolar NRZ digital data sequence.

- The presence of the Dirac delta function $\delta(f)$ in the second term accounts for one half of the power contained in the unipolar NRZ data format.
- Specifically the power spectral density $P_X(f)$ is normalized with respect to V^2T_b, and the frequency f is normalized with respect to the bit rate $\frac{1}{T_b}$.

Figure C.1 shows the power spectral density of unipolar NRZ waveform.

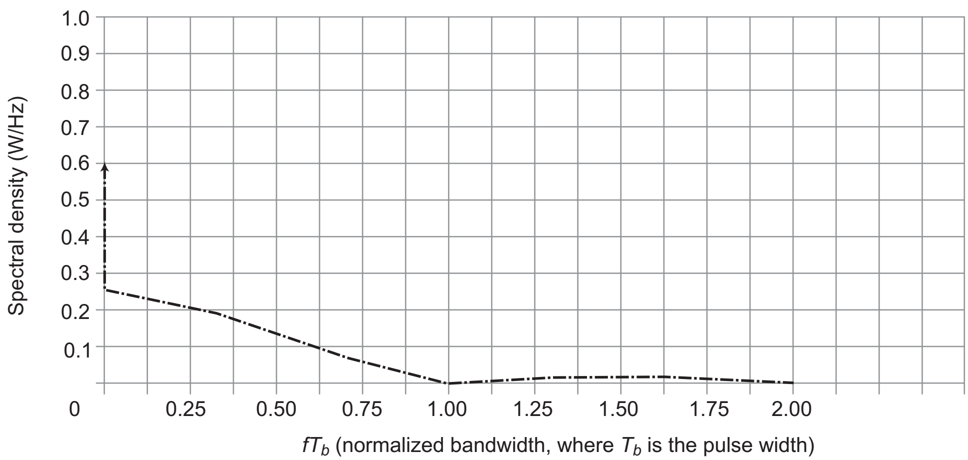


Fig. C.1 Power Spectra of Unipolar NRZ Data Format

The PSD of unipolar NRZ digital data sequence has the following properties:

1. The value of $P_X(f)$ is maximum at $f = 0$.
2. $P_X(f)$ at all multiples of the bit rate, $f_b = \frac{1}{T_b}$.
3. The peak value of $P_X(f)$ between f_b and $2f_b$ occurs at $f = 1.5f_b$, and is 14 dB lower than the peak value of $P_X(f)$ at $f = 0$.
4. The main lobe centred around $f = 0$ has 90% of the total power.
5. When the NRZ signal is transmitted through an ideal low-pass filter with cut-off frequency at $f = f_b$, the total power is reduced by 10% only.
6. The NRZ signal has no dc component.
7. The power in the frequency range from $f = 0$ to $f = \pm\Delta f$ is $2G(f)\,\Delta f$.
8. In the limit as Δf approaches to zero, the PSD becomes zero.

C.2 PSD OF POLAR NRZ DATA FORMAT

Assume that the occurrence of binary 0s and 1s of a random binary sequence is equally probable. For polar NRZ format of the digital data representation, we have

$$p[A_k = -V] = p[A_k = +V] = \frac{1}{2}$$

For $n = 0$; $$E[A_k]^2 = (-V)^2\, p[A_k = 0] + (+V)^2\, p[A_k = +V]$$

$\Rightarrow$ $$E[A_k]^2 = V^2 \times \frac{1}{2} + V^2 \times \frac{1}{2} = V^2$$

For $n \neq 0$, the dibit represented by the product $A_k\, A_{k-n}$ can assume only the following four possible values:

$(A_k\, A_{k-n})$;	(00);	$(-V \times -V) = V^2$
$(A_k\, A_{k-n})$;	(01);	$(-V \times V) = -V^2$
$(A_k\, A_{k-n})$;	(10);	$(V \times -V) = -V^2$
$(A_k\, A_{k-n})$;	(11);	$(V \times V) = V^2$

Assuming that the successive symbols in the binary sequence are statistically independent, these four values occur with equal probability, that is, ¼ each. Hence,

For $n \neq 0$; $$R_A(n) = E\left[A_k A_{k-n}\right] = V^2 \times \frac{1}{4} + (-V^2) \times \frac{1}{4} + (-V^2) \times \frac{1}{4} + V^2 \times \frac{1}{4} = 0$$

Now, the Fourier transform of the basic pulse, $v(t)$ can be given as

$$V(f) = T_b \sin c(fT_b); \text{ where } \sin c(fT_b) = \frac{\sin(\pi f T_b)}{\pi f T_b}$$

The power spectral density of the polar NRZ format can be derived as follows:

$$P_X(f) = \frac{1}{T_b}\left|T_b \sin c(fT_b)\right|^2 \left(\sum_{n=0} R_A(n) e^{(-j2\pi n f T_b)} + \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} R_A(n) e^{(-j2\pi n f T_b)} \right)$$

$\Rightarrow$ $$P_X(f) = \frac{1}{T_b} T_b^2 \sin c^2(fT_b) \left(\frac{V^2}{2} + 0 \times \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} e^{(-j2\pi n f T_b)} \right)$$

$\Rightarrow$ $$\boxed{P_X(f) = \frac{V^2 T_b}{2} \sin c^2(fT_b)}$$

This is the expression for PSD of polar NRZ digital data sequence.

Specifically, the power spectral density $P_X(f)$ is normalized with respect to V^2T_b, and the frequency f is normalized with respect to the bit rate $\frac{1}{T_b}$.

Figure C.2 shows the power spectral density of a polar NRZ waveform.

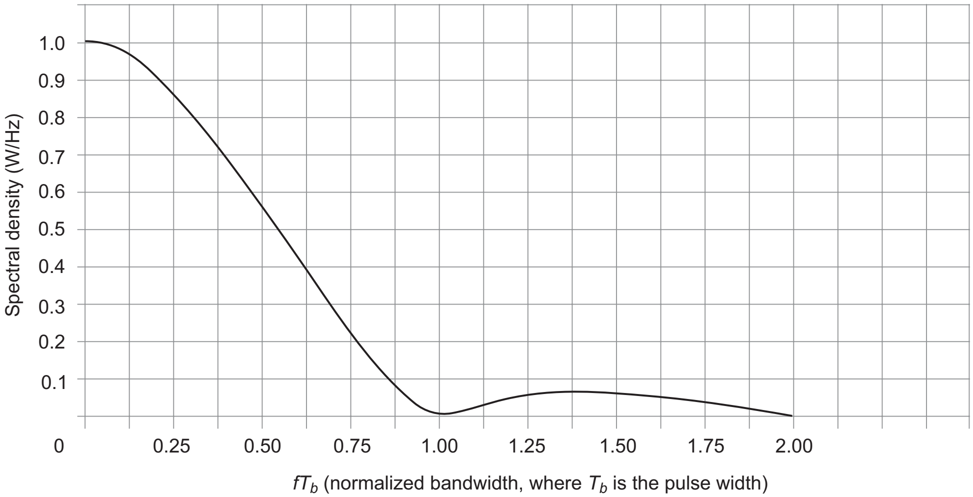


Fig. C.2 Power Spectra of Polar NRZ Data Format

It is observed that most of the power of the polar NRZ format lies within the main lobe of the sine-shaped curve, which extends up to the bit rate $\frac{1}{T_b}$.

C.3 PSD OF BIPOLAR NRZ DATA FORMAT

In bipolar NRZ data format (also known as pseudo-ternary signaling), there are three different levels, $-V$, 0, and $+V$, $+V$ and $-V$ levels are used alternately for transmission of successive 1s, and no pulse for transmission of binary 0. Assume that the occurrence of binary 0s and 1s of a random binary sequence is equally probable. For bipolar NRZ format of the digital data representation, the respective probability of occurrence of three levels will be

$$p\left[A_k = +V\right] = \frac{1}{4}; \quad p\left[A_k = 0\right] = \frac{1}{2}; \quad p\left[A_k = -V\right] = \frac{1}{4}$$

For $n = 0$; $$E[A_k]^2 = (+V)^2\, p[A_k = V] + (0)^2\, p[A_k = 0] + (-V)^2\, p[A_k = -V]$$

$\Rightarrow$ $$E[A_k]^2 = V^2 \times \frac{1}{4} + 0 \times \frac{1}{2} + V^2 \times \frac{1}{4} = \frac{V^2}{2}$$

For $n = 1$, the dibit represented by the product $A_k\, A_{k-n}$ can assume only the following four possible values

$(A_k\, A_{k-n})$;	(00);	$(0 \times 0) = 0$
$(A_k\, A_{k-n})$;	(01);	$(0 \times \mathrm{V})$ or $(0 \times -V) = 0$
$(A_k\, A_{k-n})$;	(10);	$(V \times 0)$ or $(-V \times 0) = 0$
$(A_k\, A_{k-n})$;	(11);	$(V \times -V)$ or $(-V \times V) = -V^2$

Assuming that the successive symbols in the binary sequence are statistically independent and occur with equal probability, then each of the four dibits occurs with equal probability, that is, ¼ each. Therefore, For $n = 1$; $R_A(n) = E[A_k A_{k-n}] = 0 \times \frac{1}{4} + 0 \times \frac{1}{4} + 0 \times \frac{1}{4} + (-V^2) \times \frac{1}{4} = \frac{-V^2}{4}$

Using the property of autocorrelation function as $R_A(n) = R_A(-n)$, we have

For $n = -1$; $R_A(n) = \frac{-V^2}{4}$

For $n > 1$, we know that the digits represented by the sequence $\{A_{k-1}, A_k, A_{k+1}\}$ has $R_A(>n) = E[A_k A_{k-n}] = 0$. Therefore,

For $\{n \neq 0; n \neq \pm 1\}$; $R_A(n) = E[A_k A_{k-n}] = 0$

Now, the Fourier transform of the basic pulse, $v(t)$ can be given as

$$V(f) = T_b \sin c(fT_b); \text{ where } \sin c(fT_b) = \frac{\sin(\pi f T_b)}{\pi f T_b}$$

Thus, the power spectral density of the bipolar NRZ format can be derived as follows:

$$P_X(f) = \frac{1}{T_b}\left|T_b \sin c(fT_b)\right|^2 \left(\sum_{n=0} R_A(n) e^{(-j2\pi n f T_b)} + \sum_{n=+1} R_A(n) e^{(-j2\pi n f T_b)} + \sum_{n=-1} R_A(n) e^{(-j2\pi n f T_b)} + \sum_{\substack{n=-\infty \\ n\neq 0 \\ n \neq \pm 1}}^{\infty} R_A(n) e^{(-j2\pi n f T_b)} \right)$$

$$\Rightarrow \quad P_X(f) = \frac{1}{T_b} T_b^2 \sin c^2(fT_b)\left[\frac{V^2}{2} + \left(-\frac{V^2}{4}\right)e^{(-j2\pi f T_b)} + \left(-\frac{V^2}{4}\right)e^{(j2\pi f T_b)} + 0\right]$$

$$\Rightarrow \quad P_X(f) = T_b \sin c^2(fT_b)\left(\frac{V^2}{2} - \frac{V^2}{4}\left[e^{(-j2\pi f T_b)} + e^{(j2\pi f T_b)}\right]\right)$$

$$\Rightarrow \quad P_X(f) = \frac{V^2 T_b}{2} \sin c^2(fT_b)\left(1 - \frac{1}{2}\left[e^{(-j2\pi f T_b)} + e^{(j2\pi f T_b)}\right]\right)$$

$$\Rightarrow \quad P_X(f) = \frac{V^2 T_b}{2} \sin c^2(fT_b)[1 - \cos(2\pi f T_b)]$$

$$\Rightarrow \quad \boxed{P_X(f) = V^2 T_b \sin c^2(fT_b) \sin^2(\pi f T_b)}$$

This is the expression for PSD of bipolar NRZ digital data sequence.

Specifically, the power spectral density $P_X(f)$ is normalized with respect to $V^2 T_b$, and the frequency f is normalized with respect to the bit rate $\frac{1}{T_b}$.

Figure C.3 shows the power spectral density of a bipolar NRZ waveform.

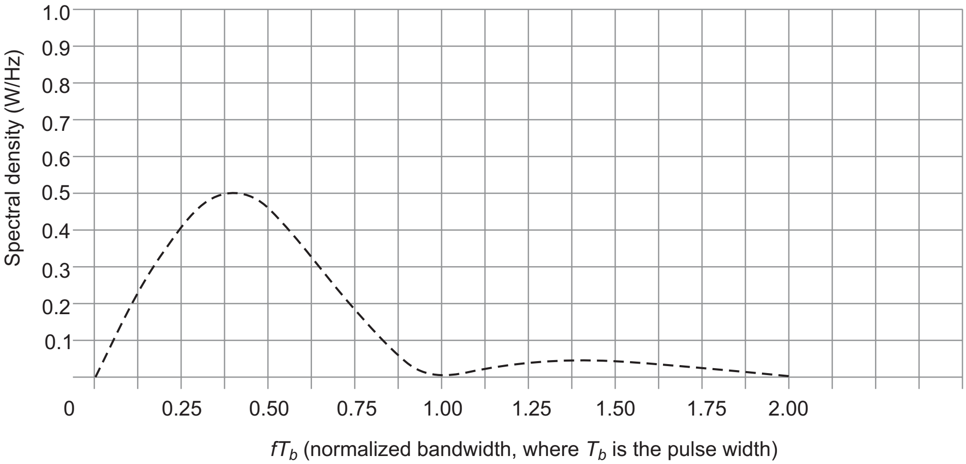


Fig. C.3 Power Spectra of Bipolar NRZ Data Format

It is observed that although most of the power lies within a bandwidth equal to the bit rate $\frac{1}{T_b}$, the spectral content is relatively small around zero frequency or dc. The spectrum of the bipolar NRZ signal has higher frequency components than are present in the unipolar or polar NRZ signal. The PSD of bipolar NRZ data format has the following properties:

1. The value of $P_X(f)$ is zero at $f = 0$
2. The main lobes extend from $f = 0$ to $f = 2f_b$.
3. The main lobes have peaks at approximately $\pm\frac{3f_b}{4}$.
4. When the bipolar NRZ signal is transmitted through an ideal low-pass filter, with cut-off frequency at $f = 2f_b$, 95% of the total power will be passed.
5. When the bipolar NRZ signal is transmitted through an ideal low-pass filter, with cut-off frequency at $f = f_b$, then only approximately 70% of the total power is passed.

C.4 PSD OF MANCHESTER ENCODING DATA FORMAT

The input binary data consists of independent and equally likely symbols. Its autocorrelation function is same as that for the polar NRZ format. Since the basic pulse $v(t)$ consists of a doublet pulse of unit magnitude and total durationT_b, its Fourier transform is given by

$$V(f) = fT_b \sin c\left(\frac{fT_b}{2}\right)\sin\left(\frac{\pi fT_b}{2}\right); \text{ where } \sin c(fT_b) = \frac{\sin(\pi fT_b)}{\pi fT_b}$$

The power spectral density of the Manchester format can be derived in the similar way and is given by

$$\boxed{P_X(f) = V^2 T_b \sin c^2\left(\frac{fT_b}{2}\right)\sin^2\left(\frac{\pi fT_b}{2}\right)}$$

This is the expression for PSD of Manchester digital data sequence.

Specifically, the power spectral density $P_X(f)$ is normalized with respect to V^2T_b, and the frequency f is normalized with respect to the bit rate $\frac{1}{T_b}$.

C.5 COMPARISON OF PSD OF DIFFERENT LINE-ENCODING FORMATS

Figure C.4 shows the power spectral density of unipolar NRZ, polar NRZ, bipolar NRZ, and Manchester line-encoding formats for comparison purpose.

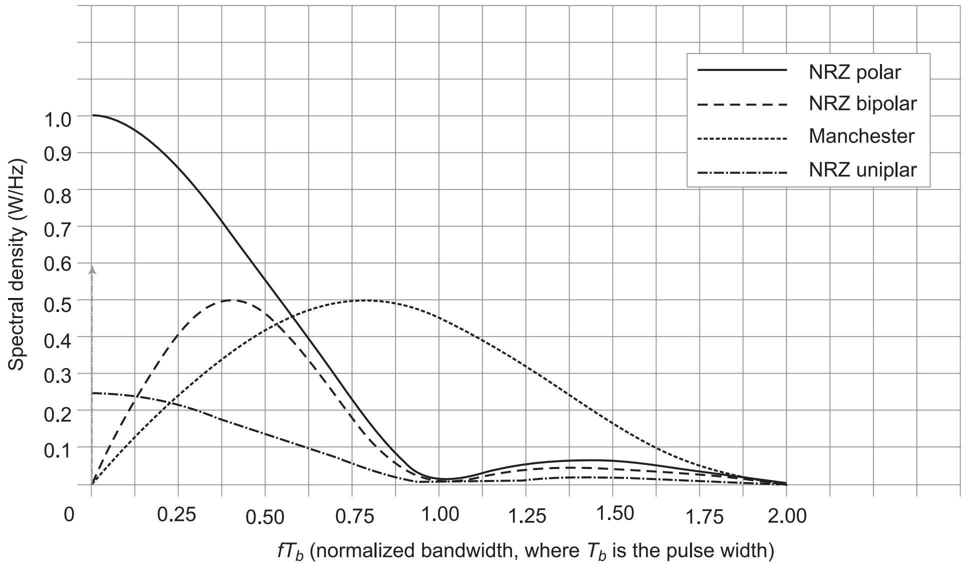


Fig. C.4 Power Spectra of different Binary Data Formats

It is observed that in case of bipolar NRZ, most of the power lies within a bandwidth equal to the bit rate $1/T_b$. The spectral content is relatively small around zero frequency in bipolar NRZ as compared to that of unipolar NRZ and polar NRZ waveforms. In Manchester encoding format, most of the power lies within a bandwidth equal to the bit rate $2/T_b$ which is twice that of unipolar NRZ, polar NRZ, and bipolar NRZ encoding formats.

APPENDIX D

Weiner Optimum Filters for Waveform Estimation

D.1 GEOMETRIC INTERPRETATION OF SIGNALS

The geometric or vector interpretation of signals greatly simplifies detection process of transmitted signals in communication system. It is desirable to have the minimum number of variables that is required to represent a signal so that lesser storage space would be needed. It implies that when the signal is transmitted, communication bandwidth is conserved. The complete set of all signals is known as a *signal space*. The collection of the minimum number of functions those are necessary to represent a given signal are called *basis functions.* These are independent and always orthogonal to each other. The collection of basis functions is called *basis set*. When the energy for all the basis functions is normalized, these are known as *orthonormal basis set*.

Geometric or vector interpretation of a signal is more convenient than the conventional waveform representation. From a geometric point of view, each basis function is mutually perpendicular to each of the other basis functions. Thus it simplifies the Euclidean distance (distance between adjacent signal points) calculations. The detection of a signal is substantially influenced by the Euclidean distance between the signal points in the signal space. Thus, in the detection process for digitally modulated signals, vector representation of a signal is more convenient than the conventional waveform representation.

Expansion in Orthogonal Functions

A complete orthogonal set is the set of sinusoidal functions (both sines and cosines terms) which generate the Fourier series. It is necessary to specify the length of the interval because of the periodicity of the functions. A periodic function $s(t)$ with period T can be expanded into a Fourier series, and is given as

$$s(t) = \frac{A_0}{\sqrt{T}} + \sum_{n=1}^{n=\infty} A_n \sqrt{\frac{2}{T}} \cos\left(\frac{2\pi nt}{T}\right) + \sum_{n=1}^{n=\infty} B_n \sqrt{\frac{2}{T}} \sin\left(\frac{2\pi nt}{T}\right)$$

where the coefficients are $A_0 = \frac{1}{\sqrt{T}} \int_T s(t)dt \quad A_n = \sqrt{\frac{2}{T}} \int_T s(t)\cos\left(\frac{2\pi nt}{T}\right)dt; n \neq 0$;

$$B_n = \sqrt{\frac{2}{T}} \int_T s(t)\sin\left(\frac{2\pi nt}{T}\right)dt$$

Thus, the orthogonal functions are $\frac{1}{\sqrt{T}}$; $\sqrt{\frac{2}{T}}\cos\left(\frac{2\pi nt}{T}\right)$, $n \neq 0$; and $\sqrt{\frac{2}{T}}\sin\left(\frac{2\pi nt}{T}\right)$.

The properties of these orthogonal functions are that any one function squared and integrated over T yields zero. When any two functions are multiplied and integrated over T yields zero. As a general case, an expansion of a periodic function $s(t)$ is valid only over the finite period T of the orthogonality. However, if it should happen that $s(t)$ is also periodic with period T, then the expansion is valid for all values of t.

D.2 GRAM–SCHMIDT (G–S) PROCEDURE

The procedure for obtaining the basis set from the original signal set is known as the *Gram-Schmidt (G–S) Orthogonalisation procedure*. If a communication system uses three non-orthogonal signal waveforms, the transmitter and the receiver are required to implement the two basis functions instead of the three original waveforms. Let a set of orthonormal basis function is to be formed a given set of finite-energy signal waveforms, represented by $\{S_k(t), k = 1, 2, \ldots. M\}$. The energy E_1 of the first finite-energy signal waveform $s_1(t)$ is given by $E_1 = \int_0^T s_1^2(t)dt$. Then, accordingly the first basis function is described as the first signal waveform $s_1(t)$ normalized to unit energy. That is,

$$\Psi_1(t) = \frac{s_1(t)}{\sqrt{E_1}}$$

The second normalized basis function $\Psi_2(t)$ is given by

$$\Psi_2(t) = \frac{s_2(t) - \left[\int_{-\infty}^{\infty} s_2(t)\Psi_1(t)\,dt\right]\Psi_1(t)}{\sqrt{E_2}}$$

Generalizing the orthonormal basis set, the k^{th} basis function is obtained as

$$\boxed{\Psi_k(t) = \frac{s_k(t) - \left[\int_{-\infty}^{\infty} s_k(t)\Psi_i(t)dt\right]\Psi_i(t)}{\sqrt{E_2}};\ i = 1, 2, \ldots. (k-1)}$$

A set of M finite-energy waveforms $\{S_k(t)\}$ can be represented by a weighted linear combination of orthonormal function $\{\Psi_n(t)\}$ of dimensionality $N \le M$. The functions $\{\Psi_n(t)\}$ are obtained by the Gram–Schmidt orthogonalisation procedure. However, the vectors $\{S_k\}$ will retain their geometric configuration. At the time of reception the decision made by the detector is based on the vector length of a received signal.

D.3 THE SAMPLING FUNCTION

The sampling function $S_a(x)$ is defined as $\boxed{S_a(x) = \frac{\sin x}{x}}$. A plot of the sampling function is given in Figure D.1

The sampling function has the following properties:

- It is symmetrical about $x = 0$.
- $S_a(x)$ has the maximum value which is unity and occurs at $x = 0$, that is $S_a(0) = 1$.
- It oscillates with amplitude that decreases with increasing x.
- It passes through 0 at equally spaced interval at values of $x = \pm n\pi$, where n is a non-zero integer.
- At $x = \pm\left[n + \frac{1}{2}\right]\pi$, where $|\sin x| = 1$, the maxima and minima occur which is approximately in the center between the zeros.

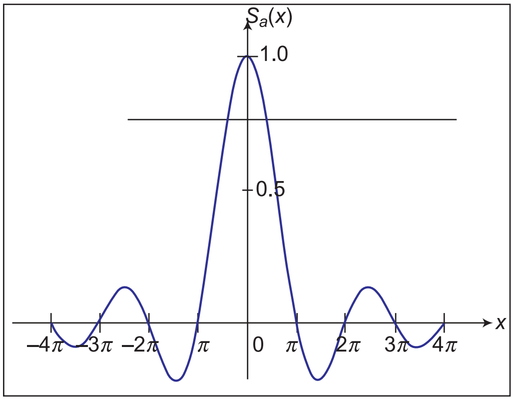


Fig. D.1 The Sampling Function

D.4 BASEBAND SIGNAL RECEIVER MODEL

The function of a receiver in a baseband binary signal system is to distinguish between two transmitted signals (corresponding to binary logic 0 and 1 respectively) in the presence of noise. Consider a receiver model which involves a linear time-invariant filter of impulse response $h(t)$, as shown in Figure D.2.

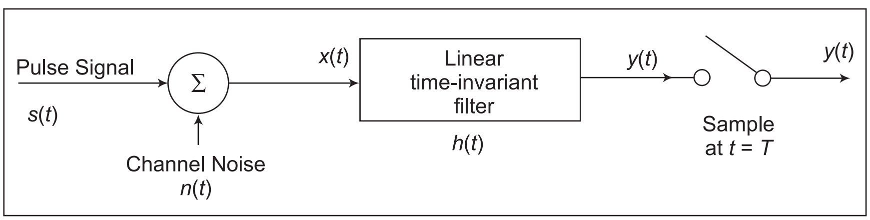


Fig. D.2 A Baseband Signal Receiver Model

The input signal to the linear time-invariant filter comprises of a pulse signal $s(t)$ corrupted by additive channel noise, $n(t)$. That is, $x(t) = s(t) + n(t)$; $0 \le t \le T$; where T is an arbitrary observation interval. In a baseband binary communication system, the pulse signal $s(t)$ may represent a binary symbol 0 or 1. The noise signal $n(t)$ is the sample function of an additive white Gaussian noise process of zero mean and power spectral density, $\frac{N_o}{2}$. Given the received signal $x(t)$, the function of the receiver is to detect the pulse signal $s(t)$ in an optimum way. The performance of the receiver is said to be optimum if it yields the minimum probability of error. It is reasonable to assume here that the receiver has *apriori* knowledge of the type of waveform of the pulse signal $s(t)$.

D.5 THE MATCHED FILTER

The *matched filter* is an optimum detector of a known pulse in the presence of additive white noise. In order to enhance the detection of the pulse signal $s(t)$, the design of the linear time-invariant filter is required to be optimized so as to minimize the effects of noise at the filter output in some statistical sense. The resulting output signal of the linear filter may be expressed as $y(t) = s_o(t) + n_o(t)$; where $s_o(t)$ is the output signal component and $n_o(t)$ is the output noise component of the input signal $x(t)$. The output *peak pulse signal-to-noise ratio*, ζ is defined as

$\zeta = \dfrac{|s_o(T)|^2}{E\left[n_o^2(t)\right]}$; where $|s_o(T)|^2$ is the instantaneous power in the output signal, E is the statistical expectation operator, and $E[n_o^2(t)]$ is the measure of the average output noise power.

It is desirable that the impulse response $h(t)$ of the matched filter should be such so as to maximize the value of peak pulse signal-to-noise ratio, ζ. We know that the Fourier transform of the output signal $s_o(t)$ is equal to $H(f)\,S(f)$, where $H(f)$ represents the frequency response of the filter, and $S(f)$ represents the Fourier transform of the known signal $s(t)$. Then $s_o(t) = \int_{-\infty}^{+\infty} H(f)S(f)e^{+j2\pi ft}df$.

At $t = T$ (T is the time at which the output of the filter is sampled),

$$|s_o(t)|^2 = \left|\int_{-\infty}^{+\infty} H(f)S(f)e^{+j2\pi ft}df\right|^2$$

This expression assumes that the channel noise is not present. The power spectral density $S_N(f)$ of the output noise $n_o(t)$ is given by $S_N(f) = \frac{N_o}{2}|H(f)|^2$; where $\frac{N_o}{2}$ is constant power spectral density of channel noise $n(t)$.

$$\Rightarrow \qquad E[n_o^2(t)] = \int_{-\infty}^{+\infty} S_N(f)df$$

$$\Rightarrow \qquad E[n_o^2(t)] = \frac{N_o}{2}\int_{-\infty}^{+\infty} |H(f)|^2 df$$

where $E[n_0^2(t)]$ denotes the average power of the output noise.

Then the peak pulse signal-to-noise ratio, ζ can be rewritten as

$$\zeta = \frac{\left|\int_{-\infty}^{+\infty} H(f)S(f)e^{+j2\pi fT}df\right|^2}{\frac{N_o}{2}\int_{-\infty}^{+\infty} |H(f)|^2 df}$$

Thus, the peak pulse signal-to-noise ratio of a matched filter depends only on the ratio of the signal energy to the power spectral density of the additive white noise at the filter input. Some important characteristics of matched filters are given as follow:

- Matched filters are specifically designed to maximize the SNR of a known signal in the presence of AWGN.
- Matched filters can be considered to be a template that is matched to the known shape of the signal being processed.
- Matched filters are applied to known signals with random parameters such as amplitude and arrival time.

- A matched filter significantly modifies the spectral structure by gathering the signal energy matched to its template, and present the result as a peak amplitude at the end of each symbol duration.
- The matched filter gathers the received signal energy, and when its output is sampled at $t = T_b$, a voltage proportional to that energy is produced for subsequent detection and post-detection processing.

D.6 PROBABILITY OF ERROR USING MATCHED FILTER

Probability of error is defined as the probability that the symbol at the output of the receiver differs from that of transmitted symbol. Consider a unipolar NRZ binary-encoded PCM signal, $s(t)$ such that when symbol 1 is sent, $s(t)$ equals $s_1(t)$ defined by

$$s_1(t) = \sqrt{\frac{E_{\max}}{T_b}};\ 0 \le t \le T_b;$$

where $E_{\max}$ is the maximum or peak signal energy, and T_b is the symbol duration. When the symbol 0 is sent, the transmitter is switched off, so $s(t)$ equals $s_0(t)$ defined by $s_0(t) = 0$; $0 \le t \le T_b$;

The channel noise $n(t)$ is modeled as additive white Gaussian noise (AWGN) with zero mean and power spectral density $\frac{N_o}{2}$. Correspondingly, the received signal, $x(t)$ equals

$$x(t) = s(t) + n(t);\ 0 \le t \le T_b;$$

where the transmitted PCM signal $s(t)$ equals either $s_1(t)$ or $s_0(t)$, depending on whether symbol 1 or 0 has been sent.

Figure D.3 depicts a receiver model for binary-encoded PCM transmission system.

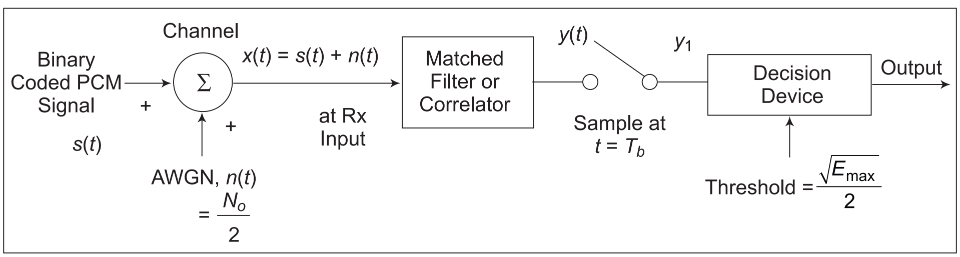


Fig. D.3 Receiver Model for Binary-encoded PCM Signal

The matched filter output is sampled at time $t = T_b$. The resulting sample value is compared with threshold by means of a decision device. If the threshold is exceeded, the receiver decides in favour of symbol 1, if not then symbol 0, and if tie (!) then it may be 1 or 0.

Since the bit duration T_b is known for a particular system, and is constant, we may define one basis function of unit energy, that is,

$$\varphi_1(t) = \sqrt{\frac{1}{T_b}}; \qquad 0 \le t \le T_b$$

$$\therefore \qquad s_1(t) = \sqrt{E_{\max}}\,\varphi_1(t); \quad 0 \le t \le T_b$$

An on-off PCM system is characterized by having a signal space that is one-dimensional and with two message points, S_{11} (corresponding to symbol 1) and S_{01} (corresponding to symbol 0), as shown in Figure D.4.

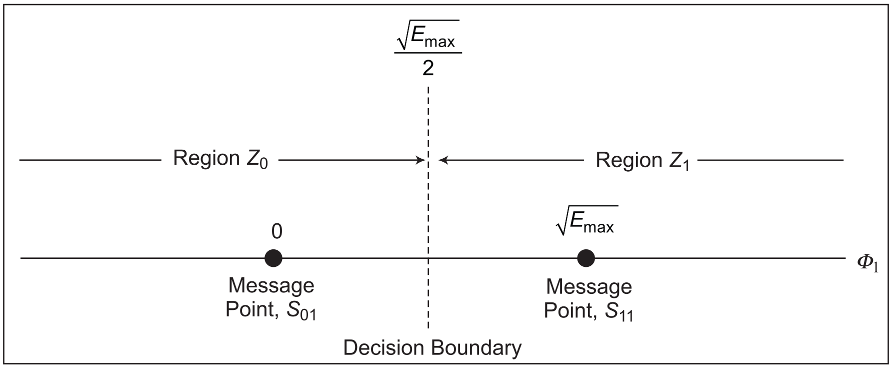


Fig. D.4 A One-dimensional Signal Space with Two Message Points

The coordinates of the two message points are given by

$$S_{11} = \int_0^{T_b} s_1(t)\varphi_1(t)dt$$

$$\Rightarrow \qquad S_{11} = \int_0^{T_b}\left[\sqrt{\frac{E_{max}}{T_b}} \times \sqrt{\frac{1}{T_b}}dt\right]$$

$$\Rightarrow \qquad S_{11} = \frac{\sqrt{E_{max}}}{T_b}\int_0^{T_b} dt = \frac{\sqrt{E_{max}}}{T_b}|t|_0^{T_b} = \frac{\sqrt{E_{max}}}{T_b} \times T_b = \sqrt{E_{max}}$$

and
$$S_{01} = \int_0^{T_b} s_0(t)\varphi_1(t)dt = \int_0^{T_b} 0 \times \sqrt{\frac{1}{T_b}}dt = 0$$

The message point corresponding to $s_1(t)$ or the symbol 1 is located at S_{11} which is equal to $\frac{\sqrt{E_{max}}}{2}$ and the message point corresponding to $s_0(t)$ or the symbol 0 is located at S_{01} which is equal to 0. It is assumed that binary symbols 1 and 0 occur with equal probability in the message sequence. Correspondingly, the threshold used by the decision device is set at $\frac{\sqrt{E_{max}}}{2}$, the halfway point or the decision boundary between two message points. To realize the decision rule as to whether symbol 1 or 0 has been sent, the one-dimensional signal space is partitioned into two decision regions—the set of points closest to the message point at $\frac{\sqrt{E_{max}}}{2}$ and the set of points closest to the second message point at 0. The corresponding decision regions are shown marked as region Z_1 and region Z_0 respectively. The decision rule is now simply to estimate signal $s_1(t)$, that is, symbol 1 would have been sent if the received signal falls in region

Z_1, and signal $s_0(t)$, that is, symbol 0 would have been sent if the received signal falls in region Z_0.

Naturally, two kinds of erroneous decisions are most likely to be made by the receiver:

1. ***Error of the first kind, $P_e(1)$*:** The symbol 1 has been sent by the transmitter but the channel noise $n(t)$ is such that the received signal points falls inside region Z_0 and so the receiver decodes it as symbol 0.
2. ***Error of the second kind, $P_e(0)$*:** The symbol 0 has been sent by the transmitter but the channel noise $n(t)$ is such that the received signal points falls inside region Z_1 and so the receiver decodes it as symbol 1.

The received signal (observation) point is calculated by sampling the matched filter output at time $t = T_b$, that is, $y_1 = \int_0^{T_b} y(t)\varphi_1(t)dt$. The received signal point y_1 may lie anywhere along the φ_1 axis. This is so because $y_1$is the sample value of a Gaussian-distributed random variable Y_1. When symbol 1 is sent, the mean of Y_1 is $\frac{\sqrt{E_{\max}}}{2}$, and when symbol 0 is sent, the mean of Y_1 is zero. Regardless of which of the two symbols is sent, the variance of Y_1 equals $\frac{N_o}{2}$, where $\frac{N_o}{2}$ is the power spectral density of the channel noise.

Mathematically, the decision region associated with symbol 1 is defined as

$$Z_1 : \frac{\sqrt{E_{\max}}}{2} < y_1 < \infty$$

Since the random variable Y_1, with sample value y_1 has a Gaussian distribution with zero mean and variance $\frac{N_o}{2}$, the likelihood function (under the assumption that symbol 0 has been sent) is defined by

$$f_{y_1}(y_1|0) = \frac{1}{\sqrt{\pi N_o}} e^{-\left(\frac{y_1^2}{N_o}\right)}$$

A plot of this function is shown in Figure D.5.

Let $P_1(0)$ denote the conditional probability of the deciding in favour of symbol 1, given that symbol 0 has been sent. The probability $P_1(0)$ is the total area under shaded part of the curve lying above $\frac{\sqrt{E_{\max}}}{2}$.

Hence, $$P_e(0) = \int_{y_1=\frac{\sqrt{E_{\max}}}{2}}^{y_1=\infty} f_{y_1}(y_1|0)dy_1 = \int_{y_1=\frac{\sqrt{E_{\max}}}{2}}^{\infty} \frac{1}{\sqrt{\pi N_o}} e^{-\left(\frac{y_1^2}{N_o}\right)} dy_1$$

$\Rightarrow$ $$P_e(0) = \frac{1}{\sqrt{\pi N_o}} \int_{y_1=\frac{\sqrt{E_{\max}}}{2}}^{\infty} e^{-\left(\frac{y_1^2}{N_o}\right)} dy_1$$

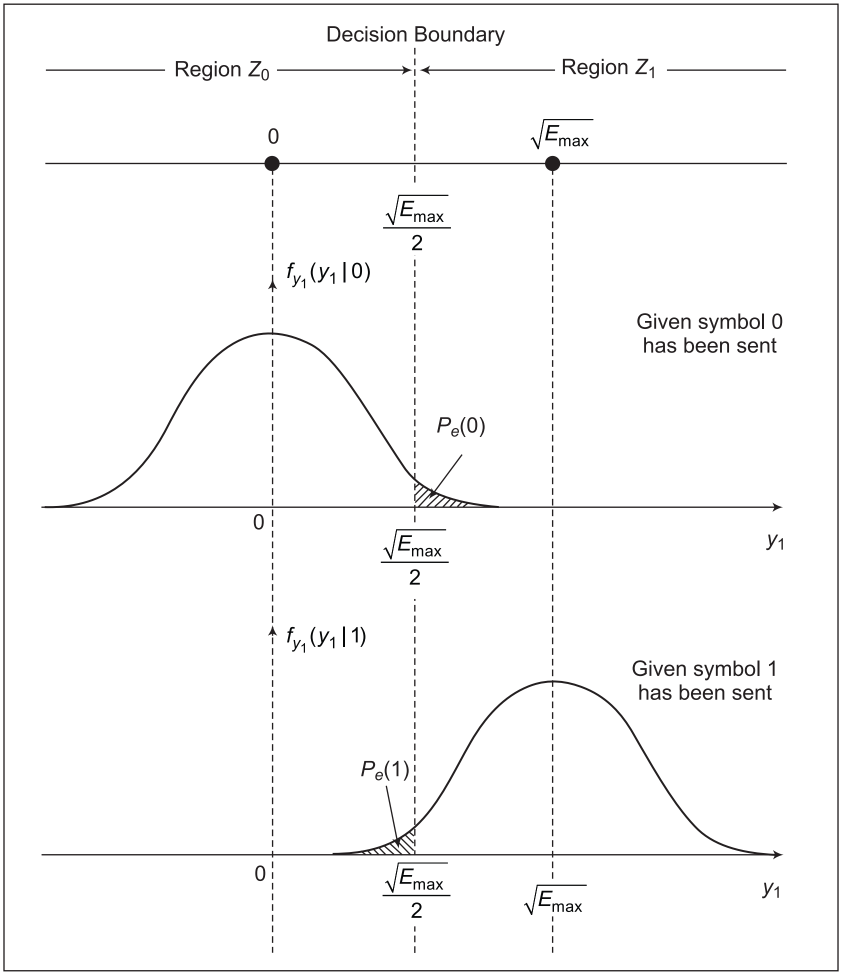


Fig. D.5 A Plot of the Likelihood Function, $f_{y1}(y_1|0)$ and $f_{y1}(y_1|1)$

Let $\frac{{y_1}^2}{N_o} = z^2$; $\Rightarrow$ $z = \frac{y_1}{\sqrt{N_o}}$

We know that $y_1 = \frac{\sqrt{E_{\max}}}{2}$;

$$\therefore \qquad z = \frac{\frac{\sqrt{E_{\max}}}{2}}{\sqrt{N_o}} = \frac{1}{2}\frac{\sqrt{E_{\max}}}{\sqrt{N_o}} = \frac{1}{2}\sqrt{\frac{E_{\max}}{N_o}}$$

Also, $y_1 = z\sqrt{N_o}$; $\Rightarrow dy_1 = \sqrt{N_o}\,dz$

Substituting these values, we get

$$P_e(0) = \frac{1}{\sqrt{\pi N_o}} \int_{z=\frac{1}{2}\sqrt{\frac{E_{max}}{N_o}}}^{\infty} e^{-z^2} \sqrt{N_o}\, dz$$

$$\Rightarrow \qquad P_e(0) = \frac{1}{\sqrt{\pi}} \int_{z=\frac{1}{2}\sqrt{\frac{E_{max}}{N_o}}}^{\infty} e^{-z^2}\, dz$$

By definition, the complementary error function is given by

$$erfc(u) = \frac{2}{\sqrt{\pi}} \int_{z=u}^{\infty} e^{-z^2}\, dz$$

Accordingly, $P_e(0)$ may be expressed as $\boxed{P_e(0) = \frac{1}{2} erfc\left(\frac{1}{2}\sqrt{\frac{E_{max}}{N_o}}\right)}$

This is the expression for probability of error for estimating transmitted symbol 0 as symbol 1 by the receiver.

Similarly, the decision region associated with the symbol 0 is defined by

$$Z_0 : -\infty < y_1 < \frac{\sqrt{E_{max}}}{2}$$

The plot of corresponding likelihood function $f_{y1}(y_1|1)$ is shown in the same figure for easy understanding.

Due to symmetric nature of memoryless binary channel, $P_1(1) = P_e(0)$. Therefore,

$$\boxed{P_e(1) = \frac{1}{2} erfc\left(\frac{1}{2}\sqrt{\frac{E_{max}}{N_o}}\right)}$$

To determine the average probability of error in the receiver, P_e, the two possible kinds of error are mutually exclusive events in that if the receiver chooses symbol 1, then symbol 0 is excluded from appearing and likewise if the receiver chooses symbol 0, then symbol 1 is excluded from appearing at the output. Remember that $P_e(0)$ and $P_e(1)$ are conditional probabilities.

Thus, assuming that the *apriori* probability of sending a 0 is p_0, and apriori probability of sending a 1 is p_1, then the average probability of error in the receiver is given by

$$P_e = p_0\, P_e(0) + p_1\, P_e(1)$$

Since $P_e\,(1) = P_e(0)$, and $p_0 + p_1 = 1$, we get

$$P_e = P_0\, P_e(0) + (1 - p_0)\, P_e(0)$$

$$\Rightarrow \qquad P_e = P_0\, P_e(0) + P_e\,(0) - p_0\, P_e(0)$$

$$\Rightarrow \qquad P_e = P_e\,(0) = P_e(1)$$

Hence, $$\boxed{P_e = \frac{1}{2} erfc\left(\frac{1}{2}\sqrt{\frac{E_{max}}{N_o}}\right)}$$

Generally, the acceptable value of probability of error is specified for a particular application. Table D.1 gives $\frac{E_{\max}}{N_o}$ values for certain given values of P_e.

Table D.1 $E_{\max}/N_o$(dB) for specified P_e

P_e	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-20}
$\frac{E_{\max}}{N_o}$, dB	10.3	14.4	16.6	18	19	20

The ratio $\frac{E_{\max}}{N_o}$ represents the peak signal energy-to-noise spectral density ratio.

If $P_{\max}$ is the peak signal power and T_b is the bit duration then peak signal energy, $E_{\max}$ may be written as $E_{\max} = P_{\max} T_b$. Therefore,

$$\frac{E_{\max}}{N_o} = \frac{P_{\max} T_b}{N_o} = \frac{P_{\max}}{N_o/T_b}$$

The ratio N_o/T_b may be viewed as the average noise power contained in a transmission bandwidth equal to the bit rate $1/T_b$. Thus, $\frac{E_{\max}}{N_o}$ is peak signal energy-to-noise power ratio. The Intersymbol interference generated by the filter is assumed to be small. The filter is matched to a rectangular pulse of amplitude A and duration T_b, since the bit-timing information is available to the receiver. Thus, the average probability of symbol error in a binary symmetric channel (for equiprobable binary symbols 1 and 0) solely depends on the ratio of the peak signal energy to the noise spectral density, $\frac{E_b}{N_o}$ measured at the receiver input. However, it is assumed here that the receiver has prior knowledge of the pulse shape, but not in polarity.

It is observed that P_e decreases very rapidly as $\frac{E_{\max}}{N_o}$ increases. The PCM receiver exhibits an exponential improvement in the average probability of symbol error with increase in $\frac{E_b}{N_o}$ value. At about $\frac{E_{\max}}{N_o} \approx 17\text{ dB}$, an error threshold occurs. The receiver performance may involve significant number of errors on the order of $>10^{-7}$ below threshold value. The effect of channel noise on the receiver performance is practically negligible above threshold value. For a practical PCM system, $P_e \leq 10^{-5}$ which corresponds to $\frac{E_{\max}}{N_o} > 15\text{ dB}$.

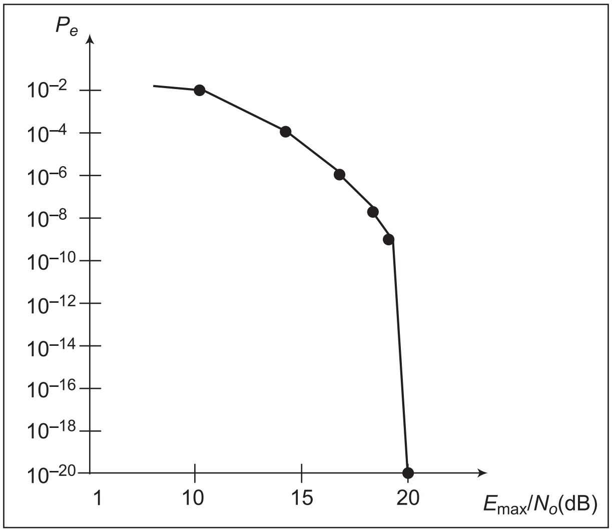


Fig. D.6 Plot of E_{max}/N_o versus P_e

D.7 WIENER–HOPF FILTER

The *Wiener–Hopf filter*, also known as the *optimum filter*, has the ability of distinguishing between the two known signal waveforms from their noisy versions as which one of the two was present at its input during each signaling interval with minimum probability of error. The optimum filter takes the form of a matched filter when the noise at its input is white noise. In fact, the optimum filter for the white noise generates zero Intersymbol interference. Using the Schwarz's inequality, we can write

$$\left|\int_{-\infty}^{+\infty} H(f)\,S(f)\,e^{+j2\pi fT}\,df\right|^2 \le \int_{-\infty}^{+\infty} |H(f)|^2\,df \int_{-\infty}^{+\infty} |S(f)|^2\,df$$

$$\therefore \qquad \zeta \le \frac{2}{N_0}\int_{-\infty}^{+\infty} |S(f)|^2\,df$$

Thus, ζ depends only on the signal energy and the noise power spectral density. It is independent of the frequency response $H(f)$ of the filter. Therefore, its maximum value if given by $\zeta\big|_{\max} = \dfrac{2}{N_o}\displaystyle\int_{-\infty}^{+\infty} |S(f)|^2\,df$. In that case, $H(f)$ assumes its optimum value, denoted by $H_{opt}(f)$, and is given by $H_{opt}(f) = kS^*(f)e^{-j2\pi fT}$; where $S^*(f)$ is the complex conjugate of the Fourier transform of the input signal $s(t)$, and k is an appropriate scaling factor. This expression specifies the optimum filter in the frequency domain. In time domain, the impulse response of the optimum filter can be described as $h_{opt}(t) = ks(T - t)$. It is time-reversed and time-delayed version of the input signal which implies that it is optimally matched to the input signal.

In a communication receiver, the signal PSD at the input of the receiving filter is $S_s(f) = \dfrac{2\gamma}{\gamma^2 + (2\pi f)^2}$, and the noise PSD appearing at its input is $S_n(f) = \dfrac{N_o}{2}$. The transfer function of the optimum filter is given by $H_{\text{opt}}(f) = \dfrac{S_s(f)}{S_s(f) + S_n(f)}$.

$$\Rightarrow \qquad h_{\text{opt}}(f) = \frac{\dfrac{2\gamma}{\gamma^2 + (2\pi f)^2}}{\dfrac{2\gamma}{\gamma^2 + (2\pi f)^2} + \dfrac{N_o}{2}} = \frac{4\gamma}{4\gamma + N_o\left[\gamma^2 + (2\pi f)^2\right]}$$

$$\Rightarrow \qquad h_{\text{opt}}(f) = \frac{4\gamma}{N_o\left[\dfrac{4\gamma}{N_o} + \gamma^2 + (2\pi f)^2\right]}$$

Let $\dfrac{4\gamma}{N_o} + \gamma^2 = \lambda^2$, then $H_{\text{opt}}(f) = \dfrac{4\gamma}{N_o\left[\lambda^2 + (2\pi f)^2\right]}$.

Using Inverse Laplace transform, we get

$$h_{\text{opt}}(t) = \frac{2\gamma}{N_o \lambda} e^{-\lambda|t|}; \text{ where } \lambda = \sqrt{\frac{4\gamma}{N_o} + \gamma^2}$$

Figure D.7 shows the transfer function of an optimum filter.

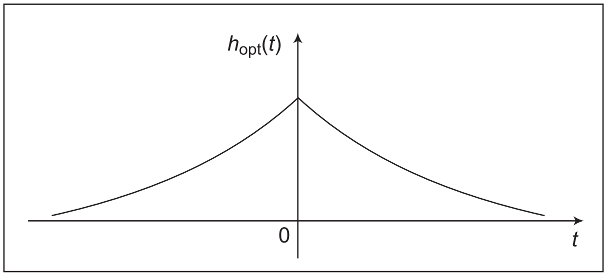


Fig. D.7 Transfer Function of Optimum Filter

It is an unrealizable filter. However, a time-shifted version $h_{opt}(t - t_0)u(t)$ is nearly realizable filter. Figure D.8 shows the transfer function of realizable optimum filter.

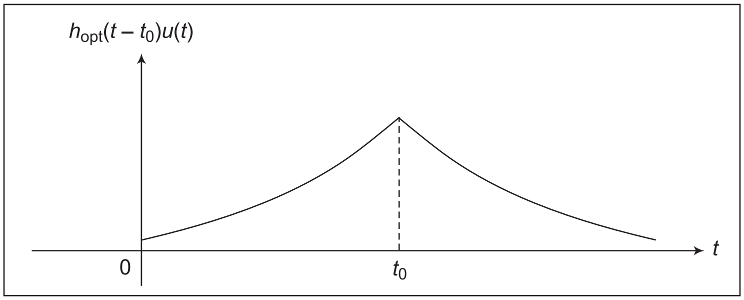


Fig. D.8 Transfer function of realizable optimum filter

The output noise power P_n of the optimum filter is given by

$$P_n = \int_{-\infty}^{\infty} \frac{S_s(f)S_n(f)}{S_s(f)+S_n(f)} df$$

$$\Rightarrow \qquad P_n = \int_{-\infty}^{\infty} \frac{\left[\dfrac{2\gamma}{\gamma^2+(2\pi f)^2}\right]\times\left[\dfrac{N_o}{2}\right]}{\left[\dfrac{2\gamma}{\gamma^2+(2\pi f)^2}\right]+\left[\dfrac{N_o}{2}\right]} df$$

$$\Rightarrow \qquad P_n = \int_{-\infty}^{\infty} \frac{2\gamma N_o}{4\gamma + N_o\left[\gamma^2+(2\pi f)^2\right]} df$$

$$\Rightarrow \qquad P_n = \int_{-\infty}^{\infty} \frac{2\gamma N_o}{N_o\left[\dfrac{4\gamma}{N_o}+\gamma^2+(2\pi f)^2\right]} df = \int_{-\infty}^{\infty} \frac{2\gamma}{\dfrac{4\gamma}{N_o}+\gamma^2+(2\pi f)^2} df$$

To solve it, let $\dfrac{4\gamma}{N_o}+\gamma^2=\alpha^2$; then,

$$\Rightarrow \qquad P_n = \int_{-\infty}^{\infty} \frac{2\gamma}{\alpha^2+(2\pi f)^2} df = \frac{2\gamma}{\alpha^2}\int_{-\infty}^{\infty} \frac{1}{1+\left(\dfrac{2\pi f}{\alpha}\right)^2} df$$

Let $\dfrac{2\pi f}{\alpha} = x$; $\quad \Rightarrow f = \dfrac{\alpha x}{2\pi}$; $df = \dfrac{\alpha}{2\pi}dx$

$$\Rightarrow \qquad P_n = \frac{2\gamma}{\alpha^2}\int_{-\infty}^{\infty} \frac{1}{1+x^2}\frac{\alpha}{2\pi} dx$$

$$\Rightarrow \qquad P_n = \frac{2\gamma}{\alpha^2}\frac{\alpha}{2\pi}\int_{-\infty}^{\infty} \frac{1}{1+x^2} dx = \frac{\gamma}{\pi\alpha}\int_{-\infty}^{\infty} \frac{1}{1+x^2} dx$$

$$\Rightarrow \qquad P_n = \frac{\gamma}{\pi\alpha}\left|\tan x\right|_{-\infty}^{\infty} = \frac{\gamma}{\pi\alpha}\left[\tan\infty - \tan(-\infty)\right]$$

$$\Rightarrow \qquad P_n = \frac{\gamma}{\pi\alpha}\left[\frac{\pi}{2}-\left(-\frac{\pi}{2}\right)\right] = \frac{\gamma}{\pi\alpha}\times\pi = \frac{\gamma}{\alpha}$$

Hence, $$P_n = \frac{\gamma}{\sqrt{\dfrac{4\gamma}{N_o}+\gamma^2}}$$ **Ans.**

The optimum filter is practical only when the input SNR is small (that is, noise is quite high). When a desired signal gets mixed with noise, the SNR can be improved by passing it through a filter that suppresses frequency components where the signal is weak but the noise is strong. An optimum filter is not an ideal filter. Thus, it will cause signal distortion.

D.8 INTEGRATE-AND-DUMP FILTER

The *integrate-and-dump filter*, also known as *correlation receiver*, is a coherent or synchronous receiver that requires a local carrier reference signal having the same frequency and phase as the transmitted carrier signal. It is a form of the optimum filter, which is different from the matched filter implementation. Additional circuitry is required at the receiver to generate the coherent local carrier reference signal. In a correlation receiver, it is required that the integration operation should be ideal with zero initial conditions. In practical implementation of the correlation receiver, the integrator has to be reset. That means the capacitor has to be discharged or dumped (hence the name integrate and dump filter) at the end of each signaling interval in order to avoid ISI. The bandwidth of the filter preceding the integrator is assumed to be wide enough to pass the received signal accompanied with white Gaussian noise without any distortion.

Integrate-and-dump filter can be used to implement the matched filter for a rectangular pulse input signal. The integrator computes the area under the rectangular pulse. The output is then sampled at $t = T$ (duration of the pulse). The integrator is restored back to its initial condition immediately. It is essential that the time constant of the integrate-and-dump filter circuit must be very much greater than the pulse width. Under this condition, the practical circuit of the integrate and dump correlation receiver will approximate quite close to an ideal integrator. It will also operate as an ideal receiver with the same probability of error. The correlation receiver performs coherent detection by using the local reference signal which is in phase with the input received signal component. Therefore, the sampling and discharging of the capacitor (dumping action) must be carefully synchronized.

APPENDIX E

Mathematical Formulae

Table E.1 Trigonometric Identities

S. No.	Identity
1.	$\sin 2\theta = 2 \sin \theta \cos \theta$
2.	$\cos 2\theta = 1 + 2 \cos^2 \theta = 1 - 2 \sin^2 \theta = \cos^2 \theta - \sin^2 \theta$
3.	$\sin^2 \theta + \cos^2 \theta = 1$
4.	$\sin(\theta_1 \pm \theta_2) = \sin \theta_1 \cos \theta_2 \pm \cos \theta_1 \sin \theta_2$
5.	$\cos(\theta_1 \pm \theta_2) = \cos \theta_1 \cos \theta_2 \mp \sin \theta_1 \sin \theta_2$
6.	$\sin\theta_1 \sin\theta_2 = \frac{1}{2}[\cos(\theta_1 - \theta_2) - \cos(\theta_1 + \theta_2)]$
7.	$\cos\theta_1 \cos\theta_2 = \frac{1}{2}[\cos(\theta_1 - \theta_2) + \cos(\theta_1 + \theta_2)]$
8.	$\sin\theta_1 \cos\theta_2 = \frac{1}{2}[\sin(\theta_1 - \theta_2) + \cos(\theta_1 + \theta_2)]$
9.	$e^{\pm j\theta} = \cos \theta \pm j \sin \theta$
10.	$\sin\theta = \frac{1}{2j}[e^{j\theta} - e^{-j\theta}]$
11.	$\cos\theta = \frac{1}{2}[e^{j\theta} + e^{-j\theta}]$

Table E.2 Series Expansion

S. No.	Series	Formulae
1.	Trigono-metric Series	$\sin x = x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \cdots$
		$\sin^{-1} x = x + \frac{1}{6}x^3 + \frac{3}{40}x^5 + \cdots$
		$\sin cx = 1 - \frac{1}{3!}(nx)^2 + \frac{1}{5!}(\pi x)^5 - \cdots$

S. No.	Series	Formulae	
		$\cos x = 1 - \frac{1}{2!}x^2 + \frac{1}{4!}x^4 - \cdots$	
2.	Exponential Series	$e^x = 1 + x + \frac{1}{2!}x^2 - \cdots$	
3.	Binomial Series	$(1+x)^n = 1 + nx + \frac{n(n-1)}{2!}x^2 + \cdots;$	for $\lvert nx \rvert < 1$
		$\frac{1}{1-x} = 1 + x + x^2 + x^3 + \ldots;$	for $\lvert x \rvert < 1$
4.	Logarithmic Series	$\ln(1+x) = x - \frac{1}{2}x^2 + \frac{1}{3}x^3 - \cdots$	
5.	Tayler Series	$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{1}{n!}\left.\frac{d^n g(x)}{dx^n}\right\rvert_{x=a}(x-a)^n$	
6.	McLaurin Series	$f(x) = f(0) + \frac{f(0)}{1!}x + \frac{f'(0)}{2!}x^2 + \cdots + \frac{1}{n!}\left.\frac{d^n f(x)}{dx^n}\right\rvert_{x=0}$	

Table E.3 Definite Integrals

S. No.	Formulae
1.	$\int_0^\infty \sin cx\, dx = \int_0^\infty \sin c^2 x\, dx = \frac{1}{2}$
2.	$\int_0^\infty e^{-ax^2} dx = \frac{1}{2}\sqrt{\frac{\pi}{a}}, a > 0$
3.	$\int_0^\infty x^2 e^{-ax^2} dx = \frac{1}{4a}\sqrt{\frac{\pi}{a}}, a > 0$
4.	$\int_0^\infty \frac{x \sin(ax)}{b^2 + x^2} dx = \frac{\pi}{2} e^{-ab}, a > 0, b > 0$
5.	$\int_0^\infty \frac{\cos(ax)}{b^2 + x^2} dx = \frac{\pi}{2b} e^{-ab}, a > 0, b > 0$

APPENDIX F

Abbreviations and Acronyms

ADC	Analog-to-Digital Converter
ADM	Adaptive Delta Modulation
ADPCM	Adaptive Differential Pulse Code Modulation
AMI	Alternate Mark Inversion
ASK	Amplitude Shift Keying
AWGN	Additive White Gaussian Noise
B3ZS	Binary Three Zero Substitution
B6ZS	Binary Six Zero Substitution
B8ZS	Binary Eight Zero Substitution
BASK	Binary Amplitude Shift Keying
BCH	Bose-Chaudhuri-Hocquenhem
BEC	Binary Erasure Channel
BER	Bit Error Rate
BFSK	Binary Frequency Shift Keying
BPF	Band Pass Filter
bpp	bits per pixel
BPRZ	bipolar return-to-zero
BP-RZ-AMI	Bipolar Return-to-Zero Alternate-Mark-Inversion
bps	bits per second
BPSK	Binary Phase Shift Keying
BSC	Binary Symmetric Channel
codec	coder-decoder
CPFSK	Continuous-Phase Frequency Shift Keying
CPM	Continuous Phase Modulation
CRC	Cyclic Redundancy Check
CVSDM	Continuous Variable Slope Delta Modulation
DAC	Digital-to-Analog Converter
dB	deciBel
dBm	DeciBel w.r.t. 1 mW

DBPSK	Differential Binary Phase-Shift Keying
DM	Delta Modulation
DPCM	Differential Pulse Code Modulation
DPSK	Differential Phase Shift Keying
DQPSK	Differential Quadrature Phase Shift Keying
DS	Direct Sequence
DSSS	Direct Sequence Spread Spectrum
E_b/N_0	bit-energy to noise-power ratio
FDD	Frequency Division Duplexing
FDM	Frequency Division Multiplexing
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FH	Frequency Hopping
FHSS	Frequency Hopping Spread Spectrum
FSK	Frequency Shift Keying
GMSK	Gaussian Minimum Shift Keying
HDB	High Density Bipolar
ISI	Intersymbol interference
ITU	International Telecommunications Union
LDPC	Low-Density Parity Check
LFSR	Linear Feedback Shift Register
LPF	Low Pass Filter
LRC	Longitudinal Redundancy Check
LZ coding	Lempel–Ziv coding
MAI	Multiple Access Interference
MC-DS-CDMA	Multi-Carrier Direct Sequence Code Division Multiple Access
MFSK	Multilevel Frequency Shift Keying or *M*-ary FSK
Modem	MODulator + DEModulator
MPSK	Multilevel Phase Shift Keying or *M*-ary PSK
MSK	Minimum Shift Keying
MUs	Mobile Users
MUX	Multiplexer
NF	Noise Figure
NRZ	Non-Return-to-Zero
OFDM	Orthogonal Frequency Division Multiplexing

OFDMA	Orthogonal Frequency Division Multiple Access
OOK	On-Off Keying
OQPSK	Offset QPSK (See QPSK)
PAM	Pulse Amplitude Modulation
PCM	Pulse Code Modulation
pdf	probability density function
PLL	Phase Locked Loop
PN	Pseudo-Noise
PSD	Power Spectral Density
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
QASK	Quadrature Amplitude Shift Keying
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RS	Reed–Solomen
RZ	Return-to-Zero
S/N	Signal-to-Noise ratio
SINR	Signal-to-Interference-plus-Noise Ratio
SIR	Signal-to-Interference Ratio
SNR	Signal-to-Noise Ratio
SQR	Signal-to-Quantization Ratio
SRs	Shift Registers
SS	Spread Spectrum
TDM	Time-Division Multiplexing
THSS	Time Hopping Spread Spectrum
UP	UniPolar
UP-NRZ	UniPolar Non-Return-to-Zero
UP-RZ	UniPolar Return-to-Zero
VRC	Vertical Redundancy Check

Index

C

F

G

H

I

T

U

V

W

Z