



Digital Techniques for Wideband Receivers

3rd Edition

James Tsui and Chi-Hao Cheng

Digital Techniques for Wideband Receivers

Related titles on electromagnetic waves:

Small Signal Microwave Amplifier Design Grosch
Small Signal Microwave Amplifier Design: Solutions Grosch
2008+ Solved Problems in Electromagnetics Nasar
Antennas: Fundamentals, design, measurement, 3rd Edition Blake & Long
Designing Electronic Systems for EMC Duff
Electromagnetic Measurements in the Near Field, 2nd Edition Bienkowski & Trzaska
Fundamentals of Electromagnetics with MATLAB®, 2nd Edition Lonngren *et al.*
Fundamentals of Wave Phenomena, 2nd Edition Hirose & Lonngren
Integral Equation Methods for Electromagnetics Volakis & Sertel
Introduction to Adaptive Arrays, 2nd Edition Monzingo *et al.*
Microstrip and Printed Antenna Design, 2nd Edition Bancroft
Numerical Methods for Engineering: An introduction using MATLAB® and computational electromagnetics Warnick
Return of the Ether Deutsch
The Finite Difference Time Domain Method for Electromagnetics: with MATLAB® simulations Elsherbeni & Demir
Theory of Edge Diffraction in Electromagnetics Ufimtsev
Scattering of Wedges and Cones with Impedance Boundary Conditions Lyalinov & Zhu
Circuit Modeling for Electromagnetic Compatibility Darney
The Wiener-Hopf Method in Electromagnetics Daniele & Zich
Microwave and RF Design: A systems approach, 2nd Edition Steer
Spectrum and Network Measurements, 2nd Edition Witte
EMI Troubleshooting Cookbook for Product Designers Andre & Wyatt
Transmission Line Transformers Raymond Mack and Jerry Sevic
Electromagnetic Field Standards and Exposure Systems Grudzinski & Trzaska
Practical Communication Theory, 2nd Edition Adamy
Complex Space Source Theory of Spatially Localized Electromagnetic Waves Seshadri
Electromagnetic Compatibility Pocket Guide: Key EMC Facts, Equations and Data Wyatt & Jost
Antenna Analysis and Design Using FEKO Electromagnetic Simulation Software Elsherbeni, Nayeri & Reddy

Related titles on radar:

Microwave Receivers with Electronic Warfare Applications Tsui
Phased-Array Radar Design: Application of radar fundamentals Jeffrey
Pocket Radar Guide: Key facts, equations, and data Curry
Principles of Modern Radar, Volume 1: Basic principles Richards, Scheer & Holm
Principles of Modern Radar, Volume 2: Advanced techniques Melvin & Scheer
Principles of Modern Radar, Volume 3: Applications Scheer & Melvin
Principles of Waveform Diversity and Design Wicks *et al.*
Pulse Doppler Radar Alabaster
Radar Cross Section Measurements Knott
Radar Cross Section, 2nd Edition Knott *et al.*
Radar Design Principles: Signal processing and the environment, 2nd Edition Nathanson *et al.*
Radar Detection DiFranco & Ruby
Radar Essentials: A concise handbook for radar design and performance Curry
Radar Foundations for Imaging and Advanced Concepts Sullivan
Radar Principles for the Non-Specialist, 3rd Edition Toomay & Hannan
Test and Evaluation of Aircraft Avionics and Weapons Systems McShea
Understanding Radar Systems Kingsley & Quegan
Understanding Synthetic Aperture Radar Images Oliver & Quegan
Radar and Electronic Warfare Principles for the Non-specialist, 4th Edition Hannen
Inverse Synthetic Aperture Radar Imaging: Principles, algorithms and applications Chen & Marotella
Stimson's Introduction to Airborne Radar, 3rd Edition Baker, Griffiths & Adamy
Test and Evaluation of Avionics and Weapon Systems, 2nd Edition McShea
Angle-of-Arrival Estimation Using Radar Interferometry: Methods and applications Holder

Digital Techniques for Wideband Receivers

3rd Edition

James Tsui

US Air Force Research Laboratory, retired

Chi-Hao Cheng

Department of Electrical and Computer Engineering
Miami University



theiet.org



Published by SciTech Publishing, an imprint of the IET
www.scitechpub.com
www.theiet.org

© Artech House, Inc. 1995, 2001
© SciTech Publishing, Inc. 2004, 2016

First published 1995 (0-89006-808-9)
Second edition 2001 (1-58053-299-3)
Corrected reprint 2004 (1-891121-26-X)
Third edition 2015 (978-1-61353-217-1)

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600, or on the web at copyright.com. Requests to the Publisher for permission should be addressed to The Institution of Engineering and Technology, Michael Faraday House, Six Hills Way, Stevenage, Herts, SG1 2AY, United Kingdom.

While the authors and publisher believe that the information and guidance given in this work are correct, all parties must rely upon their own skill and judgement when making use of them. Neither the authors nor publisher assumes any liability to anyone for any loss or damage caused by any error or omission in the work, whether such an error or omission is the result of negligence or any other cause. Any and all such liability is disclaimed.

978-1-61353-217-1 (hardback)
978-1-61353-218-8 (pdf)

Typeset in India by MPS Limited
Printed in the UK by CPI Antony Rowe Ltd

Contents

Preface	xix
Acknowledgements	xxi
Abbreviations	xxiii
1 Introduction	1
1.1 Wideband Systems	1
1.2 Digital Approach	1
1.3 Obstacles in EW Receiver Development	3
1.4 Organization of the Book	3
1.5 Specific Remarks	5
References	5
2 Requirements and Characteristics of Electronic Warfare Receivers	7
2.1 Introduction	7
2.2 Introduction to Electronic Warfare	7
2.3 Difference between Intercept and Communication Receivers	8
2.4 Signal Environment for EW Receivers	10
2.5 Requirements of EW Receivers	11
2.6 Parameters Measured by an EW Receiver	12
2.7 Frequency Information	13
2.8 AOA Information	15
2.9 Outputs of an EW Receiver	16
2.10 Overview of Analog EW Receivers	17
2.11 IFM Receivers	18

2.12	Channelized Receivers	19
2.13	Bragg Cell Receivers	20
2.14	Compressive (Microscan) Receivers	21
2.15	Digital Receivers	22
2.16	Characteristics and Performance of EW Receivers	24
2.17	Potential Trend in EW Receiver Development	25
2.18	EW Processor	27
2.19	EW Receiver Design Goals	30
2.20	Summary	31
	References	31
3	Fourier Transform	35
3.1	Introduction	35
3.2	Fourier Series and Continuous Fourier Transform	35
3.3	Useful Functions	37
3.4	Properties of the Fourier Transform	42
3.5	Examples	49
3.6	DFT	57
3.7	Signal Digitization	57
3.8	Derivation of the DFT	58
3.9	More Discussions on the DFT	62
3.10	Window Functions	69
3.11	FFT	73
3.12	Possible Advantages of DFT over FFT in Receiver Applications	78
3.13	Summary	80
	References	81
4	Fourier Transform-Related Operations	83
4.1	Introduction	83
4.2	Periodogram	83
4.3	Zero Padding	86
4.4	Peak Position Estimation for Different Windows	88
4.5	Peak Position Estimation Through Iteration	92
4.6	Actual Frequency Determination by FFT	94

4.7	Autocorrelation	95
4.8	Autocorrelation (Blackman–Tukey) Spectrum Estimation	98
4.9	Application of FFT to Spectrum Estimation from the Autocorrelation Function	99
4.10	Basic Idea of a Sub-Nyquist Sampling Scheme	101
4.11	Phase Relation in a Sub-Nyquist Sampling System	104
4.12	Problems and Potential Solutions in the Sub-Nyquist Sampling Scheme	107
4.13	DFT Through Decimation	110
4.14	Applications of the Decimation Method in EW Receivers	111
4.15	Simplified Decimation Method	113
	References	115
5	Analog-to-Digital Converters, Amplifiers, and Their Interface	117
5.1	Introduction	117
5.2	Key Component Selection	117
5.3	Comparison of Analog and Digital Receiver Sensitivity	118
5.4	Basic Sample and Hold Circuit	119
5.5	Basic ADC Performance and Input Bandwidth	120
5.6	Apparent Maximum and Minimum Signals to an ADC	122
5.7	Quantization Noise of an Ideal ADC	123
5.8	Noise Level Determined by Processing Bandwidth and Dithering Effect	124
5.9	Spurious Response	125
5.10	Analysis on Spur Amplitudes	126
5.11	Further Discussion of Spur Amplitudes	130
5.12	Noise Effects in ADCs	134
5.13	Sampling Window Jittering Effect	136
5.14	Requirements of ADCs	139
5.15	Notation	139
5.16	Noise Figure and Third-Order Intercept Point	141
5.17	Characteristics of the Amplifiers in Cascade	143
5.18	ADC	147
5.19	Noise Figure of an Amplifier and ADC Combination	148

5.20	Amplifier and ADC Interface	149
5.21	The Meaning of M and M'	150
5.22	Computer Program and Results	151
5.23	Design Example	153
5.24	Experimental Results	154
	References	159
	Appendix 5.A	161
	Appendix 5.B	162
	Appendix 5.C	163
	Appendix 5.D	164
	Appendix 5.E	165
6	Frequency Downconverters	167
6.1	Introduction	167
6.2	Baseband Receiver Frequency Selection	168
6.3	Frequency Conversion	168
6.4	In-Phase (I) and Quadrature-Phase (Q) Channel Conversion	171
6.5	Imbalance in I and Q Channels	172
6.6	Analog I and Q Downconverters	175
6.7	Digital Approach to Generate I and Q Channels	178
6.8	Hilbert Transform	179
6.9	Discrete Hilbert Transform	181
6.10	Examples of the Discrete Hilbert Transform	184
6.11	Narrowband I and Q Channels Through a Special Sampling Scheme	185
6.12	Wideband I and Q Channels Through a Special Sampling Scheme	187
6.13	Hardware Considerations for Filter Design for Wideband Digital I - Q Channels	189
6.14	Digital Correction of I and Q Channel Imbalance	191
6.15	Wideband Digital Corrections of I and Q Channel Imbalance	193
	References	197
	Appendix 6.A	198
7	Sensitivity and Detection Problems	201
7.1	Introduction	201
7.2	Electronic Warfare (EW) Receiver Detection Approach	201

7.3	Potential Detection Advantage in a Digital EW Receiver	202
7.4	False Alarm Time and Probability of False Alarm for One Data Sample	204
7.5	Threshold Setting for One Data Sample	205
7.6	Probability of Detection for Single-Sample Detection	207
7.7	Detection Based on Multiple Data Samples	208
7.8	Detection Scheme for Multiple Samples (L -out-of- N)	208
7.9	Probability Density Function and Characteristic Function	209
7.10	Probability Density Function of Sum Samples with a Square Law Detector	210
7.11	Detection of Multiple Samples Based on Summation	212
7.12	An Example of Single-Sample Detection	213
7.13	An Example of Multiple-Sample (L -out-of- N) Detection	216
7.14	Selection of Threshold Level	218
7.15	Optimizing Selection of the Threshold	220
7.16	An Example of N Sample Detection (Summation Method)	221
7.17	Introduction to Frequency Domain Detection	222
7.18	A Suggested Approach to Frequency Domain Detection	224
7.19	Probability of False Alarm in the Frequency Domain	225
7.20	Input Signal Conditions in Frequency Domain Detection	226
7.21	Probability of Detection in the Frequency Domain	227
7.22	Examples of Frequency Domain Detection	229
7.23	Comments on Frequency Domain Detection	231
	References	231
	Appendix 7.A	232
	Appendix 7.B	233
	Appendix 7.C	234
	Appendix 7.D	235
	Appendix 7.E	235
	Appendix 7.F	236
	Appendix 7.G	236
	Appendix 7.H	236
	Appendix 7.I	237
	Appendix 7.J	237

Appendix 7.K	238
Appendix 7.L	238
Appendix 7.M	239
8 Phase Measurements and Zero Crossings	241
8.1 Introduction	241
8.2 Digital Phase Measurement	242
8.3 Angle Resolution and Quantization Levels	243
8.4 Comparison of Phase Measurement and Fast Fourier Transform (FFT) Results	244
8.5 Application of the Phase Measurement Scheme	245
8.6 Analysis of Two Simultaneous Signals	246
8.7 Frequency Measurement on Two Signals	249
8.8 Single-Frequency Measurement from Zero Crossing	250
8.9 Ill Condition in Zero Crossing for a Single Signal and a Remedy	253
8.10 Simplified Zero Crossing Calculation for a Single Signal	253
8.11 Experimental Results from Single-Frequency Zero Crossing Methods	256
8.12 Application to Coherent Doppler Radar Frequency Measurement	257
8.13 Zero Crossing Used for General Frequency Determination	258
8.14 Basic Definition of Zero Crossing Spectrum Analysis	259
8.15 Generating Real Zero Crossings	260
8.16 Calculating Coefficients for Zero Crossing Spectrum Analysis	262
8.17 Possible Configuration of a Zero Crossing Spectrum Analyzer	264
References	265
9 Monobit Receiver	267
9.1 Introduction	267
9.2 Original Concept of the Monobit Receiver	268
9.3 Monobit Receiver Idea	268
9.4 Design Criteria	270
9.5 Receiver Components	270
9.6 RF Chain, ADC, and Demultiplexer	271
9.7 Basic FFT Chip Design	274

9.8	Frequency Encoder Design	275
9.9	Selection of Thresholds	277
9.10	Preliminary Performance of a Monobit Receiver	279
9.11	Possible Improvements	281
9.12	Chip Layout	283
	References	284
10	Frequency Channelization and Afterwards Processing	285
10.1	Introduction	285
10.2	Filter Banks and FFT	286
10.3	Overlapping Input Data in the FFT Operation	287
10.4	Output Data Rate from FFT Operation	288
10.5	Decimation and Interpolation	290
10.6	Decimation and Interpolation Effects on the DFT	293
10.7	Filter Bank Design Methodology	294
10.8	Decimation in the Frequency Domain	296
10.9	Output Filter Shape from a Decimated FFT Widened by a Weighting Function	298
10.10	Channelization Through a Polyphase Filter and Its Operation	301
10.11	Processing After Channelization	304
10.12	Basic Considerations of the Channelized Approach	305
10.13	Filter Shape Selection	306
10.14	Analog Filters Followed by Phase Comparators	310
10.15	Monobit Receivers Followed by Phase Comparators	311
10.16	Digital Filters Followed by Phase Comparators	313
10.17	Analog Filters Followed by Monobit Receivers	314
10.18	Considerations of Digital Filters Followed by Monobit Receivers	317
10.19	Increase the Output Sampling Rate by Two	319
10.20	Digital Filters Followed by Monobit Receivers	321
10.21	Digital Filter Bank Followed by Monobit Receivers and Phase Comparators	322
10.22	Digital Filter Bank Followed by Another FFT	322
	References	323

11	High-Resolution Spectrum Estimation	325
11.1	Introduction	325
11.2	AR Method	326
11.3	Yule–Walker Equation	327
11.4	Levinson–Durbin Recursive Algorithm	330
11.5	Input Data Manipulations	332
11.6	Backward Prediction and the Modified Covariance Method	335
11.7	Burg Method	338
11.8	Order Selection	341
11.9	Prony’s Method	342
11.10	Prony’s Method Using the Least Squares Approach	345
11.11	Eigenvectors and Eigenvalues	346
11.12	MUSIC Method	348
11.13	ESPRIT Method	350
11.14	Minimum Norm Method	353
11.15	Minimum Norm Method with DFT	355
11.16	Adaptive Spectrum Estimation	357
	References	361
	Appendix 11.A	364
	Appendix 11.B	365
	Appendix 11.C	366
	Appendix 11.D	366
	Appendix 11.E	367
	Appendix 11.F	368
	Appendix 11.G	370
	Appendix 11.H	371
	Appendix 11.I	373
12	Detection of Biphase Shift Keying (BPSK) Signals	375
12.1	Introduction	375
12.2	Basic Barker Code Properties	376
12.3	Generation of BPSK Signals with Synchronized Radio Frequency (RF) and Chip Time	377

12.4	Comparison of 11- and 13-Bit Barker Codes in the Frequency Domain	378
12.5	Threshold and Probability of Detection	381
12.6	Using Long FFT to Determine the Existence of a BPSK Signal	382
12.7	Differentiate a BPSK Signal from a CW Signal Through Eigenvalue Calculation	385
12.8	Using Frame Outputs to Differentiate CW and BPSK Signals: FFT and Eigenvalue Methods	388
12.9	Signal Generation for Detection of Phase Transitions	391
12.10	Approaches to Detect Phase Transitions	394
12.11	The FFT Method on One Frame of Data	395
12.12	Eigenvalue Method to Determine Phase Transition Using One Frame of Data and Comparison with the FFT Method	396
12.13	Eigenvalue Method to Determine Phase Transition Using Two Frames of Data	398
12.14	A Phase Comparison Method to Determine Phase Transition	401
12.15	Using Both the One-Frame Eigenvalue and the Phase Comparison Methods	405
12.16	Input Data with Two Phase Transitions in Consecutive Frames	405
12.17	Conclusion	407
	References	408
	Appendix 12.A	408
13	Detection of Frequency Modulated Signals	411
13.1	Introduction	411
13.2	Potential Problems with FM Signal Detection	412
13.3	Amplitude and Phase Comparison Approaches	413
13.4	Frequency Measured from the Three Methods	416
13.5	Phase Comparison Method Aided by the Amplitude Comparison Method	420
13.6	Chirp Signal Detection with Amplitude and Phase Comparison	423
13.7	Find the Chirp Rate of the Measured Signal	426
13.8	Four Approaches to Detect a Long Weak Signal	427
13.9	Threshold Setting for Both the Summation and Maximum Summation Methods	429

13.10	Threshold for the L -out-of- N Method	431
13.11	Threshold for the Eigenvalues and Summary of Thresholds	436
13.12	Sensitivity Study of All Detection Methods	436
13.13	Summary of Sensitivity	439
13.14	Brute Force Approach	441
13.15	Using the Eigenvalue Method to Detect the Existence of Chirp Signal	443
13.16	Conclusion	444
	References	446
14	Concept of Analog-to-Information	447
14.1	Introduction	447
14.2	Data Collection	448
14.3	Frequency Calculation	449
14.4	Probability of False Alarm and Probability of Detection	451
14.5	Frequency Measurement Accuracy	453
14.6	Amplitude Measurement	455
14.7	Study of Determining Two-Signal Frequencies	458
14.8	Two-Frame Approach for Detecting Two Signals	460
14.9	Two Signals with Different Amplitudes without Noise	463
14.10	Issues of Generating Thresholds for the Two-Frame Approach	464
14.11	Sensitivity and False Detection of a Second Detection of the Two-Frames Approach	466
14.12	Second Threshold Obtained from the First Input Signal	469
14.13	Sensitivity of Second Signal Detection and Instantaneous Dynamic Range	470
14.14	Conclusion	472
	References	473
15	Angle of Arrival Measurements	475
15.1	Introduction	475
15.2	Queuing Concept	476
15.3	Digital Data from a Linear Antenna Array	477

15.4	Outputs from a Circular Antenna Array	479
15.5	Two-Element Phase Array Antenna	482
15.6	AOA Measurement Through Zero Crossing	484
15.7	Phase Measurement in AOA Systems with Multiple Antennas	485
15.8	Fourier Transform Over the Space Domain	486
15.9	Two-Dimensional Fourier Transform	489
15.10	Frequency Sorting Followed by AOA Measurements	491
15.11	Minimum Antenna Spacing	492
15.12	Chinese Remainder Theorem	494
15.13	Application of the Chinese Remainder Theorem to AOA Measurements	495
15.14	Practical Considerations in the Chinese Remainder Theorem	497
15.15	Hardware Considerations for Digital AOA Measurements	498
	References	500
16	Time Reversal Study	503
16.1	Objective	503
16.2	Application to AOA Measurement	503
16.3	Measuring the Impulse Response	505
16.4	Quality of the Impulse Response Generated by a Chirp Signal	506
16.5	Impact on Input Data Length	508
16.6	Impact on Impulse Response from an Input Chirp Signal	508
16.7	Recovering the Input Signal from the Output Signal	511
16.8	Results from Recovered Signals	512
16.9	Fractional Time Delay	514
16.10	Phase Adjustment	516
16.11	A Different Approach to Introduce the Phase Angle	517
16.12	Switching the Delay from the Input to the Local Signal	518
16.13	Generate Input Data from One Set of Input Measurements	520
16.14	Conclusions	521
	References	521

17	Multiple Fast Fourier Transform Electronic Warfare Receiver	523
17.1	Introduction	523
17.2	EW Receiver Diagram	523
17.3	Encoder Description	524
17.4	PDW Combiner Description	525
17.5	Simulation Results and Discussions	531
17.6	Conclusions	536
	References	537
18	Receiver Tests	539
18.1	Introduction	539
18.2	Types of Receiver Tests	540
18.3	Preliminary Considerations in Laboratory Receiver Tests	541
18.4	Receiver Tests Through Software Simulation	543
18.5	Laboratory Test Setup	544
18.6	Anechoic Chamber Test Setup	546
18.7	Preliminary Tests	547
18.8	Single-Signal Frequency Test	547
18.9	False Alarm Test	549
18.10	Sensitivity and Single-Signal Dynamic Range	550
18.11	PA and PW Measurements	551
18.12	AOA Accuracy Test	553
18.13	TOA Test	553
18.14	Shadow Time, Throughput Rate, and Latency Tests	554
18.15	Two-Signal Frequency Resolution Test	555
18.16	Two-Signal Spurious-Free Dynamic Range Test	555
18.17	Instantaneous Dynamic Range Test	556
18.18	Anechoic Chamber Test	556
18.19	AOA Resolution Test	557

	Contents	xvii
18.20 Simulator Test		558
18.21 Field Test		559
References		560
Appendix 18.A		560
Appendix 18.B		561
Index		563

Preface

The purpose of this book is to introduce digital signal processing approaches that have been used for or are potentially applicable to wideband receiver designs. The emphasis is on techniques rather than theoretical discussions. Since the final goal in receiver design is to process the input data in near real time, the calculation speed of these processing methods is of primary concern.

This book is written for researchers in electronic warfare and communication areas. It assumes a senior or first-year graduate school level of knowledge in electrical or computer engineering. In order to help the reader to understand the subjects, numerous examples are included. Many computer programs are also included to further illustrate some of the ideas. Compared with other receiver books on the market, the uniqueness of this edition and previous editions is the detailed instruction concerning how to simulate and implement receiver techniques. The book's down-to-earth approach has made it very popular among receiver researchers because it is almost like a manual providing step-by-step instructions. This edition will follow the same approach in covering new topics that were not covered by the 2004 edition.

Digital signal processing has been widely applied in many technical areas. In the past, digital signal processing has been used after implementation of crystal detectors in electronic warfare (EW) receivers. The technology advances in analog-to-digital converters (ADCs) have opened a new era in receiver design. An ADC can replace a crystal detector and retain valuable information that would otherwise be lost after detection.

The requirements of EW and communication used to be very different from each other. EW receivers required a very wide bandwidth, while communication receivers required a relatively narrow bandwidth. However, in recent years the requirements in these two areas have become more closely aligned. The most significant requirements are wide input bandwidth coverage and high dynamic range in both types of systems. As a result, many techniques that were previously considered unique for EW receivers are now applicable to communication systems as well. However, there remains a fundamental difference between a communication receiver and an EW receiver: the communication receiver operates upon signals from cooperative transmitters, while the EW receiver is designed to intercept signals emitted from noncooperative transmitters. Most application examples considered in this book are for EW applications, although readers might find some techniques can also be applied in the design of a communication receiver.

This book is divided into 18 chapters. Chapter 1 is an introduction. Chapters 2, 17, and 18 are the only chapters devoted to the EW receiver. Chapter 2 provides a very brief review of EW, Chapter 17 presents an EW receiver algorithm to estimate signal characteristics, and Chapter 18 discusses the evaluation of EW receivers. Fourier transforms, discrete Fourier transforms, and some related subjects that are especially interesting for wide bandwidth receiver designs are included in Chapters 3 and 4. Chapters 5 and 6 cover receiver hardware. Chapter 5 discusses the impact of ADC on receiver performance and presents designs of a receiver front end consisting of an amplifier chain and ADC. Chapter 6 discusses in-phase and quadrature-phase converter design. The probability of false alarm and probability of detection are discussed in Chapter 7. Phase measurement and zero crossing methods to measure frequency are discussed in Chapter 8. Chapters 9 through 11 discuss methods closely related to receiver design. Chapter 9 presents a simple design of an EW receiver. Chapter 10 discusses frequency channelization and processing methods after channelization. High-frequency resolution is discussed in Chapter 11. The detections of two exotic radar signals, biphas shift keying (BPSK) and chirping radar signals, are discussed in Chapters 12 and 13, respectively. Analog-to-information techniques allowing a receiver to cover a broad bandwidth using a relatively low sampling rate are introduced in Chapter 14. Angle measurement is discussed in Chapter 15, and calibration of an angle measurement system is presented in Chapter 16.

The MATLAB[®] program and data files described in this book can be obtained by emailing the publisher at books@theiet.org.

Acknowledgements

Authors

It must be noted that there are several new chapters included in this edition. Among the new chapters, the materials of Chapters 14 and 16 are closely related to the work of Dr. Lihyeh Liou and David Lin, both engineers from the Air Force Research Laboratory (AFRL). We greatly benefited from their assistance. Dr. Liou kindly reviewed these two chapters and provided valuable comments.

Numerous organizations and individuals provided assistance in the process of writing this book, including both the current and previous editions, and we would like to acknowledge their help. The authors have very much appreciated valuable discussion with Dr. William McCormick and Dr. Arnab Shaw from Wright State University and Dr. Richard Sanderson from the Air Force Research Laboratory. Special thanks to two engineers, Timothy Fields from System Research Laboratory and David Sharpin from the Air Force Research Laboratory, for their technical discussions. We would also like to thank many of our colleagues (in alphabetical order): Debbie Ables, Peter Buxa, Joseph Caschera, Dr. Vasu Chakravarthy, Luis Concha, Ed Culpepper, Thomas Dalrymple, Robert Davis, Lt. George Dossot, Keith Graves, Dr. Paul Hadorn, Dr. Stephen Hary, James Hedge, David Jones, Keith Jones, Matthew Longbrake, Lt. Jamie Marciniec, Emil Martinsek, John McCall, Darnetta Meeks, Lt. Christine Montgomery, William Moore, Thomas Pemberton, Nicholas Pequignot, Capt. Daniel Richards, Scott Rodrique, Dr. Steve Schneider, Rudy Shaw, Lt. Jason Shtrax, Joseph Tenbarge, Anthony White, Capt. Daniel Zahirmiak from the Air Force Research Laboratory, and Jerry Bash, James McCartney, Mark Thompson, and Jeff Wagenbach from the System Research Laboratory. Miami University provided tremendous help in the process of writing this edition and we would like to thank Dr. Qihou Zhou, Dr. Donald Ucci, Dr. Jade Morton, and Michele Lea for their generous help. This edition is published by SciTech Publishing, an imprint of IET. We would like to acknowledge the assistance we received from Dudley Kay, Jennifer Grace, Joanna Hughes, Paul Deards, and Rachel Williams from the IET editorial office and Vijay Ramalingam from MPS Limited.

Last, but not least, we wish to thank our spouses, both of whom are named Susan, for their encouragement and understanding of our spending considerable time on this book.

Publisher

SciTech-IET is indebted to peer reviewers who have unselfishly given their time and expertise to help refine draft chapters in order to improve clarity and accuracy for future readers. We wish to thank the following reviewers:

Bevan D. Bates, Adjunct Professor, School of Electrical and Electronic Engineering, University of Adelaide, Adelaide, SA, Australia

Dr. Jim P. Y. Lee, Former Expert-Level Defense Scientist, Defence R&D Canada, Ottawa, ON, Canada

Professor Douglas Moody, Mercer University, Macon, GA, USA

Charles R. Ward (Retired), Senior Technical Consultant, ITT Industries/Excelis, McLean, VA, USA

Abbreviations

ADC	analog-to-digital converter
AIC	Akaike information criterion
AM	amplitude modulation
AOA	angle of arrival
AR	autoregression
ARMA	autoregressive moving average
BPSK	biphase shift keying
CAT	criterion autoregression transfer
CW	continuous wave
DC	direct current
DFT	discrete Fourier transform
DMA	direct memory addressing
ECCM	electronic counter-countermeasures
ECM	electronic countermeasure
ELINT	electronic intelligence
EM	electromagnetic
EOB	electronic order battle
ESM	electronic support measures
ESPRIT	estimation of signal parameters via rotational invariance
EW	electronic warfare
FFT	fast Fourier transform
FIR	finite impulse response
FM	frequency modulation
FPE	final prediction error
IDFT	inverse discrete Fourier transform
IF	intermediate frequency

xxiv **Abbreviations**

IFM	instantaneous frequency measurement
IIR	infinite impulse response
LMS	least mean square
LSB	least significant bit
MA	moving average
MDL	minimum description length
MEM	maximum entropy method
MSB	most significant bit
MUSIC	multiple signal classification
PC	personal computer
PDW	pulse descriptor word
PRF	pulse repetition frequency
PRI	pulse repetition interval
PSK	phase shift keying
RF	radio frequency
SNDR	signal-to-noise plus spur distortion ratio
STFT	short time Fourier transform
TOA	time of arrival

Introduction

1.1 Wideband Systems

This book discusses digital signal processing schemes that are potentially applicable to electronic warfare (EW) receivers. These receivers must have very wide instantaneous input bandwidth (about 1 GHz) to fulfill their operational requirements. This means any signal within the input bandwidth will be received all the time without tuning the receiver. In contrast, a communication receiver has a relatively narrow bandwidth. For example, television channels are allotted 6 MHz (although digital TV has become a reality in the United States, the channel bandwidth of a digital TV channel is the same as an analog TV channel), frequency modulated (FM) radio channels are allotted about 200 kHz, and amplitude modulation (AM) stations are allotted only 10 kHz [1]. If one turns on 10 television sets simultaneously and each one is receiving a different channel, the instantaneous bandwidth of such an arrangement is considered to be 60 MHz (ten 6-MHz channels).

However, communication bandwidth is increasing, because the wider the bandwidth, the more information per unit time can be transmitted from one point to another. Some current wireless communication systems, such as the 4G-LTE system, might use up to 100 MHz of bandwidth to support high data transmission rate [2]. If this trend continues, the working bandwidth of an EW receiver and a communication receiver might be very comparable in the future. Further discussion with communication engineers reveals that many of the hardware considerations and digital signal processing approaches primarily designed for EW receivers are equally applicable to communication receivers. That is the reason for selecting the name of this book. However, there are fundamental differences between an EW receiver and a communication receiver. A communication receiver is designed to recover signals from a cooperative transmitter. In contrast, an EW receiver is used to intercept signals from noncooperative radars and determine signal characteristics, thus using this information to determine the types of radars. The primary emphasis of this book is still on EW receivers rather than communication receivers, and most of the examples in this book are from EW receivers, although the reader might find that some of techniques presented can also be used to design a communication receiver.

1.2 Digital Approach [3–6]

Many communication and control problems have been solved through digital approaches. Today there is little doubt that many engineering problems (i.e., communication and control)

can be best solved through digital signal processing. Digital circuits have long been used in EW receiver applications, such as digital controlling of receiver operation modes. Traditional EW receivers use crystal video detectors to convert radio frequency (RF) signals into video signals. Once the RF input signal is converted into video signals through crystal detectors, the video signals are further processed digitally. However, the crystal video detector destroys the carrier frequency and phase information in the signal. If the crystal detector is replaced by an analog-to-digital converter (ADC), all the information is maintained. In past decades, significant progress has been made in the development of digital wideband receiving systems.

The ADC is used to convert analog signals into digital data for further processing. In order to convert signals in a wideband receiver, the ADC must operate at a very high sampling rate. To digitize signals with fewer quantization errors, the ADC must also use a large number of bits to represent each sample. It is difficult to achieve both goals in an ADC at the same time, but advancements in ADC technology are moving at an astonishing pace. It is even difficult to make a meaningful assessment of ADC technology because of its rapid development. Figure 1.1 shows a survey on ADCs as of April 1993. This figure is based on information obtained from Walden [3]. The “×DARPA goal” indicates that the Defense Advanced Research Projects Agency (DARPA) has a program to build a 100-MHz 12-bit ADC, while the “×WL goal” indicates that the Wright Laboratory (WL), whose name is later changed to Air Force Research Laboratory (AFRL), has a project to build a 20-GHz 4-bit ADC. Currently (2014), some commercially available high-speed ADCs can operate above 2 GHz with a more than 10-bit sample size [4].

High-speed ADC outputs must be processed by high-speed digital circuits. Otherwise the digitized data is lost and the system cannot operate in a real-time mode. Current operating speeds of digital hardware do not yet match the speed of state-of-art ADCs, and the speed of digital processors may never catch up with the speed of ADCs. Nevertheless, by applying techniques such as parallel processing, the capability of a digital system such as a field-programmable gate array (FPGA) has been demonstrated for high-speed digital signal processing [5].

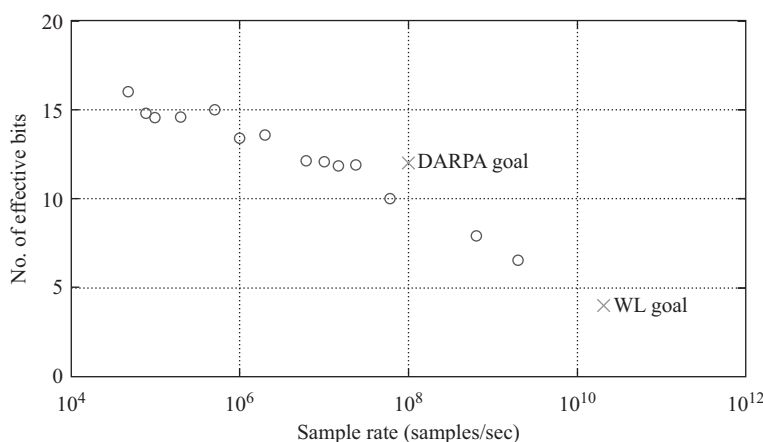


Figure 1.1 Performances of ADCs.

Because of improvements in ADCs and digital circuits, it is anticipated that ADCs will replace crystal detectors in all RF receivers, thereby preserving RF and phase information. Furthermore, the ADC might move toward the front end of the receiver; that is, moving from the intermediate frequency (IF) toward the RF end. In the future, it might be possible to design receivers with only RF amplifiers and bandpass filters between the antenna and the ADC and the mixer is no longer needed [6].

1.3 Obstacles in EW Receiver Development

Technically, it appears that there are at least two research and development obstacles in the field of EW receivers. First, scientists and engineers not directly working in the EW field have very little knowledge of it and are not aware of its requirements. The communication field is very much different from EW. An understanding of communication systems is often required in an undergraduate electrical engineering curriculum, therefore most electrical engineers have been exposed to basic concepts of communication systems. (It is interesting to notice that the word “communication” is used both in liberal arts and engineering schools with the same meaning but different emphases. As a result, this word has different interpretations. In liberal arts it means improving one’s interpersonal communication skills, while in engineering it means how to modulate and demodulate a signal for transmitting and receiving.)

In contrast, EW systems are only taught at a very few military academies or in short courses. Therefore it is difficult for an EW engineer to communicate with scientists and engineers outside the EW field to solicit new ideas. To remedy this problem, Chapter 2 is devoted to a discussion of EW, with an emphasis on intercept systems.

The other problem is, in EW receivers there is no universally accepted evaluation standard. All kinds of performance values are used to describe EW receivers. Unfortunately, many of these receivers cannot even be considered as EW receivers because they do not provide the desired data format required by EW operations. This problem is addressed in Chapter 18. Chapters 2 and 18 are the two chapters in this book that deal with both analog and digital EW receivers.

1.4 Organization of the Book

Many different subjects are discussed in this book and we attempted to arrange these subjects in a coherent way. The subjects are divided into the following groups.

Chapters 2, 17, and 18 cover EW receivers. Chapter 2 provides a general discussion of EW with an emphasis on EW receivers, as mentioned in the previous section. In an EW receiver, an encoder is used to generate a pulse descriptor word (PDW) consisting of the estimates of signal characteristics such as frequency, power, pulse width, etc., often considered as the most difficult function of an EW receiver. In Chapter 17, an encoder algorithm based on multiple fast Fourier transform (FFT) frame lengths is presented as a design example. Chapter 18 presents some measurement methods for EW receivers to obtain the values discussed in Chapter 2.

Fourier transform and related subjects are covered in Chapters 3 and 4. Because Fourier transform is needed in discussing other subjects, this subject is discussed first in Chapter 3. Some commonly used Fourier transform examples are included in Chapter 3 as a quick reference. Readers familiar with Fourier transform may skip this chapter. Chapter 4 presents several subjects that are closely related to Fourier transform.

Hardware used in wideband receivers is covered in Chapters 5 and 6. Chapter 5 discusses ADCs, amplifiers, and their interface. While discussing ADCs, the major focus is the effect of their performance on receiver performance rather than the ADC implementation technology. Because the ADC is a nonlinear device, mathematical analysis is difficult and limited. A large number of computer simulations are used to provide a rough estimate of its performance. In Chapter 5, the design of a digital receiver's RF front end consisting of an amplifier chain and ADC is also considered. It involves a trade-off in terms of sensitivity and dynamic range. Chapter 6 discusses frequency conversion. Both analog and digital approaches are included. The imbalance occurring in down-converting a real-valued signal to a complex-valued signal deteriorates the receiver performance. An imbalance compensation technique is presented at the end of Chapter 6.

Sensitivity, frequency measurement, and receiver designs are covered in Chapters 7 through 10. Chapter 7 presents several signal detection schemes. The approaches include detection in both the time and frequency domains. Chapter 8 presents two simple schemes to improve frequency measurement accuracy. These methods have the potential to make the frequency resolution dependent on pulse width such that higher frequency resolution can be obtained on longer pulses. This is a very desirable feature in EW receivers. Chapter 9 presents a simple monobit receiver design with inferior performance but the potential for fabrication on a single chip. Chapter 10 presents an effective approach to channelize the frequency. The concept of decimation and multirate will be introduced. Some potential approaches for improving frequency measurement after channelization are discussed in Chapter 10 as well. These methods might be adopted in an encoder design. Several advanced EW receiver techniques are covered in Chapters 11 through 14. High-frequency resolution measurement is covered in Chapter 11. Chapter 11 also presents some high-resolution parametric spectrum estimation methods developed in last several decades. These methods usually provide finer frequency resolution than the FFT approach, but they are computationally intensive. An EW receiver is designed to detect radar signals; most of the application examples covered in previous editions of this book considered continuous-wave (CW) signals. In this edition, two exotic radar signals, biphase shift keying (BPSK) and frequency modulated (FM) signals, are considered in Chapters 12 and 13, respectively, as important updates in EW receiver technologies. The bandwidth of a digital receiver is determined by the sampling rate of its ADC. To increase receiver bandwidth, the most straightforward way is to increase the sampling rate of the ADC. However, this might not always be a feasible solution. Chapter 14 presents a technology known as analog-to-information (A-to-I), which allows a digital receiver to cover a broad bandwidth with a low sampling rate.

Most receiver examples considered in this book use only one antenna, therefore they cannot measure a signal's angle of arrival (AOA), but the AOA is an important signal characteristic. Chapter 15 discusses AOA measurements. Some practical problems and suggested solutions are also included. An AOA system calibration method based on the time-reversal concept is discussed in Chapter 16.

1.5 Specific Remarks

Many computer programs are included in this book to help readers understand the design of a receiver. All these programs are written in MATLAB® (MathWorks, Natick, MA, USA). Most of the high-resolution schemes in Chapter 11 have a computer program associated with them.

Many of the figures presented in this book were produced using MATLAB. The time scale is often labeled as “time sample” since it represents samples of the time domain. The corresponding frequency plot is labeled as “frequency bin” since it is the spectrum calculated from the FFT operation on the time domain samples.

If one is not familiar with this technical area or does not start from the beginning of the book, it may be difficult to recognize some of the acronyms. Acronyms and abbreviations are useful for expressing technical terms. For convenience, all acronyms are spelled out the first time they appear in a chapter, even if it is a very common one. In addition, a list of all the acronyms is provided at the back of the book.

References

- [1] Westman HP (ed.). *Reference Data for Radio Engineers*, 5th ed. Indianapolis: Howard W. Sams & Co.; 1968.
- [2] Bleicher A. ‘4G gets real’. *IEEE Spectrum Magazine* 2014;**51**(1):38–62.
- [3] Walden R. Hughes Research Laboratories, Malibu, CA. Private communication.
- [4] Knowles J. ‘Technology survey: a sampling of analog-to-digital converters and modules’. *Journal of Electronic Defense* 2013;**36**(7):35–41.
- [5] Longbrake M. *Derivation of Parallel Polyphase Filter for FPGA Implementation*. Technical report [unpublished]. Air Force Research Laboratory; 2009.
- [6] Brown A, Wolt B. ‘Digital L-band receiver architecture with direct RF sampling’. *Proceedings of the Position Location and Navigation Symposium*, Las Vegas, NV, USA, New York: IEEE; April 1994:209–216.

Requirements and Characteristics of Electronic Warfare Receivers

2.1 Introduction

One of the most difficult problems in inviting researchers to work on electronic warfare (EW) receivers is that the subject is not well understood, especially by academic scholars (perhaps due to a lack of exposure to the problem). The main purpose of this chapter is to introduce the concept of an EW receiver. In order to provide a broader view, the subject of EW will be briefly discussed. The signal environment and the requirements of the EW receiver will also be discussed. Since EW is basically a responsive action to a hostile electronic environment, the requirements will change with time. If the enemy creates a new threat, EW engineers and systems must respond in a timely manner.

This chapter will first discuss the information contained in a radar pulse, then some of the difficulties encountered in receiver research will be presented. A very simple discussion on analog and digital receivers will then be presented and, finally, the characteristics of EW receivers will be discussed. If several definitions of one term are available, only the ones that have a direct impact on EW receivers will be discussed. All the characteristics presented here are measurable quantities, and the measurements are presented in Chapter 18. Finally, the research trend in EW receivers will be discussed. Readers with an EW background can skip this chapter.

2.2 Introduction to Electronic Warfare [1–6]

An EW system is used to protect military resources from enemy threats. The field of EW is recognized as having three components:

- Electronic support measures (ESM), which collects information on the electronic environment.

- Electronic countermeasures (ECM), which jams or disturbs enemy systems.

- Electronic counter-countermeasures (ECCM), which protects equipment against ECM.

Because it does not radiate electromagnetic (EM) energy, the first is often referred to as a passive EW system. The second is referred to as an active EW system since it radiates EM energy. Because they do not emit EM energy, techniques such as stealth targets (that avoid

being detected by enemy radars) and deployment of decoys or chaff (thin metallic wires) to confuse enemy radars are also considered as passive EW. ECCM is usually included in radar designs, hence it will not be discussed here.

EW intercept systems can be divided into the following five categories:

1. Acoustic detection systems are used to detect enemy sonar and noise generated by ship movement. These systems detect acoustic signals and usually operate at frequencies below 30 kHz.
2. Communication intercept receivers are used to detect enemy communication signals. These systems usually operate below 2 GHz, although a higher operating frequency is required to intercept satellite communication. These receivers are designed to receive communication signals.
3. Radar intercept receivers are used to detect enemy radar signals. These systems usually operate in the range of 2 to 18 GHz. However, some researchers intend to cover the entire 2 to 100 GHz range. These receivers are designed to receive pulsed signals.
4. Infrared intercept receivers are used to detect the plume of an attacking missile. These systems operate at near through far infrared wavelengths (from 3 to 15 μm).
5. Laser intercept receivers are used to detect laser signals, which are used to guide weapons systems (i.e., attack missiles).

The intercept receivers often operate with EW signal processors. The processors are used to process the information intercepted by the receivers to sort and identify enemy threats. After the threats are identified, the information is passed to an ECM system. The ECM system must determine the most effective way to disrupt the enemy, which may include throwing out chaff. The actions of an ECM system against radars include noise and deceptive jamming. Noise jamming is intended to mask the radar's return signals from targets by noise so that the radar cannot detect a signal and its screen is covered with noise. Deceptive jamming creates false targets on the radar screen such that the radar will lose the true targets.

An EW system with all its different functions is shown in Figure 2.1. Except for the summary material in this chapter, this book is devoted to only a small part of the EW system, namely, the EW radar intercept (or just EW) receiver section, which is used to intercept radar signals and convert them to digital pulse descriptor words (PDWs).

In the past, the EW receiver has been used to demodulate radar signals, convert them into video pulses, and generate a tone in an earphone. An EW officer would listen to the tone and determine whether it was a threat radar. Under this type of operation, the EW officer works as the processor. However, this operation cannot satisfy modern requirements. In a modern EW system, in order to cope with the complexity of the electronic environment, digital EW processors are used to identify threats. As a result, EW receivers must generate digital outputs, which are used as the inputs of EW processors.

2.3 Difference between Intercept and Communication Receivers [7–13]

Most people have some exposure to communication receivers (i.e., a television set or a car radio). In designing these kinds of receivers, the frequency, types of modulation,

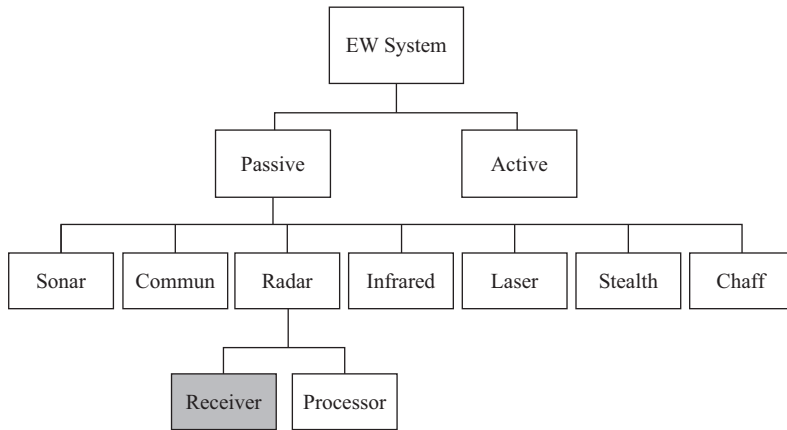


Figure 2.1 Different functions of an EW system.

and bandwidth of the incoming signal are known. Thus the input signal can be considered as a cooperative type and the receiver can be designed very efficiently. A radar receiver can be considered a communication receiver because the input signal is known. In an intercept (or EW) receiver, not only is the information of the input signal unknown, but the transmitting signal may be specifically designed to avoid detection by an intercept receiver.

The other major difference between an EW receiver and other types of receivers is that the outputs of an EW receiver are digital words describing the characteristics of every individual radar pulse intercepted. The receiver generates a PDW that includes frequency, incident direction, pulse width, pulse amplitude, and time of arrival for each pulse. This unique characteristic of EW receivers sometimes causes misunderstandings for designers of other types of receivers. For example, the main purpose of a communication receiver is to recover the information emitted by the transmitter. If the transmitted signal is analog (i.e., voice, pictures), the receiver will produce voice or pictures as the final output. In most cases, a digital output is not required. A recent trend in communication systems is to convert the analog signal into digital for transmitting and processing and then to convert it back to analog form. In this sense, the two types of receivers are becoming alike, but still the communication receiver does not perform parameter encoding.

In an EW receiver, the incoming information (the radar pulses) is analog, but the outputs are always presented in digital form as PDWs. In most older EW receivers, the radio frequency (RF) signals are first converted into video signals by diode envelope video detectors. Specially designed encoders are used to convert these video signals into PDWs, as shown in Figure 2.2. Although some EW receivers may have very impressive video outputs, matching computer-simulated results, these outputs do not guarantee the generation of satisfactory PDWs. Past experience indicates that most of the receiver design problems occur in converting video signals into PDWs. Thus a good EW receiver must produce satisfactory PDWs as output.

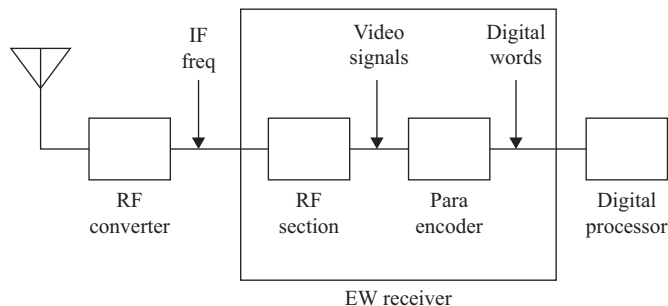


Figure 2.2 A traditional EW receiver.

2.4 Signal Environment for EW Receivers [1–3, 10, 12–19]

Since an EW receiver is used to intercept radar signals, the signal environment in which an EW receiver operates will be discussed here. Weapon radars are the primary interest of an EW receiver. In contrast to a communication signal, a weapon radar generates very simple waveforms. Most radars generate pulsed RF signals. Some radars generate frequency modulated (FM) pulsed signals, which are often referred to as chirp signals. RF ranges roughly from 2 to 100 GHz, but the most popular frequency range is from 2 to 18 GHz. The duration of these pulses may be between tens of nanoseconds and hundreds of microseconds. Some radars produce continuous wave (CW) signals for low altitude surveillance or weapons guidance. The pulse repetition frequency (PRF), or its reciprocal the pulse repetition interval (PRI), is an important parameter of pulsed radar signals. The PRF ranges roughly from a few hundred hertz to about 1 MHz. Most radars have a stable PRF, which means the PRF is a constant. Some radars have a staggered PRI; that is, a group of pulses (i.e., a few to tens) repeat themselves at a certain PRF. Some radars even generate agile or random PRIs; that is, the PRI varies from pulse to pulse. Agile PRI usually means that the PRI varies in a certain pattern and random PRI means the PRI does not have any predetermined pattern.

There are radars called low probability of intercept (LPI) radars. One of their main design goals is to avoid detection by enemy intercept receivers. These types of radars can either control their radiation power or generate wideband (spread spectrum) or frequency agile signals. A radar with power control capability only radiates enough power for target detection. If a detected target is getting closer to the radar, the radar can reduce its transmitter power. Its main goal is to provide just enough power to keep the detected target in range. This operation reduces the probability of being detected by an intercept receiver.

Some radars generate wideband signals to improve ranging resolution. The radar receiver can use matched filtering or signal processing to produce processing gain. It is difficult to detect a wideband signal with an intercept receiver because if the exact waveform is not known, one cannot produce the matched filter gain. However, the spread spectrum generated by a radar is relatively simple in comparison with communication signals. Only three types of spread spectra are of concern to EW receiver designers: FM (chirp) pulsed signal, pulsed biphasic shift keying (BPSK), and polyphase coded signals. The pulse width of these types of signals can range from several milliseconds to a few hundred milliseconds. Against these spread spectrum signals, detecting the signal becomes the primary task. Once the signal is

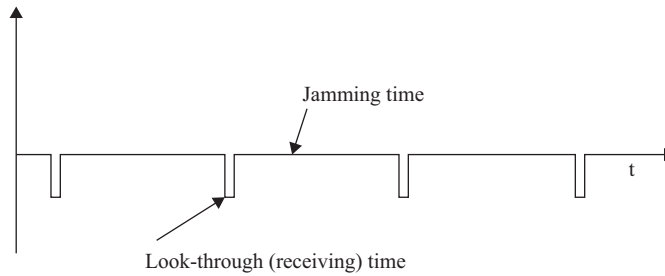


Figure 2.3 Look-through time.

detected, one can perform identification. In a frequency hopping radar, the RF of the pulse changes from pulse to pulse. This kind of radar is usually not of concern to an intercept receiver because the receiver can intercept these pulses without much difficulty. However, this type of radar may cause problems for the EW signal processor following the receiver because it is difficult to deinterleave them into a pulse train.

A threat radar can obtain the necessary information and take action against an airplane or a ship in a few seconds. If a missile guidance signal is detected by an EW receiver, the impact could be imminent. Thus an EW system must respond to input signals as soon as possible. If an EW system cannot respond within some critical time, it is equivalent to no EW system at all, as it cannot protect the aircraft or ship as desired.

During conventional EW operation, a jammer works almost continuously. When a jammer is working, it usually disturbs the operation of the EW information collection since the jammer is located close to the receiver. Its power might block the receiver from receiving. In actual operation, the jammer usually stops jamming temporarily so the receiver can collect information, such as to determine whether the signals being jammed are still present. This time duration is called the look-through time, and it has a 5% or less duty cycle, as shown in Figure 2.3. The actual data collecting time is a few to tens of milliseconds.

In an electronic order battle (EOB), many different radars are present, including friendly ones. Although the pulse density depends on the location of the intercept receiver and the scenario, it is usually assumed that a receiver will face a few million pulses per second. This signal environment determines the requirements of EW receivers.

2.5 Requirements of EW Receivers [1–3, 6, 13, 20–27]

From the discussion in the previous section, an EW receiver should fulfill the following requirements.

1. It requires near real-time response. In general, after the receiver intercepts a pulse, the measured information (i.e., the PDW) must be passed to an EW processor within a few microseconds.
2. The input signal range (say, 2 to 18 GHz) is often divided into many subbands. The frequencies of these subbands will be converted to outputs with a common intermediate frequency (IF). An EW receiver will be time-shared among all the outputs. In order to cover the input frequency band in a rapid manner, the IF bandwidth must be wide.

This means the instantaneous bandwidth of the EW receiver must be wide. Instantaneous bandwidth means all signals with enough energy in this bandwidth will be detected instantly. An optimum bandwidth of an EW receiver has not yet been obtained, because it depends on the input scenario, the capability of the digital EW processor following the receiver, and other factors. As a result, too many parameters need to be optimized to obtain the shortest response time. The present method for determining the bandwidth is to make it as wide as technology allows. It usually ranges from about 0.5 to 4 GHz, and a receiver with a bandwidth of less than 500 MHz is usually considered unacceptable for EW applications. Of course, if one can make a narrowband (less than 500 MHz) receiver compact and low cost, theoretically one can use many such receivers in parallel to cover a wide bandwidth.

3. The receiver is required to process simultaneous signals. If more than one pulse arrives at the receiver at the same instant in time, the receiver should obtain the information on all the pulses. The maximum number of simultaneous pulses a receiver is required to process is often considered to be four.
4. A proper trade-off of sensitivity and dynamic range in an EW receiver must be achieved. Of course, high sensitivity is always desirable since one can detect a radar at a greater distance, which provides more time to respond, or it can detect radars from their antenna sidelobes. Receivers with high dynamic range can receive simultaneous signals without generating spurious signals. In receiver design, these two parameters work against each other. Higher sensitivity almost always leads to lower dynamic range. Hence a compromise between these two quantities must be carefully evaluated.

2.6 Parameters Measured by an EW Receiver [28, 29]

An EW receiver must be able to obtain all the information from a pulse transmitted by a radar. Figure 2.4 illustrates a pulse transmitted by a radar. When the pulse reaches the

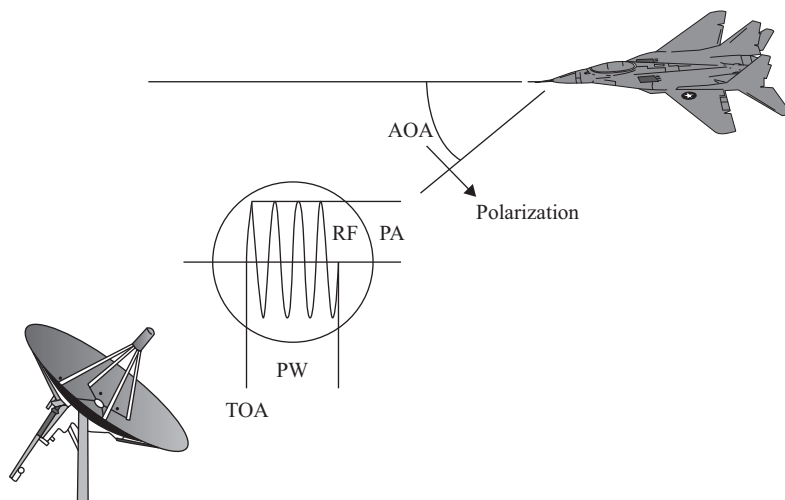


Figure 2.4 Parameters in a radar pulse.

intercept receiver at the aircraft, the following information can be measured: pulse amplitude (PA), pulse width (PW), time of arrival (TOA), carrier frequency (also referred to as the RF), and angle of arrival (AOA). In very limited cases, the electric polarization of the input signal is also measured. PA and PW measurements are self-explanatory. In EW applications, an input signal may be designated as CW when the pulse width is longer than a certain pre-determined value (i.e., tens to a few hundreds of microseconds). The TOA measurement assigns a specific time tag from an internal clock in the receiver to the leading edge of a received pulse. The TOA information is used to generate the PRF of a radar. The differences in EW receiver designs are largely based on the techniques used to measure the carrier frequencies of pulses. The AOA information is of primary importance, and it is also the most difficult to obtain. In the following sections the discussion will concentrate on frequency and AOA measurements.

2.7 Frequency Information [5, 30–40]

An intercept receiver measures only the center of the carrier frequency of a pulse. In general, the distribution of the spectrum is not needed. If the input is a chirp signal, the information of interest is usually the starting and ending frequencies and the PW. The chirp is often assumed to be linear. The chirp rate R_c can be calculated by dividing the difference in frequency from the leading and trailing edges by the PW. Mathematically this can be expressed as

$$R_c = \frac{f_t - f_l}{PW}, \quad (2.1)$$

where f_l and f_t are the frequencies at the leading and trailing edges, respectively.

In order to measure the frequency of a signal with variable frequency, theoretically the concept of instantaneous frequency should be considered [28, 29]. In an EW receiver, these frequencies are measured as the average frequency over a short period (i.e., 100 ns at the leading and trailing edges). If the input is phase coded, the carrier frequency and the chip rate (the phase shift clock) are of interest. In contrast with a communication receiver, the desired information from a pulse is quite different.

It is desirable to obtain fine frequency resolution on the input signal because the resulting high accuracy permits the jammer to concentrate its power on the victim radar. However, in the conventional approach, a receiver is designed to intercept the signal with the minimum anticipated PW. A typical minimum PW can be considered as 100 ns. In order to design a bandpass filter to process this signal, the required bandwidth is approximately 10 MHz (or 1/100 ns). This bandwidth limits the frequency resolution to about 10 MHz. Various designs have been attempted to obtain accuracies of frequency measurements to a small fraction of the frequency resolution (e.g., by comparing outputs of adjacent filters to interpolate within the filter and signal bandwidth). Some of these approaches do provide limited success, but at the cost of causing other problems such as increased spurious signal detections.

In a receiver, the system noise bandwidth is usually set by the narrowest filter bandwidth in the RF chain. The noise floor may be defined as the effective input noise level of a system

operating with an input temperature T_0 of 290K. For example, the noise floor $[N \text{ (dBm)}]$ of a 10-MHz system is

$$N(\text{dBm}) = 10 \cdot \log(kTB) = -104 \text{ dBm}, \quad (2.2)$$

where k is Boltzmann's constant ($= 1.38 \times 10^{-20} \text{ mJ/K}$), $B = 10^7 \text{ Hz}$, and

$$P(\text{dBm}) = 10 \cdot \log[P(\text{mW})]. \quad (2.3)$$

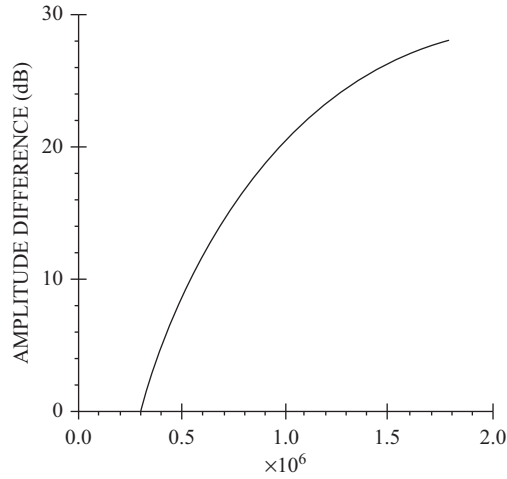
If the receiver has a noise figure of 15 dB and a threshold of 15 dB, the sensitivity is about -74 dBm ($-104 + 15 + 15$).

It is also desirable to build an intercept receiver that has adaptive frequency resolution (i.e., to make the frequency resolution PW dependent). For short pulses, the receiver can only generate coarse frequency resolution, whereas for long pulses the receiver can generate fine frequency resolution. This concept can be extended to the sensitivity design of a receiver (i.e., the receiver would have moderate sensitivity for short pulses and higher sensitivity for longer pulses). The adaptivity requirement might be difficult to attain in an analog receiver, which is hardware oriented, but it should be easier to accomplish with software in a digital design.

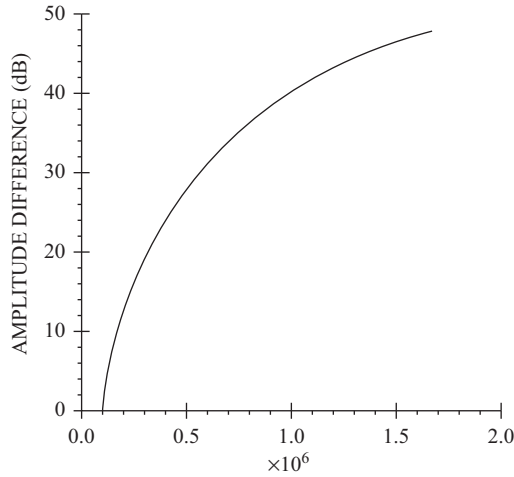
Another problem with frequency measurement is theoretical. The question is what frequency accuracy can a receiver measure on two signals given frequency separation, PW, and the signal-to-noise ratio (S/N). In most signal processing articles the Cramer–Rao bound is used as the maximum likelihood upper bound. The Cramer–Rao bound determines the minimum variance of the unbiased estimate. Two examples are shown here [36, 37]. In Figure 2.5(a), $PW = 0.1 \mu\text{s}$, $S/N = 20 \text{ dB}$, and the desired frequency measurement accuracy is 1 MHz. Under this condition, one should be able to measure two signals to 1-MHz accuracy when they fall on the right side of the curve. For example, if two frequencies are separated by 10 MHz and 18 dB in amplitude, one can measure both of them to 1-MHz accuracy. Figure 2.5(b) shows similar results for two 1- μs signals. Under this condition, if two signals are separated by 2 MHz in frequency and 50 dB in amplitude, one can measure both of them to 1-MHz accuracy.

In reality, there is no known EW receiver that can measure two simultaneous signals of 0.1- μs pulse width with 1-MHz resolution, no matter how far apart they are separated in frequency and how close they are in amplitude. The only receiver that can measure a 0.1- μs pulse with 1-MHz accuracy is the instantaneous frequency measurement (IFM) receiver, which cannot measure simultaneous signals. The IFM receiver will be briefly discussed in Section 2.11. Most of the EW receivers cannot detect two signals separated by more than 40 dB in amplitude, but according to Figure 2.5(b), 55 dB should be achievable. Both figures predict that when two signals are separated by a few megahertz with close amplitudes, their frequencies can be measured. However, most EW receivers are designed to begin measuring signals separated by at least 20 MHz.

From the above discussion, it is obvious that there is a large discrepancy between the theory and a practical EW receiver. Either the receivers are poorly designed or the Cramer–Rao bound is not suitable for EW applications. The latter might be true. Because the input signals of an EW receiver are unknown, one cannot design an optimum receiver under this condition. Thus, when an EW receiver design is initialized, the desired performance is obtained on past experience rather than from a theoretical analysis. Unfortunately, this kind



(a) FREQUENCY SEPARATION (Hz)



(b) FREQUENCY SEPARATION (Hz)

Figure 2.5 Cramer–Rao bound (a) for a 0.1-μs pulse and (b) for a 1-μs pulse.

of approach is not very scientific. One solution is to study whether there are other bounds besides the Cramer–Rao bound that apply to actual intercept receiver design.

2.8 AOA Information [41–44]

The AOA is a valuable parameter that is used in deinterleaving radar signals since a radar cannot rapidly change its position. Even an airborne radar cannot significantly change its position in the few milliseconds of PRI time. As a result, the AOA measured by an intercept

receiver on the radar is a relatively stable value. Unfortunately, the AOA parameter is also the most difficult one to measure. It requires a large number of antennas and receivers in addition to the necessary AOA measurement circuits. All these antennas and receivers must be matched, either in amplitude or in phase, thus the cost of such systems is usually very high.

A narrowband AOA system can be cost effective. For example, one can measure the frequency of an incoming pulse then tune narrowband receivers connected to different antennas to that frequency to measure the AOA of the next incoming pulse. As mentioned previously, an EW receiver should measure the AOA information on a pulse-by-pulse basis; thus this approach cannot satisfy this requirement. In addition, the receivers must be able to measure the AOA on simultaneous signals, and this requirement makes the design even more difficult.

Commonly used AOA measurement methods in intercept receivers are amplitude comparison and phase comparison methods. If the angle coverage of the two approaches is the same, they produce similar AOA accuracy. An amplitude comparison system can be easily designed to cover a wide angle, while a phase comparison system is used for narrow angle coverage. An airborne amplitude comparison scheme usually covers a 360-deg azimuth and can produce an AOA accuracy of about ± 15 deg. In this approach, the amplitude of every receiver must be matched from the antenna to the AOA measurement circuit. If multiple signal capability is required, even this approach can be very complicated.

The phase comparison system usually covers a much narrower angle range with approximately ± 1 -deg angle accuracy, which is desirable for modern EW applications. A phase measurement system requires all phase measurement channels to be phase matched. If the system must cover a wide instantaneous bandwidth and measure AOA on simultaneous signals, the phase among different channels must be matched, and this is definitely not a trivial task. If the phase cannot be matched among different channels, theoretically a calibration table can be used to remedy this shortcoming. However, if the phase is poorly matched, the calibration table will be very large.

2.9 Outputs of an EW Receiver

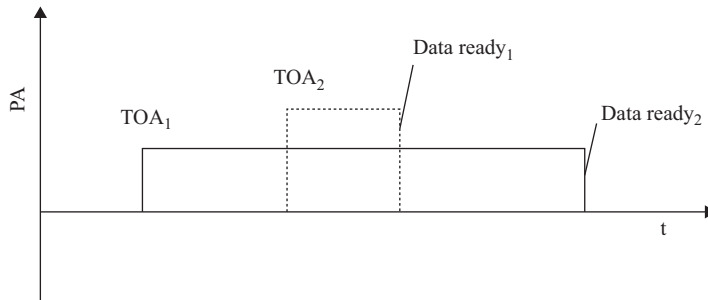
The output of an EW receiver is the PDW. Depending on its design, each receiver will have a unique PDW format. The PDW usually includes all five parameters discussed in Section 2.6, but each parameter may have a different number of bits. For example, a receiver that can only detect the existence of BPSK and chirp signals may report the data shown in Table 2.1. In this example, the word length is 75 bits. In some receivers the word length can be much longer, but usually less than 128 bits.

The range is an approximate value obtained from the resolution and total number of bits. For example, the 32-GHz frequency range is obtained from $2^{15} \times 1$ MHz. It does not represent the capability of the receiver. These PDWs are often passed to the digital processor through three 32-bit words. If there are two simultaneous signals, two PDWs will be generated by the receiver and the two PDWs will have the same TOA values. Thus there is no need to flag simultaneous signals. They can be detected by the TOA readings in the PDWs.

The TOA may be reported in reverse order in some special cases. For example, a long pulse may arrive before a short one, but the trailing edge of the long pulse is after the short

Table 2.1 A typical PDW format.

Parameters	Range	No. of Bits
Frequency	Up to 32 GHz	15 (1-MHz resolution)
Pulse amplitude	Up to 128 dB	7 (1-dB resolution)
Pulse width	Up to 204 μ s	12 (0.05- μ s resolution)
TOA	Up to 50 sec	30 (0.05- μ s resolution)
AOA	360 deg	9 (1-deg resolution)
BPSK signal flag		1
Chirp signal flag		1
Total no. of bits		75

**Figure 2.6** TOA reported in reverse order.

one, as shown in Figure 2.6. Under this condition, the measurement on the short pulse will be completed first and the corresponding PDW will be sent out to the EW processor. However, the TOA data of the short pulse are later than the long pulse, which is reported at the end of the trailing edge of the long pulse. Under this condition, the first reported TOA corresponds to the short pulse, whereas the second reported TOA is meant for the longer pulse. The EW processor must have the ability to process signals that arrive in this order.

2.10 Overview of Analog EW Receivers [45, 46]

Traditionally, EW receivers are classified into six categories by their structures: crystal video, superheterodyne, IFM, channelized, compressive (microscan), and Bragg cell receivers. These receivers are referred to as analog receivers. The input signals are converted into video signals through crystal detectors. These video signals are further processed to generate the PDW, which includes all the desired parameters. The classification can be considered somewhat arbitrary. For example, a channelized receiver may use the superheterodyne technique and a digital receiver may use the channelization approach. A discussion of these types of EW receivers can be found in Tsui [13]. Crystal and superheterodyne receivers cannot process simultaneous signals, therefore these two types of receivers will not be discussed here. An IFM receiver cannot process simultaneous signals either, but the operation concept will be used in later chapters, so it will be included.

Channelized, compressive, and Bragg cell receivers can process simultaneous signals. In all these receivers, a critically important topic is the parameter encoder, which is shown in Figure 2.2. Almost all receiver problems occur in the parameter encoder design. The front-end designs (i.e., the RF input to the video outputs) usually produce satisfactory results. Converting these video outputs into digital frequency data sometimes produces deficiencies, such as reporting an erroneous frequency. These deficiencies often occur in those receivers that can process simultaneous signals.

In almost all the well-designed receivers, the parameter encoder and the RF section are designed as a single unit. The RF components (i.e., filter shape and delay line weighting) must generate the desired video signals to feed the parameter encoder. Many times an RF front end is constructed first, meaning the video signal is available, but it cannot be made into a functioning receiver because a satisfactory encoder design is difficult to achieve.

2.11 IFM Receivers [47–49, 65]

An IFM receiver cannot process simultaneous signals; however, this receiver is very attractive in terms of instantaneous bandwidth, frequency measurement accuracy, size, weight, and cost. This is the type of receiver that can measure frequency accuracy to 1 MHz on a 0.1- μ s pulse. The instantaneous input bandwidth can achieve 16 GHz (from 2 to 18 GHz). Since the IFM receiver has such good performance, it will be briefly discussed here.

Basically, an IFM receiver uses the nonlinear property of crystal detectors to generate the autocorrelation of the input signal. The correlator (or frequency discriminator) is the heart of an IFM receiver. A basic configuration of a correlator is shown in Figure 2.7. The delay line with delay time τ in combination with the correlator generates the autocorrelation of the input signal with lag τ , which can be used to determine the input frequency. Theoretically, one can solve for multiple signals if the autocorrelations with many lags can be obtained. Therefore one should be able to solve the simultaneous signal problem in the IFM receiver. Many attempts have been made to improve its capability to process simultaneous signals, though with only very limited success for the following reason.

In an actual receiver there are four crystal detectors in the correlator. The detectors have a dynamic range of about 15 dB. In order to increase the single frequency dynamic range of the receiver, a limiting amplifier is used in front of the correlator. The limiting amplifier is a

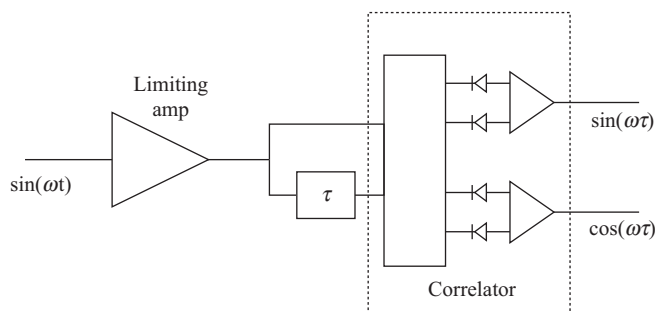


Figure 2.7 A basic IFM receiver.

nonlinear device. If there is only one signal, the strongest output from the nonlinear device is the true signal. This signal is measured by the receiver. If there are multiple inputs at the input of the limiting amplifier, the nonlinear effect cannot be neglected. As a result, the outputs of the correlator are no longer the desired autocorrelation for multiple signals. This is one of the main difficulties in solving the simultaneous signal problem in this type of receiver.

If the autocorrelations can be obtained, some of the high-frequency solution methods discussed in Chapter 11 can be used to solve for multiple signals. Even if these approaches are theoretically viable, they must be implemented in real time.

2.12 Channelized Receivers [40–53]

The idea of a channelized receiver is very simple and uses a filter bank to sort signals with different frequencies. Amplifiers are used after the filter outputs to improve receiver sensitivity. These amplifiers placed after the filter bank can improve sensitivity without affecting the dynamic range. Since one signal appears in one channel after the filter bank, intermodulation (often referred to as intermod) is not a problem. If two signals appear in one channel, this input condition is beyond the capability of the receiver and may generate erroneous frequency information. Two types of amplifiers are often used: limiting amplifiers (or linear amplifiers used at saturation level) and log video amplifiers. Log video amplifiers can be used to measure the pulse amplitude information at the outputs of the filter bank. When limiting amplifiers are used, the amplitude information is lost, thus the pulse amplitude information must be measured somewhere else in the receiver.

To find the center frequency of the input signal, intuitively one will look for the filter with the highest output compared with its adjacent ones. As a result, amplitude comparators between adjacent channels are often used to determine the frequency of the input signal. This approach can successfully provide the correct frequency information if the required instantaneous dynamic range is low. If high instantaneous dynamic range is required, this approach often generates spurious responses. This deficiency can be explained with the help of Figure 2.8,

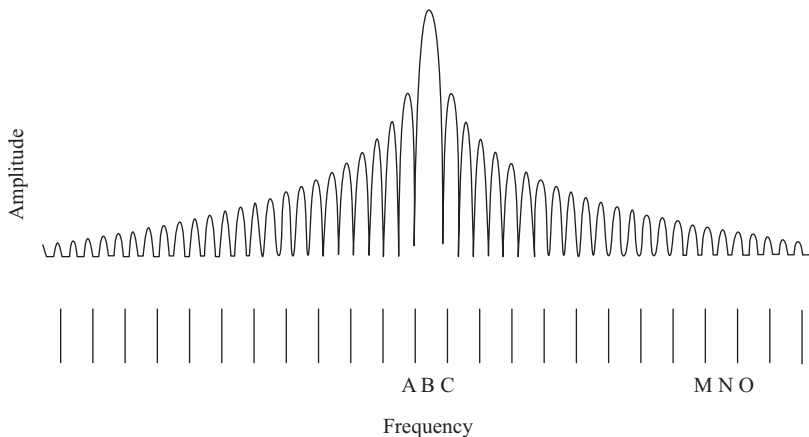


Figure 2.8 Spectrum display and filter bank.

which shows the spectrum of a square pulse. There is one mainlobe and many sidelobes. The energy differences between successive sidelobes that are close to the mainlobe are relatively significant. The energy differences between two successive sidelobes that are far away from the mainlobe are very small.

The amplitude comparison scheme works well when the filters A, B, and C are close to the mainlobe, as shown in Figure 2.8. In this case, the outputs are $A < B > C$. Because the outputs from these filters are far apart in amplitude, this condition can be easily detected and the correct frequency will be reported. On the other hand, the filters M, N, and O are far away from the mainlobe. The outputs for this case should be $M > N > O$. Under this condition, no frequency should be reported. Since the amplitudes of these three filter outputs are very close, any gain imbalance may violate the above conditions. If the outputs are $M < N > O$, a false frequency report will be generated at output N. Balancing the gains among the channels appears to be an impossible task. When the required instantaneous dynamic range is low, outputs from filters far away from the mainlobe are neglected and spurious responses can be avoided.

In many channelized receiver designs, techniques to determine whether a signal is inside or outside of a certain filter are used. These approaches do not compare outputs from adjacent channels, but use the output from one single filter. The time domain response (the transient effect) of a signal passing through a filter is used to make the decision. Circuits following the filter are designed to measure the shape of the output. If the output shape meets certain criteria, the signal frequency is considered inside the filter, otherwise it is outside the filter. In this type of design, the detection filter bandwidth is usually 1.5 times wider than the separation between filters to avoid channel boundary alignment problems. As a result, the frequency resolution is half the separation between filters. These types of approaches are among the most successful.

2.13 Bragg Cell Receivers [54–58]

A Bragg cell receiver uses an optical Bragg cell to perform frequency separation. The input RF signal is converted into an acoustic wave traveling in the Bragg cell, which diffracts a collimated laser beam. The position of the diffracted laser beam is a function of the input frequency. A photodetector array is used to convert the laser output into a video signal. In this arrangement, the input is RF and the outputs are channelized video signals. It is equivalent to the front end of a channelized receiver, including video detectors. The major advantage of a Bragg cell receiver ignoring the parameter encoder is its simplicity. A large number of channels (i.e., 100) can be accommodated in a very few components: a laser, a collimator, two optical lenses, a Bragg cell, and a photodetector array. This arrangement can be built very small.

The major disadvantage of a Bragg cell receiver is that the Bragg cell has optical outputs. With today's technology, it is difficult to place optical amplifiers between the Bragg cell and the photodetectors to improve sensitivity and not affect the dynamic range. Amplifiers can be added in front of the Bragg cell to improve receiver sensitivity. The intermod generated in these amplifiers by simultaneous signals limits the instantaneous dynamic range. Because of the lack of low-cost optical amplifiers, the dynamic range of a Bragg cell receiver is usually low. This type of receiver is also referred to as the power Bragg cell receiver because the

power of the laser output is measured, in contrast to the interferometric Bragg cell receiver discussed in the following paragraph.

The interferometric approach was developed to improve the dynamic range of the Bragg receiver. In this approach, two Bragg cells are used: one as a reference cell and one as the signal cell. The laser beam is divided into two paths, each containing a Bragg cell. The two output beams beat against each other through a photodetector to produce an IF. This IF output is an electric signal, therefore IF amplifiers can be added to improve the dynamic range. Theoretically this approach may improve the dynamic range of the receiver. However, due to the limited power in the source laser and the difficulty in generating a proper reference signal, very limited improvement in dynamic range has been realized. The configuration of an interferometric Bragg cell receiver is very complicated. After the Bragg cells, each channel has a photodetector working as a mixer to generate the desired IF. This IF signal is lowpass (or bandpass) filtered, amplified, and converted into a video signal by a crystal video detector. Taking all the components into consideration, it is probably more complicated than a conventional channelized receiver that uses bandpass filters.

In conclusion, the optical portion of the power Bragg receiver is essentially a combination of a filter bank followed by crystal detectors. In comparison with an RF channelizer, the Bragg cell approach is simpler, but the performance is also inferior.

The most common encoder design for a Bragg cell receiver is the amplitude comparison scheme against adjacent channels. As mentioned before, this approach usually has limited dynamic range. In order to improve the dynamic range of a Bragg cell receiver, not only the optical arrangement should be improved, but the frequency encoder should be studied as well.

2.14 Compressive (Microscan) Receivers [49–61]

In a compressive receiver, the Fourier transform is performed on the input signal to convert signals with different frequencies into short pulses in the time domain. A simple front end of a compressive receiver is shown in Figure 2.9. The input signal is converted into a chirp

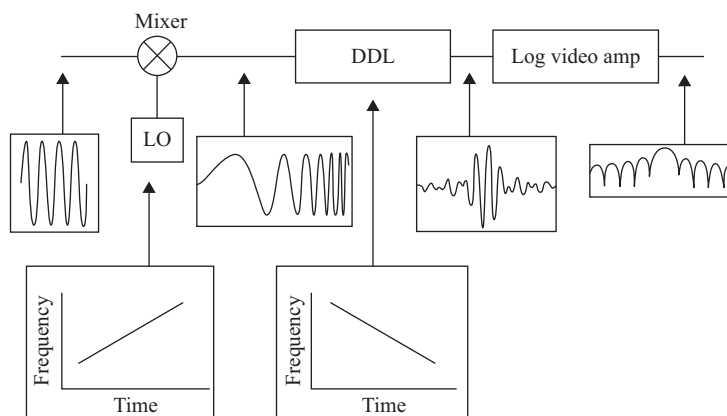


Figure 2.9 A basic compressive receiver.

signal through a mixer fed by an FM local oscillator. The chirp signal is compressed into short pulses through a compressive (or dispersive delay) line. These short pulses pass through a log video amplifier and are converted into video signals. The video circuit must have very wide bandwidth to process the narrow video pulses. The time position of each short output pulse relative to the initiation of the LO sweep represents the frequency of the corresponding input signal.

A parameter encoder is required to convert these video pulses into the desired PDW. Since the video pulses come out in time sequence from one output port, less hardware is required in the parameter encoder when compared with a channelized receiver. However, this hardware must operate at very high speed, equal to the bandwidth of the receiver. In most receivers the input bandwidth equals the bandwidth of the dispersive delay line. Under this condition, if a receiver has an instantaneous bandwidth of 2 GHz, the logic circuit must also operate at 2 GHz. Each compressed pulse has a mainlobe with some sidelobes. The parameter encoder must be able to detect the mainlobe and neglect the sidelobes. Detection of sidelobes will produce spurious signal reports. In general, a pulsed signal can be intercepted in many consecutive scans. The information needed by an EW processor is on a pulse-by-pulse basis and not on a scan-by-scan basis. The parameter encoder must combine all the information generated by each scan within the same pulse and produce the PDW at the end of the pulse.

The most attractive feature of a compressive receiver is its potential to simplify AOA measurement. All the information on the input signal containing amplitude and phase is maintained at the compressed pulse before the log video amplifier. Both amplitude comparison and phase interferometric approaches can be used to measure AOA. Since the information coming out of the receiver is in series, less hardware is required to measure the AOA. For example, assume that four antennas/receivers are required to measure the AOA through phase comparison and each receiver generates 100 frequency resolution cells. Four microscan receivers with four outputs and four phase comparison circuits are needed. If channelized receivers are used to achieve the same results, 400 channels and comparators would be required, which is impractical to build.

2.15 Digital Receivers

Digital receivers are the main subject of this book. Because of advancements in analog-to-digital converters (ADCs) and increases in digital signal processing speed, present research has concentrated on digital EW receivers. In this type of receiver, the input is downconverted into an IF, which is then digitized with high-speed ADCs with a large number of quantization levels. Digital signal processing is then used to produce the desired PDW.

A digital receiver does not have a crystal video detector. The output from the ADC is digital. Some of the major advantages are related to digital signal processing. Once a signal is digitized, the following processing will all be digital. Digital signal processing is more robust because there is no temperature drifting, gain variation, or dc level shifting as in analog circuits, therefore less calibration is required. The frequency resolution can be very fine if high-resolution spectrum estimation techniques can be applied. In many spectrum estimation schemes the results are comparable with the Cramer–Rao bound at high S/N ratios, which analog receivers cannot achieve.

The two areas in a digital EW receiver that need to be investigated are increasing the input instantaneous bandwidth and real-time processing to produce the desired PDW. These requirements can be solved by increasing the ADC and digital signal processing speeds. The Nyquist sampling criterion limits the input bandwidth. In order to cover a 1-GHz bandwidth for real data (contrast to complex data), the ADC must operate at least at 2 GHz. Because of advances in ADCs, the operating speed and number of bits have greatly increased. The allowable receiver bandwidth is directly proportional to the ADC sampling rate and the number of bits is directly related to the dynamic range.

The main problem in a digital EW receiver is processing the ADC output at a rate as high as 2 GHz at 8 bits. One possible approach is to multiplex the ADC output. If an ADC operates at 2000 MHz and the fast Fourier transform (FFT) chip can only operate at 400 MHz, one can divide the output of the ADC into four parallel outputs feeding an FFT chip placed at each of the outputs. Another approach is to use conventional multirate digital filter designs. In this approach, the output of the ADC is also multiplexed and many parallel filters are used to sort the signals.

A brute force approach is to build many narrowband digital receivers. A number of these receivers are combined together to cover a wide instantaneous bandwidth. All the receiver outputs must be properly combined to determine the number of input signals and their center frequencies. In essence, this approach may have design criteria similar to the analog channelized receiver.

A digital EW receiver can be represented in functional blocks, as shown in Figure 2.10. This figure is similar to the analog receiver shown in Figure 2.2. The output from the ADC is digital. These data are in the time domain and must be converted to frequency domain. In the frequency domain, the information is available as spectral lines or spectrum density. However, these outputs do not satisfy the EW requirements. The spectral lines must be converted into the carrier frequencies of the input signals. In order to emphasize this process, a parameter encoder is identified separately from the spectrum estimator. The parameter encoder converts the frequency information into the desired PDW.

Research on digital receivers should be concentrated on many areas, including the sensitivity and dynamic range of receivers as well as the nonlinear effect of the ADC.

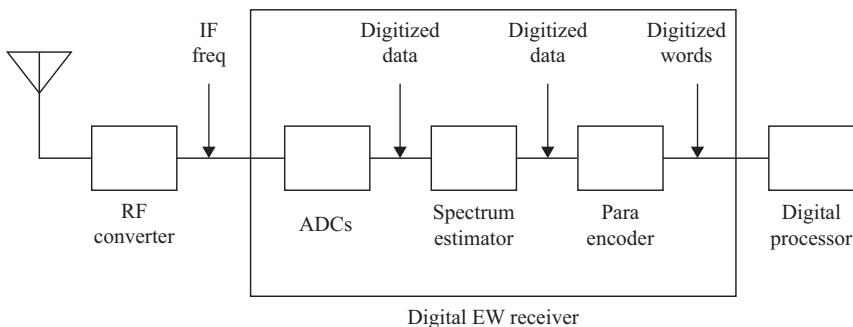


Figure 2.10 Function of a digital EW receiver.

2.16 Characteristics and Performance of EW Receivers [25, 62–63]

The most prominent problem in EW receiver development is the lack of a universally accepted performance standard. Making this matter even worse is the intentional and unintentional reporting of misleading results, causing confusion among researchers in the field. Researchers may not be aware of where the deficiencies in the receivers lie and, as a result, may not know where they should focus their research resources.

Sometimes the performance of an incomplete receiver is reported. For example, the receiver may not produce a PDW in real time. In other words, the reported data may not be for a complete receiver, but from some video outputs or some types of displays. It is not uncommon to have different performances reported on the same receiver. For example, the sensitivity of a wideband receiver can change from -55 to -65 dBm across the band. The optimistic engineer may report the best result and the pessimistic engineer may report the worst result. Worse yet, some person may report -75 dBm by observing the video output. The correct way is to report the sensitivity as the minimum power level at which the receiver can repetitively generate the correct PDW.

In addition, the sensitivity versus frequency or the maximum and minimum values should be reported. For example, there are three different types of dynamic ranges: single signal, third-order intermod, and instantaneous. All these dynamic ranges are important to EW receiver performance. A receiver may have a 70-dB single-signal dynamic range, but only a 20-dB instantaneous dynamic range. If one person reports the best value and another one reports the worst without clarification, one can imagine the potential confusion. Of course, the correct way is to report all three values.

In order to keep the discussion simple and exact, only the definitions that can be measured on receivers with PDW output will be presented here. These definitions can be applied to analog as well as digital receivers. The inputs are limited to one signal and two simultaneous signals, although some receivers can process more than two simultaneous signals. The measurements to obtain these defined values are presented in Chapter 18. These characteristics are as follows.

2.16.1 Single Signal

1. Frequency data resolution: the finest increment in measured frequency data.
2. Accuracy of frequency measurement: the error between the measured frequency and the input frequency.
3. Precision of frequency measurement: the repeatability of the frequency measurement.
4. False alarm rate: the number of false alarms per unit time when there is no signal applied to the input of the receiver.
5. Sensitivity: the lowest signal power that can be properly detected and encoded by the receiver. Properly encoded means the measured parameter must be within a pre-determined tolerance.
6. Dynamic range (single signal): the ratio of power of the strongest signal that the receiver can properly detect without generating spurious responses to the signal at sensitivity level.

7. Pulse amplitude data resolution: the finest increment in measured amplitude data. It is usually measured in decibels.
8. Pulse width data resolution: the finest increment in measured pulse width data. Pulse width is often measured in a nonuniform scale. High pulse width data resolution is used to measure short pulses and low pulse width data resolution is used to measure long pulses.
9. Angle of arrival data resolution: the finest increment in measured AOA data.
10. Time of arrival data resolution: the finest increment in measured TOA data. Because the TOA is referenced to an internal clock in the receiver, it is impractical to compare the measured TOA against the incoming pulses. The common approach is to measure the TOA difference (ΔTOA).
11. Throughput rate: the maximum number of pulses that can be processed by the receiver per unit time.
12. Shadow time: the minimum time between the trailing edge of one pulse and the leading edge of the next that permits the receiver to properly encode both of them. This quantity is usually PW dependent and is defined here at the minimum PW.
13. Latency time: the delay between the arrival time of the pulse at the receiver and the output of the digital word from the receiver.

2.16.2 Two Simultaneous Signals

In order to keep the discussion simple, the following definitions are applicable only for two input signals of the same pulse width and coincident in time.

1. Frequency resolution: the minimum frequency separation of two simultaneous signals with the same incident angle that permits the receiver to properly encode both of them.
2. Spurious free dynamic range: the power ratio of the strongest signal (one of two equal amplitude signals) that the receiver can properly encode without generating detectable third-order intermod to the power at the sensitivity level. When two strong signals at frequencies f_1 and f_2 arrive at the receiver, third-order intermod will be generated. The third-order intermod is often measured with the two input signals kept at the same amplitude. The third-order intermod occurs at frequencies $2f_1 - f_2$ and $2f_2 - f_1$.
3. Instantaneous dynamic range: the power ratio of the maximum and minimum simultaneously received pulses that can be properly encoded by the receiver.
4. Angle of arrival resolution: the minimum angular separation between two sources received simultaneously at the same frequency that permits the receiver to properly encode both of them.

2.17 Potential Trend in EW Receiver Development [13]

It is very difficult to correctly assess future trends in EW receiver development. For example, the Bragg cell receiver was first built as an electronic intelligent (ELINT) receiver around 1974. A unit was used in an airborne system to collect data and the results were very impressive. However, after many years of research, some key problems still do not have satisfactory solutions. The discussion in this section is based on present needs and anticipated technology trends.

2.17.1 Theoretical Problem Solutions

The solutions of two theoretical problems are needed in EW receivers: determination of the optimum bandwidth of an EW receiver and a theoretical bound for a receiver that can process two simultaneous signals. The first problem is very much system oriented, and a solution may not be reached easily from a purely theoretical point of view.

The second problem is to find the instantaneous dynamic range and the frequency resolution if the pulse width and S/N ratio are given. This question should be answered with real-time processing in mind. This is a problem similar to the Cramer–Rao bound. Analog receiver performance is far from the Cramer–Rao bound because a receiver has not yet been designed as a maximum likelihood receiver. It will be useful to find a different bound. It is possible that the bound is receiver design dependent. If this is the case, it is difficult to specify the requirements of a receiver without designing it first.

It is anticipated that future developments of EW receivers will be concentrated in four areas: queuing, compressive, channelized IFM, and digital.

2.17.2 Queuing Receiver

This receiver actually consists of two or more types of receivers: at least one coarse and one fine measurement receiver. The basic idea is to measure one parameter coarsely (i.e., frequency or AOA) and use this information to direct other receivers to further measure the information. There can be many different types of designs. This subject is discussed in Chapter 15.

2.17.3 Compressive Receiver Used for AOA Measurement

As mentioned before, less hardware is needed in a compressive receiver when it is used to measure AOA. The research is expected to concentrate on the phase comparison system because it can produce better AOA accuracy with narrower angle coverage compared with an amplitude comparison system. For this approach, the receivers must be phase matched. One of the anticipated problems is in measuring the phase difference in the compressive pulse. The compressed pulse is very short (e.g., a 1-GHz input bandwidth generates a compressive pulse of 1 ns). It is not an easy task to measure the phase difference in such a short time. One possible approach is to stretch the compressed pulse artificially to provide more time for the measurement. Because the width of the compressed pulse is related to frequency resolution, increasing the compressed pulse width artificially reduces the receiver's capability to separate signals close in frequency. For example, it is possible that a compressive receiver can measure the frequencies of two signals 20 MHz apart, but the AOA measurement circuit may not be able to measure them because the extended compressed pulse width degrades the frequency separation capability.

2.17.4 Channelized IFM Receiver

The concept of a channelized IFM receiver is not new. An IFM receiver can be very small and measure frequency accurately on a short pulse, but it cannot process simultaneous signals. Placing a narrowband filter in front of the receiver to limit the probability of simultaneous signal occurrence seems to be an obvious solution. However, the narrowband

filter will cause transients on the pulsed signal. This effect must be carefully studied. The bandwidth of the filters needs to be determined based on the total number of channels and the minimum pulse width anticipated. The major problem with this concept is in the parameter encoders. One strong signal may be detected in several channels, but the receiver must report the one frequency correctly. On the other hand, two simultaneous signals reaching two adjacent channels must be reported as two signals with the correct frequency information. Usually, when two simultaneous signals reach the same channel, one should expect some probability of erroneous frequency data.

2.17.5 Digital EW Receiver

Because of advancements in ADC technology, ADCs may be used to build a receiver with wide instantaneous bandwidth (2 GHz) and reasonable dynamic range (50 dB) in the near future. Although some spectrum estimation schemes can generate very high frequency resolution, they are usually computation intensive and may not be suitable for real-time applications. In the near future, the FFT might still be the most promising method since high-speed FFT chips are available. Another promising approach is to use decimation in multirate signal processing to build a digital channelized receiver. There is one possible advantage in digital channelization in comparison with an analog approach in that all the channels may be better balanced. It should be noted that because the ADC response is frequency dependent, even the digital channelizer may not have perfectly balanced outputs.

Even in digital receivers, the parameter encoder will remain one of the most important components. This work has seldom been discussed in the literature. After the frequency analysis (i.e., FFT), the carrier frequency of the input signal and its amplitude must be obtained. The frequency encoding scheme should avoid the sidelobes and recognize the mainlobes. It is inevitable in an EW receiver that a strong signal will saturate the ADC; this saturation problem and many other problems not found in a communication receiver must be investigated in the future. One encoder design example is presented in Chapter 17.

2.18 EW Processor

This section will introduce the very basic concept of an EW processor. An EW processor is expected to perform the following functions: deinterleaving, generating PRI data, identifying individual radars, tracking, and revisiting. Each of the functions will be discussed separately.

2.18.1 Deinterleaving

We will use an example to demonstrate this operation. If there are three simple radars with constant PRIs, the radars will emit three stable pulse trains as shown in Figure 2.11(a–c). When an EW receiver intercepts these pulse trains, the result can be represented as in Figure 2.11(d). One can imagine that in this figure the receiver can measure the total number of simultaneous signals. This result is from the interleaving of the three radars. From this figure it is difficult to determine which pulse comes from which radar. If the EW receiver cannot determine the number of simultaneous signals, the result is shown in Figure 2.11(e). Under this condition, it is even more difficult to identify the pulse train because one does not know how many pulses are received at any instant.

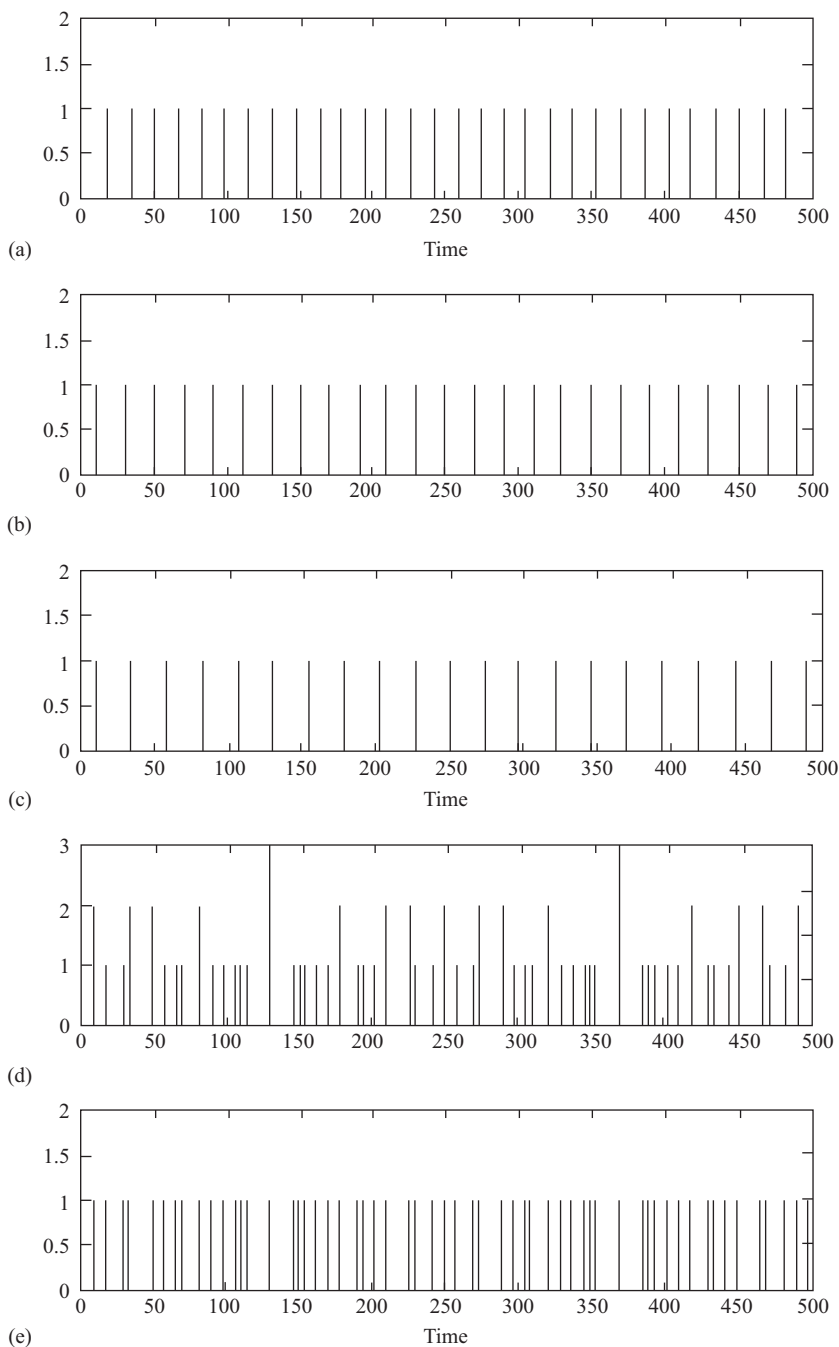


Figure 2.11 Pulse interleaving: (a) radar 1, (b) radar 2, (c) radar 3, (d) intercepted pulses with simultaneous signal identified, and (e) intercepted pulses without simultaneous signal identified.

An EW processor must deinterleave the intercepted pulse train into an individual radar pulse train. In order to solve this problem, each intercepted pulse will be compared to see whether it originates from the same radar. The common parameters used to perform this comparison are the center frequency (or RF), TOA difference (ΔTOA), and the AOA of the received pulses. If two pulses are very close in RF, they can be considered to be from the same radar. In cases where a radar changes its RF on a pulse-by-pulse basis, it is difficult to compare the RF to sort out the pulse train. A similar argument can be applied to the TOA difference. For radars with frequency hopping and agile PRI capabilities, AOA is the most effective parameter to compare. From parameter comparison, the intercepted pulses can be deinterleaved into different radar trains.

Pulse amplitude measured by an EW receiver depends on the directions of the transmitting and receiving antenna, thus it is not a dependable parameter. Multipath disturbs the PW measurement accuracy. Multipath means that one signal arrives at the receiver through different paths (e.g., reflection from a building). Thus PA and PW are usually not used in pulse deinterleaving.

One can consider deinterleaving as a two-dimensional pattern recognition problem. RF and AOA are the parameters that are used to identify patterns. It is anticipated that new parameters may be generated (i.e., components from certain types of transforms) and that these new parameters may be easier to obtain than the AOA. Although research in this area has begun, no real-time system that can operate with EW receivers has been reported.

Deinterleaving is the major function of an EW processor. It should be designed for maximum efficiency to perform this operation. In other words, the processor should be able to perform deinterleaving with a minimum number of received pulses.

2.18.2 PRI Generation

Once the pulse trains from individual radars are identified, the TOA information can be used to generate the PRI. As mentioned before, the PRI is the TOA difference between successive pulses of a deinterleaved pulse train. If the TOA difference is used in deinterleaving, the PRI information is already available. Some radars have stable PRIs, as shown in Figure 2.11, while other radars have staggered PRIs, which means there are several PRI values. Some radars even have random (or jittered) PRIs.

2.18.3 Radar Identification

The RF, PRI, and pulse width can be considered as the intrinsic characteristics of a radar since they are generated by the radar. They can be used to determine the type of the radar. In contrast, pulse amplitude and AOA are not generated by the radar, but are functions of the relative positions of the radar and intercept receiver. From the RF, PRI, and PW one can identify the type of the radar. If it is a threat radar, one can determine the jamming technique to use against it.

2.18.4 Tracking

An EW processor can process only limited pulse density. This density is usually lower than the pulse density an EW receiver can intercept. If an EW processor can process

100,000 pulses/s and a receiver can intercept 1,000,000 pulses/s, the received pulse density will choke the processor. However, once a radar pulse train is identified, it is no longer necessary to deinterleave on these pulses. Trackers are built to prevent pulses that are a continuation of an identified train from reaching the deinterleaving portion of the processor.

Trackers can be considered as two-dimensional filters that stop pulses from getting to the deinterleaving portion of the processor. One dimension is the PRI in the time domain and the other one is the RF in the frequency domain. It gates pulses in certain periods of time and within certain RF ranges. If the receiver cannot produce frequency information (i.e., a crystal video receiver), the tracker can only work in the time domain on the PRI of the pulse train. A tracker usually tracks one signal, but a processor can have many trackers. In some cases, trackers are considered part of the active ECM because it is part of the technique generator. A technique generator provides the desired video pulses modulated by RF signals for jamming.

2.18.5 Revisiting

Once a pulse train is identified as a threat, a jammer can be turned on to jam the signal. At the same time the pulse train is tracked, the information will not reach the deinterleaving circuit. As a result, one does not know whether the signal is still transmitting (or being intercepted by the receiver). It is important to know whether the signal being jammed is still in operation. Otherwise, energy may be wasted on jamming a signal that is no longer in operation. In order to find out whether the signal is still being intercepted, the tracker temporarily stops tracking and passes the information to the processor. This processing is often called revisiting. If the signal is still being intercepted, the parameters measured on the pulse are presented in the processor and the tracker can continue to track the signal. If the signal is no longer being intercepted, the jammer and tracker actions can be stopped.

2.19 EW Receiver Design Goals [64]

Theoretical limits such as the Cramer–Rao bound discussed in Section 2.7 should be very useful. Such bounds, however, are not available for EW receivers. Therefore the performance discussed in this section can be considered as a goal. Whether these goals can be achieved simultaneously is uncertain. Table 2.2 lists the goals. The instantaneous bandwidth, the spatial coverage, and the number of simultaneous signals usually can be achieved if the receiver is so designed.

The data length used for signal processing usually matches the minimum pulse width. This choice can provide the best S/N ratio [64]. It also simplifies the parameter encoder design. Short data length processing is equivalent to a wideband filter, which may have a simpler parameter encoder design, but it will have a low S/N ratio and low-frequency data resolution. Long data processing is equivalent to narrowband filters. A narrowband filter will disturb a pulsed signal both in the carrier frequency and the pulse shape. As a result, the measurements on the carrier frequency, the pulse amplitude, and pulse width may not be accurate. In order to cover the same input bandwidth, many parallel narrowband channels are needed, therefore it is difficult to build the parameter encoder. A short pulse can produce outputs from many filters. Under this condition, not only the carrier frequency is difficult to determine, but also the number of signals.

Table 2.2 EW receiver performance goals.

Instantaneous bandwidth		>1 GHz
Spatial field of view		4π solids angle
Simultaneous signal capability		up to 4
Minimum pulse width		100 ns ^a
Sensitivity		−65 to −90 dBm ^b
Dynamic range	Single-signal spur free	75 dB ^a
	Instantaneous	50 dB ^a
	Two-signal spur free	55 dB ^a
Two-signal frequency resolution		20 MHz ^a
Parameter measurement precision	Radio frequency (RF)	1 MHz ^a
	Angle of arrival	1 deg ^a
	Time resolution	1 ns–1 μs ^a
	Pulse amplitude	1 dB ^a
Real-time operation (latency time)		<2 μs ^a

^aThis can be considered a design goal.
^bIt is desirable to have higher sensitivity on longer pulse.

2.20 Summary

Although research in EW receivers is an old topic, there are still many problems that need to be solved. In the past, most research has been concentrated on analog receivers. The main effort was to improve the performance of receivers that could process simultaneous signals, such as channelized, compressive, and Bragg cell receivers. Because of technical difficulties and funding limitations, very few receivers of the mentioned types have actually been built. As far as research is concerned, the EW area is still an open field. It is anticipated that many interesting problems will be investigated and that research and development of digital EW receivers will accelerate.

References

[1] Boyd JA, Harris DB, King DD, Welch HW, Jr. (eds.). *Electronic Countermeasures*. Los Altos, CA: Peninsula Publishing; 1978.

[2] Fitts RE (eds.). *The Strategy of Electromagnetic Conflict*. Los Altos, CA: Peninsula Publishing; 1980.

[3] Price A. *The History of U.S. Electronic Warfare*, **vol. 1**. Alexandria, VA: Association of Old Crows; 1984.

[4] Davies CL, Hollands P. ‘Automatic processing for ESM’. *IEE Proceedings-F: Radar and Signal Processing* 1982;**129**(3):164–171.

[5] Hovanessian SA. ‘Noise jammers as electronic countermeasures’. *Microwave Journal* 1985; **28**(9):113–116.

[6] Mardia HK. ‘New techniques for the deinterleaving of repetitive sequence’. *IEE Proceedings-F: Radar and Signal Processing* 1989;**136**(4):149–154.

- [7] Skolnik MI. *Introduction to Radar Systems*. New York: McGraw-Hill; 1962.
- [8] Skolnik MI (ed.). *Radar Handbook*. New York: McGraw Hill; 1970.
- [9] Stremler FG. *Introduction to Communication Systems*, 2nd ed. Reading, MA: Addison-Wesley; 1982.
- [10] Wiley RG. *Electronic Intelligence: The Analysis of Radar Signals*. Norwood, MA: Artech House; 1982.
- [11] Rohde UL, Bucher TTN. *Communications Receivers Principles and Design*. New York: McGraw Hill; 1988.
- [12] Ziemer RE, Tranter WH. *Principles of Communications: Systems, Modulation and Noise*, 2nd ed. Boston: Houghton Mifflin; 1985.
- [13] Tsui JBY. *Microwave Receivers with Electronic Warfare Applications*. New York: John Wiley & Sons; 1986.
- [14] Cook CE, Bernfeld M. *Radar Signals: An Introduction to Theory and Application*. New York: Academic Press; 1967.
- [15] Dixon RC. *Spread Spectrum Systems*. New York: John Wiley & Sons; 1976.
- [16] Barton DK. *Radar System Analysis*. Norwood, MA: Artech House; 1976.
- [17] Stimson GW. *Introduction to Airborne Radar*. El Segundo, CA: Hughes Aircraft; 1983.
- [18] Rogers JAV. 'ESM processor system for high pulse density radar environments'. *IEEE Proceedings-F: Communications, Radar and Signal Processing* 1985;**132**(7):621–625.
- [19] Eaves JL, Reedy EK (eds.). *Principles of Modern Radars*. New York: Van Nostrand Reinhold; 1987.
- [20] Rappolt F, Stone N. 'Receivers for signal acquisition'. *Microwave Journal* 1977;**20**(1):29–33.
- [21] Hoffmann CB, Baron AR. 'Wideband ESM receiving systems I'. *Microwave Journal* 1980;**23**:24, 26, 28, 30–34.
- [22] Cochrane JB, Markl FA. 'Broadband building blocks shape tomorrow's warning receivers'. *Microwaves* 1980;**November**:92–95.
- [23] Hinshaw J, Carpenter JA. 'A primer on digital output detector video receiver systems'. *Microwave Journal* 1984;**27**(10):151–152, 154–156.
- [24] Moore RA, Marinaccio RE. 'Advancing EW system strategies and supporting technologies'. *Microwave Journal* 1986;**February**:26.
- [25] Watson R. 'Receiver dynamic range. I – guidelines for receiver analysis'. *Microwave & RF* 1986;**25**:113–116, 118, 119, 122.
- [26] Lochhead DL. 'Receivers and receiver technology for EW systems'. *Microwave Journal* 1986;**29**:139–140, 142 144, 146, 148.
- [27] Steinbrecher DH. 'Achieving maximum dynamic range in a modern receiver'. *Microwave Journal* 1985;**28**(9):129–130, 134, 136, 140, 142, 146, 150.
- [28] Engelson M. 'Sharpen pulse signal measurement accuracy'. *Microwave & RF* 1986;**25**(8): 81–83.
- [29] Ball F. 'Measure the real impulse bandwidth'. *Microwave & RF* 1987;**26**(1):93–96

- [30] Ackroyd MH. 'Instantaneous and time varying spectra—an introduction'. *Radio and Electronic Engineer* 1970;**39**(3):145–152.
- [31] Boashash B. 'Estimating and interpreting the instantaneous frequency of a signal. I. Fundamentals'. *Proceedings of the IEEE* 1992;**80**(4):520–538.
- [32] Boashash B. 'Estimating and interpreting the instantaneous frequency of a signal. II. Algorithms and applications'. *Proceedings of the IEEE* 1992;**80**(4):540–568.
- [33] Slepian D. 'Estimation of signal parameters in the presence of noise'. *Transactions of the IRE Professional Group on Information Theory* 1954;**3**(3):68–89.
- [34] Rife DC, Vincent GA. 'Use of the discrete Fourier transform in the measurement of frequencies and levels of tones'. *Bell System Technical Journal* 1970;**49**(2):197–228.
- [35] Rife DC. *Digital Tone Parameter Estimation in the Presence of Gaussian Noise*. Ph.D. dissertation, Polytechnic Institute of Brooklyn, 1973.
- [36] Rife DC, Boorstyn RR. 'Single-tone parameter estimation from discrete-time observations'. *IEEE Transactions on Information Theory* 1974;**20**(5):591–598.
- [37] Rife DC, Boorstyn RR. 'Multiple tone parameter estimation from discrete-time observations'. *Bell System Technical Journal* 1976;**55**(9):1389–1410.
- [38] Friedlander B, Porat B. 'A general lower bound for parametric spectrum estimation'. *IEEE Transactions on Acoustic, Speech, Signal Processing* 1984;**32**(4):728–733.
- [39] Tsui JBY, Thompson MH, McCormick W. 'Theoretical limit on instantaneous dynamic range of EW receivers'. *Microwave Journal* 1987;**30**:147.
- [40] Tsui JBY. *Digital Microwave Receivers: Theory and Concepts*. Norwood, MA: Artech House; 1989.
- [41] Earp CW, Godfrey RM. 'Radio direction-finding by the cyclical differential measurement of phase'. *Journal of the Institute of Electrical Engineers – Part IIIA: Radiocommunication* 1947;**94**(15):705–721.
- [42] Chubb E, Grindon JR, Venters DC. 'Omnidirectional instantaneous direction finding system'. *IEEE Transactions on Aerospace and Electronic Systems* 1967;**AES-3**(2):250–256.
- [43] Baron AR, Davis KP, Hofmann CP. 'Passive direction finding and signal location'. *Microwave Journal* 1982;**25**(9):59–76.
- [44] Mosko JA. 'An introduction to wideband, two-channel direction finding systems'. *Microwave Journal* 1984;**27**:91–92, 96–98, 100.
- [45] Harper T. *Hybridization of Competitive Receivers*. Tech-Notes, **vol. 7**. Palo Alto, CA: Watkins-Johnson; 1980.
- [46] Tsui JBY. 'An introduction to EW microwave receivers'. *Journal of Electronic Defense* 1989;**12**:39.
- [47] Cumming RC, Myers GA. *Performance of Receivers and Signal Analyzers Using Broadband Frequency Sensitive Devices*. Technical Report 1905-1. Stanford, CA: Stanford Electronics Laboratories; 1967.
- [48] East PW. 'Design techniques and performance of digital IFM'. *IEE Proceedings-F: Communications, Radar and Signal Processing* 1982;**129**(3):154–163.

- [49] Bowler BL. 'Tradeoffs in digital IFM receiver design'. Presented at the Joint DADC/Dmpire AOC Technical Seminar, Griffiss Air Force Base, Rome, NY, Nov. 4, 1982.
- [50] Harper T. 'New Trends in EW Receivers'. *Countermeasures* 1976;**December/January**:34.
- [51] 'The channelized receiving systems'. *Microwave System News* 1975/1976;**6**:63.
- [52] Hennessy P, Quick JD. 'The channelized receiver comes of age'. *Microwave System News* 1979;**9**(7):36.
- [53] Anderson GW, Webb DC, Spezio AE, Lee JN. 'Advanced channelization technology for RF microwave and millimeterwave applications'. *Proceedings of the IEEE* 1991;**79**(3):355–388.
- [54] Goodman JW. *Introduction to Fourier Optics*. New York: McGraw-Hill; 1968.
- [55] Chang IC. 'I. Acoustooptic devices and applications'. *IEEE Transactions on Sonics and Ultrasonics* 1976;**23**(1):2–21.
- [56] Lugt AV. 'Interferometric spectrum analyzer'. *Applied Optics* 1981;**20**(16):2770–2779.
- [57] Wilby WA, Gatenby PV. 'Theoretical study of the interferometric Bragg-cell spectrum analyser'. *IEE Proceedings J-Optoelectronics* 1986;**133**(1):47–59.
- [58] Goutzoulis AP, Abramovitz IJ. 'Digital electronics meets its match'. *IEEE Spectrum* 1988;**25**(8):21–25.
- [59] White WD. *Signal Translation Apparatus Utilizing Dispersive Network and the Like, for Panoramic Reception, Amplitude-Controlling Frequency Response, Signal Frequency Gating, Frequency Time Domain Conversion, etc.* U.S. Patent 2,954,465, September 27, 1960.
- [60] Kincheloe WR. *The Measurement of Frequency With Scanning Spectrum Analyzers*, Technical Report 557-2. Stanford, CA: Stanford Electronics Laboratories; 1962.
- [61] Daniels WD, Churchman M, Kyle R, Skudera W. 'Compressive receiver technology'. *Microwave Journal* 1986;**29**(4):175–176, 178.
- [62] Watson R. 'Receiver dynamic range. II – Use one figure of merit to compare all receivers'. *Microwave & RF* 1987;**26**:99–100, 102.
- [63] Tsui JBY, Shaw RL, Davis RL. 'Performance standards for wideband EW receivers'. *Microwave Journal* 1989;**32**(2):46, 48, 50, 52, 54.
- [64] Lawson JL, Uhlenbeck GE (eds.). *Threshold Signals*. New York: McGraw-Hill; 1950:199–210.
- [65] East PW. 'Fifty years of instantaneous frequency measurement'. *IET Radar, Sonar & Navigation* 2012;**6**(2):112–122.

Fourier Transform

3.1 Introduction

In this chapter the continuous Fourier transform and discrete Fourier transform (DFT) will be discussed. The concepts presented in this chapter serve as the foundation of wideband receiver processing techniques presented in this book. There are plenty of good books on this subject and a few of them are listed at the end of this chapter. Readers with a background in this area can skip this chapter.

3.2 Fourier Series and Continuous Fourier Transform [1–4]

The Fourier series was introduced in 1807 by a French mathematician and physicist, Jean Baptiste Joseph de Fourier (1768–1830). He suggested that any arbitrary function, defined over a finite interval by any piecewise graph, continuous or discontinuous, could be represented as an infinite sum of continuous functions such as sine and cosine. Although almost all the members of the French Academy questioned its validity, but it turned out to be one of the most powerful tools in signal processing.

The Fourier series can be explained as follows. A periodic real-valued function $x(t)$ can be represented by the sum of an infinite number of sinusoidal waves with discrete frequencies. The Fourier series can be written in several forms. One of these is

$$x(t) = A_0 + \sum_{n=1}^{\infty} A_n \cos \frac{2\pi n t}{T} + \sum_{n=1}^{\infty} B_n \sin \frac{2\pi n t}{T}, \quad (3.1)$$

where T is the period of $x(t)$ and A_0 is the average value of $x(t)$, which can be found as

$$A_0 = \frac{1}{T} \int_{-T/2}^{T/2} x(t) dt. \quad (3.2)$$

The constants A_n and B_n can be determined as

$$A_n = \frac{2}{T} \int_{-T/2}^{T/2} x(t) \cos \frac{2\pi n t}{T} dt, \quad B_n = \frac{2}{T} \int_{-T/2}^{T/2} x(t) \sin \frac{2\pi n t}{T} dt. \quad (3.3)$$

Because an exponential term can be decomposed into sine and cosine functions as

$$e^{-j\theta} = \cos \theta - j \sin \theta, \quad (3.4)$$

where $j = \sqrt{-1}$.

The Fourier series of a complex-valued signal can be represented as

$$x(t) = \sum_{n=-\infty}^{\infty} X_n e^{\frac{j2\pi nt}{T}}, \quad (3.5)$$

where X_n can be determined as

$$X_n = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{\frac{-j2\pi nt}{T}} dt. \quad (3.6)$$

When $x(t)$ is a real-valued signal, (3.5) can be written in the format of (3.1). Plug (3.6) into (3.5) and rearrange and you get

$$x(t) = \sum_{n=-\infty}^{\infty} \frac{1}{T} \left(\int_{-T/2}^{T/2} x(t) e^{\frac{-j2\pi nt}{T}} dt \right) e^{\frac{j2\pi nt}{T}}. \quad (3.7)$$

For an aperiodic signal, its period T can be considered as infinite. When T in (3.7) approaches infinity, n/T becomes a continuous variable f , $1/T$ becomes the differential df , and the summation in (3.7) becomes an integral and results in

$$x(t) = \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \right) e^{j2\pi ft} df. \quad (3.8)$$

The continuous Fourier transform of a function $x(t)$ is thus defined as

$$F[x(t)] \equiv X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt, \quad (3.9)$$

where f represents frequency. The inverse continuous Fourier transform is defined as

$$F^{-1}[X(f)] \equiv x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df. \quad (3.10)$$

Equation (3.9) can also be written as

$$X(f) = \int_{-\infty}^{\infty} x(t) \cos(2\pi ft) dt - j \int_{-\infty}^{\infty} x(t) \sin(2\pi ft) dt. \quad (3.11)$$

This equation can be used to find the properties of even and odd functions under a Fourier transform.

In order to have a physical picture, one can imagine that a periodic function in the time domain can be represented by an infinite number of frequencies in the frequency domain and these frequencies are discrete. This is the Fourier series. If the function in the time domain is no longer a periodic function, that is, the period is infinite, it can be represented by an infinite number of frequencies and these frequencies are continuous.

3.3 Useful Functions [5–11]

Three important functions—rectangular, impulse, and comb functions—are introduced in this section and will be extensively used throughout this book.

3.3.1 Rectangular Function

A rectangular function is defined as

$$\text{rect}(t/T) = \begin{cases} A, & \text{for } -\frac{T}{2} < t < \frac{T}{2} \\ 0, & \text{elsewhere} \end{cases} \quad (3.12)$$

This function is an even function of t (symmetric with respect to $t = 0$). However, its product with a sine function is antisymmetric with respect to $t = 0$, because the sine function is antisymmetric with respect to $t = 0$. The integral of an antisymmetric function of t from $t = -\infty$ to ∞ is zero, because the positive and negative parts cancel each other out. Thus the Fourier transform of $\text{rect}(t/T)$ is

$$\begin{aligned} \text{RECT}(f) &= A \int_{-\frac{T}{2}}^{\frac{T}{2}} e^{-j2\pi ft} dt = A \int_{-\frac{T}{2}}^{\frac{T}{2}} \cos(2\pi ft) dt \\ &= \frac{A \sin(\pi f T)}{\pi f} = \frac{AT \sin(\pi f T)}{\pi f T} = AT \text{sinc}(\pi f T). \end{aligned} \quad (3.13)$$

The formula is represented using a sinc function, which is defined as

$$\text{sinc}(x) = \frac{\sin(x)}{x}. \quad (3.14)$$

The functions $\text{rect}(t/T)$ and $\text{RECT}(f)$ are shown in Figure 3.1. In the figure, $A = 2$ and $T = 1$. From (3.13) and (3.14), the first zero of $\text{RECT}(f)$ occurs at $f = 1/T$ (or at $\sin(\pi) = 0$). Thus the width of the mainlobe is $2/T$.

3.3.2 Impulse Function

An impulse function is represented by the symbol $\delta(t)$. Sometimes it is also referred to as the Dirac delta function. The output of a system with an impulse function as the input is called

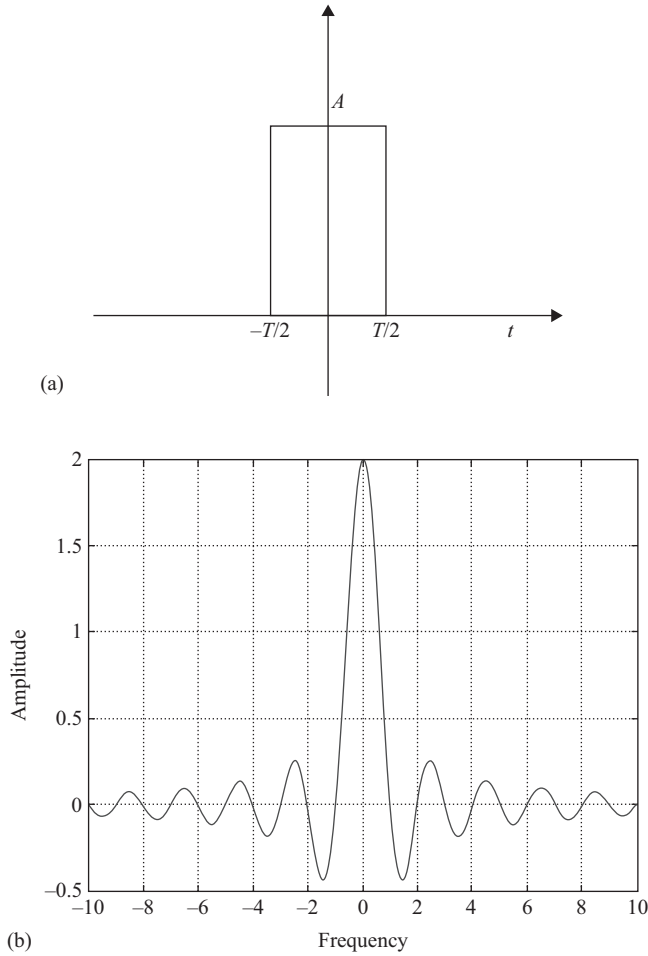


Figure 3.1 A rectangular function and its Fourier transform: (a) time domain and (b) frequency domain.

the impulse response of the system. An impulse function $\delta(t)$ can be considered as a very narrow rectangular function at $t = 0$, as shown in Figure 3.1(a), with $A = 1/T$ and T approaching zero. The area of this rectangular function is 1. The function $\delta(t - t_0)$ is the impulse function at $t = t_0$. Integrating the impulse function, we get its result as

$$\int_a^b \delta(t - t_0) dt = \begin{cases} 1, & \text{for } a \leq t_0 \leq b \\ 0, & \text{otherwise} \end{cases}. \quad (3.15)$$

If the integral includes the impulse function, the integral is unity, otherwise the integral is zero. The impulse function is only conceptual since $\delta(t)$ cannot be generated in practice. However, it is very useful for conceptualization of the digitization process; one can think of “sampling” as multiplication of the input function by a periodic train of impulses.

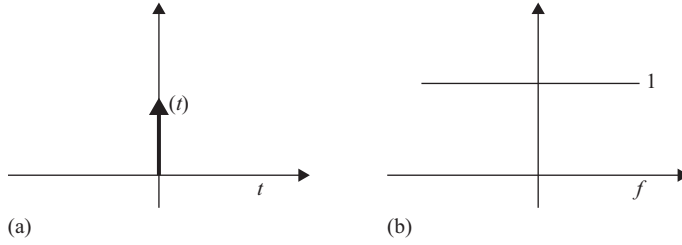


Figure 3.2 An impulse function and its Fourier transform: (a) time domain and (b) frequency domain.

The impulse function can also be defined more generally by its sampling property as

$$\int_a^b x(t)\delta(t-t_0)dt = \begin{cases} x(t_0), & \text{for } a \leq t_0 \leq b \\ 0, & \text{otherwise} \end{cases}, \quad (3.16)$$

which represents a specific value of $x(t)$ at $t = t_0$.

From the definition in (3.9), the Fourier transform of $\delta(t)$ can be determined as

$$X(f) = \int_{-\infty}^{\infty} \delta(t)e^{-j2\pi ft}dt = e^0 = 1. \quad (3.17)$$

This result is shown in Figure 3.2.

For the regular function covered in the previous subsection, when T approaches infinity, its Fourier transform approaches $A\delta(f)$. In other words, the Fourier transform of a constant function is an impulse and the Fourier transform of an impulse function is a constant.

The inverse Fourier transform of the impulse function can be found from (3.10) as follows:

$$\begin{aligned} \delta(t) &= \int_{-\infty}^{\infty} e^{j2\pi ft} df = \lim_{f \rightarrow \infty} \frac{e^{j2\pi ft}}{j2\pi t} \Big|_{-f}^f = \lim_{f \rightarrow \infty} \frac{\cos(2\pi ft) + j\sin(2\pi ft)}{j2\pi t} \Big|_{-f}^f \\ &= \lim_{f \rightarrow \infty} \frac{\sin(2\pi ft)}{2\pi t} \Big|_{-f}^f = \lim_{f \rightarrow \infty} \frac{2\sin(2\pi ft)}{2\pi t} = \lim_{f \rightarrow \infty} \frac{\sin(2\pi ft)}{\pi t}. \end{aligned} \quad (3.18)$$

One can consider this as another definition of the impulse function; that is,

$$\delta(t) = \lim_{f \rightarrow \infty} \frac{\sin(2\pi ft)}{\pi t}. \quad (3.19)$$

The time-shifted impulse function can be represented as

$$\delta(t-t_0) = \lim_{f \rightarrow \infty} \frac{\sin[2\pi f(t-t_0)]}{\pi(t-t_0)}. \quad (3.20)$$

More examples of the impulse function are illustrated in later sections.

3.3.3 Comb Function

A comb function is defined as the sum of an infinite number of uniformly distributed impulse functions; they are shaped like a comb. A comb function is shown in Figure 3.3(a). The function can be defined in either the time or frequency domain. Mathematically, the comb function in the time domain can be written as

$$x(t) = \text{comb}(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT), \quad (3.21)$$

where T is the separation between the δ functions.

The Fourier transform of the comb function can be obtained from directly integrating the above equation. The following manipulation demonstrates the approach [5]:

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(t - nT) e^{-j2\pi ft} dt \\ &= \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(t - nT) e^{-j2\pi ft} dt = \sum_{n=-N}^N e^{-j2\pi fnT} \Big|_{N \rightarrow \infty}. \end{aligned} \quad (3.22)$$

Using the relation

$$\sum_{n=-N}^N z^n = \frac{z^{N+1} - z^{-N}}{z - 1}, \quad (3.23)$$

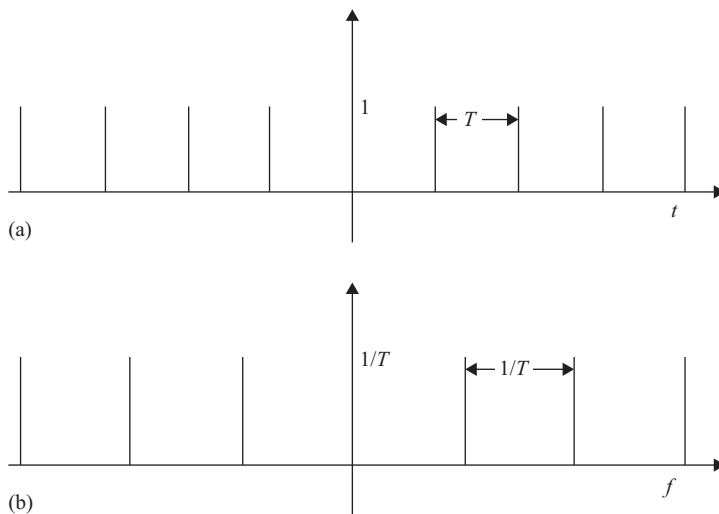


Figure 3.3 A comb function and its Fourier transform: (a) time domain and (b) frequency domain.

(3.22) can be written as

$$X(f) = \frac{e^{-j2\pi f(N+1)T} - e^{j2\pi fNT}}{e^{-j2\pi fT} - 1} = \frac{e^{-j2\pi fT/2} \left[e^{-j2\pi f(N+\frac{1}{2})T} - e^{j2\pi f(N+\frac{1}{2})T} \right]}{e^{-j2\pi fT/2} \left[e^{-j2\pi f(\frac{1}{2})T} - e^{j2\pi f(\frac{1}{2})T} \right]} = \frac{\sin[2\pi(N + \frac{1}{2})fT]}{\sin(\pi fT)}. \quad (3.24)$$

In this function, f is the variable. The result of this equation is shown in Figure 3.4. In this figure, $N = 7$ and f changes from -3.9 to 3.9 . Where $f = n/T$, whenever n is an integer, there is a peak. This figure has the rough shape of a comb. The last step is to prove that when $N \rightarrow \infty$, the result of (3.24) approaches a δ function. The result of the above function can be written as

$$\left. \frac{\sin[2\pi(N+1)fT]}{\sin(\pi fT)} \right|_{N \rightarrow \infty} = \frac{\sin[2\pi(N+1)(f - \frac{n}{T})T]}{f - \frac{n}{T}} \frac{f - \frac{n}{T}}{\sin(\pi fT)} \Big|_{N \rightarrow \infty}. \quad (3.25)$$

Using the result of (3.20), one of the definitions of a δ function, the first part of the right side of (3.25) is

$$\lim_{N \rightarrow \infty} \frac{\sin[2\pi(N+1)(f - \frac{n}{T})T]}{f - \frac{n}{T}} = \pi \delta\left(f - \frac{n}{T}\right). \quad (3.26)$$

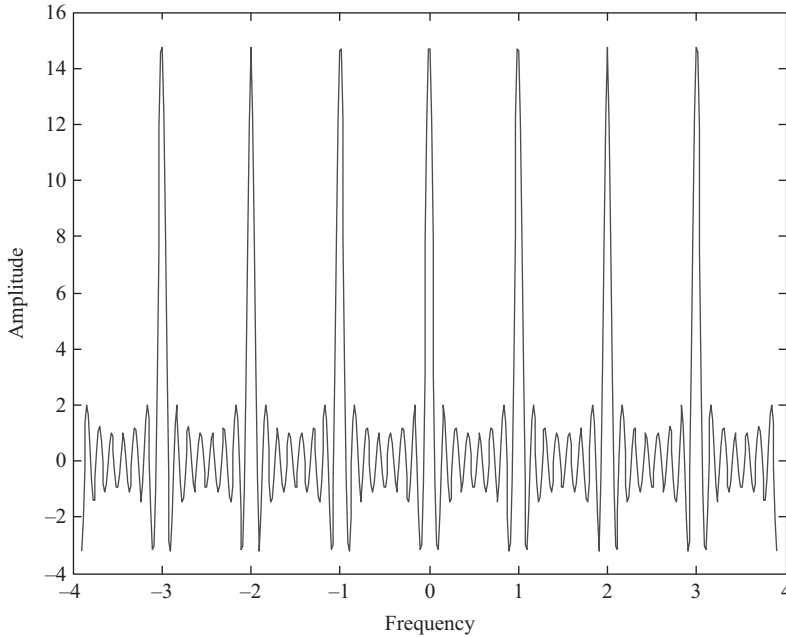


Figure 3.4 Plot of $\sin[2\pi(N + 1/2)fT]/\sin(\pi fT)$.

The second part of the right side of (3.25) can be calculated from L'Hospital's rule as

$$\left. \frac{f - \frac{n}{T}}{\sin(\pi f T)} \right|_{f=\frac{n}{T}} = \left. \frac{\frac{d(f - \frac{n}{T})}{df}}{\frac{d[\sin(\pi f T)]}{df}} \right|_{f=\frac{n}{T}} = \left. \frac{1}{\pi T \cos(\pi f T)} \right|_{f=\frac{n}{T}} = \frac{1}{\pi T}. \quad (3.27)$$

Thus any peak can be written as

$$\frac{\sin[(N+1)2\pi f T]}{\sin(\pi f T)} = \frac{\delta(f - \frac{n}{T})}{T}. \quad (3.28)$$

Thus the Fourier transform of the comb function is another comb function with an amplitude of $1/T$ and a separation of $1/T$, as shown in Figure 3.3(b). This result can be mathematically written as

$$X(f) = \mathcal{F} \left[\sum_{n=-\infty}^{\infty} \delta(t - nT) \right] = \frac{1}{T} \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T}\right). \quad (3.29)$$

This result is very important for processing radar signals, because a radar signal with a stable pulse repetition frequency (PRF) can be represented by a comb function.

3.4 Properties of the Fourier Transform [5–12]

Some of the properties of the Fourier transform are discussed in this section. If the relation is obvious, no proof will be provided.

3.4.1 Linearity

The Fourier transform is a linear operator, which means

$$\mathcal{F}[ax(t) + \beta y(t)] = aX(f) + \beta Y(f). \quad (3.30)$$

Using the definition of the Fourier transform, this property can be easily proved.

3.4.2 Even and Odd Functions

This property can be obtained by observing (3.11). If $x(t)$ is an even function, the second term on the right (the imaginary part) equals zero and the output is real as

$$X(f) = \int_{-\infty}^{\infty} x(t) \cos(2\pi f t) dt. \quad (3.31)$$

Since $\cos(\theta) = \cos(-\theta)$, $X(f)$ is symmetrical with respect to $f = 0$.

If $x(t)$ is an odd function in (3.11), the first term on the right (the real part) equals zero and the output is imaginary as

$$X(f) = -j \int_{-\infty}^{\infty} x(t) \sin(2\pi ft) dt. \quad (3.32)$$

Since $\sin(\theta) = -\sin(-\theta)$, $X(f)$ is antisymmetric with respect to $f = 0$.

3.4.3 Duality

Duality means that if the function in the time domain is $x(t)$ and its Fourier transform is $X(f)$, then $X(t)$ has a Fourier transform of $x(-f)$. The proof of this property can be accomplished from the inverse Fourier transform shown in (3.10). The proof is very simple and only requires three steps and a change of notations for two variables. In (3.10) one can treat both variables t and f as dummy variables. The first step is to replace f by t' . The result is

$$x(t) = \int_{-\infty}^{\infty} X(t') e^{j2\pi t't} dt'. \quad (3.33)$$

The second step is to replace t by $-f$. The result is

$$x(-f) = \int_{-\infty}^{\infty} X(t') e^{-j2\pi ft'} dt'. \quad (3.34)$$

Since t' is a dummy variable, it can be written as t , thus this equation can be written as

$$x(-f) = \int_{-\infty}^{\infty} X(t) e^{-j2\pi ft} df, \quad (3.35)$$

which is the desired result.

3.4.4 Conjugation and Conjugation Symmetry

If the function in the time domain is $x(t)$ and its Fourier transform is $X(f)$, then the Fourier transform of its complex conjugate, $x^*(t)$ is $X^*(-f)$. Taking the complex conjugate on both sides of (3.9), we get

$$X^*(f) = \left[\int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \right]^* = \int_{-\infty}^{\infty} x(t)^* e^{j2\pi ft} dt. \quad (3.36)$$

Replacing f with $-f$, we then get

$$X^*(-f) = \int_{-\infty}^{\infty} x^*(t) e^{-j2\pi ft} dt, \quad (3.37)$$

which completes the proof.

The implication of this property is that, for a real-valued signal, $x(t)$, its Fourier transform is conjugate symmetric, that is, $X(f) = X^*(-f)$. One way to increase the input bandwidth of a digital receiver is to convert real-valued signals to complex-valued signals, thus breaking the symmetry of its Fourier transform.

3.4.5 Scaling

If in a function $x(t)$, the time variable t is replaced by t/a , the corresponding Fourier transform can be found by replacing t/a with t' and the result is

$$F\left[x\left(\frac{t}{a}\right)\right] = \int_{-\infty}^{\infty} x\left(\frac{t}{a}\right) e^{-j2\pi ft} dt = a \int_{-\infty}^{\infty} x(t') e^{-j2\pi af t'} dt' = aX(af). \quad (3.38)$$

This expression means that when time is divided by a factor a , the amplitude and the frequency of the Fourier are expanded by the factor a .

Similarly, if the frequency is divided by a factor a , the inverse Fourier transform can be written as

$$F^{-1}\left[X\left(\frac{f}{a}\right)\right] = ax(at). \quad (3.39)$$

The proof of this equation is similar to the proof of (3.38).

3.4.6 Time Shift

When the time of the input signal $x(t)$ is shifted by t_0 as $x(t - t_0)$, the Fourier transform can be obtained by replacing $t - t_0$ with t' , the result is

$$F[x(t - t_0)] = \int_{-\infty}^{\infty} x(t - t_0) e^{-j2\pi ft} dt = \int_{-\infty}^{\infty} x(t') e^{-j2\pi f(t' + t_0)} dt' = e^{-j2\pi ft_0} X(f). \quad (3.40)$$

This means that a shift in the time domain causes a phase shift in the frequency domain and the amplitude of the output does not change.

3.4.7 Frequency Shift

When there is a frequency shift f_0 in the Fourier output $X(f)$ as $X(f - f_0)$, the effect on the input signal can be found by substituting $f' = f - f_0$ as

$$\mathcal{F}^{-1}[X(f - f_0)] = \int_{-\infty}^{\infty} X(f - f_0) e^{j2\pi f t} df = \int_{-\infty}^{\infty} X(f') e^{j2\pi(f' + f_0)t} df' = e^{j2\pi f_0 t} x(t). \quad (3.41)$$

The corresponding effect is a frequency shift in the time domain as expected.

3.4.8 Derivative

The derivative of the input $x(t)$ with respect to t can be represented by $x'(t)$. Its Fourier output can be found through integration by parts as

$$\begin{aligned} \mathcal{F}[x'(t)] &= \mathcal{F}\left[\frac{dx(t)}{dt}\right] = \int_{-\infty}^{\infty} \frac{dx(t)}{dt} e^{-j2\pi f t} dt \\ &= x(t) e^{-j2\pi f t} \Big|_{-\infty}^{\infty} - (-j2\pi f) \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t} dt = j2\pi f X(f). \end{aligned} \quad (3.42)$$

In the above expression, it is assumed that $x(t)$ is zero at $t = \pm\infty$. This assumption makes the first term equal to zero.

3.4.9 Integral

If the time domain function $\theta(t)$ is

$$\theta(t) = \int_{-\infty}^t x(\tau) d\tau \quad (3.43)$$

and $\theta(\tau)|_{\tau=-\infty} = 0$, the Fourier transform can be found as follows. This equation can be written as

$$\frac{d\theta(t)}{dt} = x(t). \quad (3.44)$$

From the relation in (3.42), the Fourier transform of $d\theta(t)/dt$ is

$$\mathcal{F}\left[\frac{d\theta(t)}{dt}\right] = j2\pi f \Theta(f). \quad (3.45)$$

This equation can be written as

$$\Theta(f) = \mathcal{F}[\theta(t)] = \frac{\mathcal{F}\left[\frac{d\theta(t)}{dt}\right]}{j2\pi f}. \quad (3.46)$$

Substituting (3.43) and (3.44) into (3.46), we get

$$\mathcal{F}\left[\int_{-\infty}^t x(\tau)d\tau\right] = \frac{\mathcal{F}[x(t)]}{j2\pi f}. \quad (3.47)$$

3.4.10 Convolution

If an input signal is applied to a linear and time invariant system, the system output is the convolution of the input signal and the impulse response of the system. Convolution of a function $h(t)$ with a function $x(t)$ is defined as

$$y(t) = h(t) \otimes x(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau. \quad (3.48)$$

The Fourier transform of $y(t)$ can be written as

$$\mathcal{F}(y(t)) = \int_{-\infty}^{\infty} y(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} x(\tau)h(t - \tau)d\tau \right) e^{-j2\pi ft}dt. \quad (3.49)$$

Changing the order of integration, this equation can be written as

$$Y(f) = \int_{-\infty}^{\infty} x(\tau) \left(\int_{-\infty}^{\infty} h(t - \tau)e^{-j2\pi ft}dt \right) d\tau. \quad (3.50)$$

Changing the variable by letting $t - \tau = u$, then

$$\begin{aligned} Y(f) &= \int_{-\infty}^{\infty} x(\tau) \left(\int_{-\infty}^{\infty} h(u)e^{-j2\pi fu}du \right) e^{-j2\pi f\tau}d\tau = \int_{-\infty}^{\infty} x(\tau)H(f)e^{-j2\pi f\tau}d\tau \\ &= H(f) \int_{-\infty}^{\infty} x(\tau)e^{-j2\pi f\tau}d\tau = H(f)X(f). \end{aligned} \quad (3.51)$$

The above derived relation also implies that the convolution in the time domain is equivalent to multiplication in the frequency domain. This relation is often expressed as

$$h(t) \otimes x(t) \Leftrightarrow X(f)H(f) \quad (3.52)$$

and is referred to as the convolution theorem. One can use the convolution in the frequency domain in the same manner as in the time domain. This is shown as

$$X(f) \otimes H(f) = \int_{-\infty}^{\infty} X(\lambda) H(f - \lambda) d\lambda, \quad (3.53)$$

where λ is a dummy variable. The inverse Fourier transform of this result is

$$\begin{aligned} \mathcal{F}^{-1}[X(f) \otimes H(f)] &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X(\lambda) H(f - \lambda) d\lambda e^{j2\pi ft} df \\ &= \int_{-\infty}^{\infty} X(\lambda) \left(\int_{-\infty}^{\infty} H(u) e^{j2\pi ut} du \right) e^{j2\pi \lambda t} d\lambda = h(t)x(t). \end{aligned} \quad (3.54)$$

The proof is identical to (3.51). The above relation can be expressed as

$$x(t)h(t) \Leftrightarrow X(f) \otimes H(f). \quad (3.55)$$

Equations (3.52) and (3.55) are referred to as the duality of convolution and Fourier transform.

3.4.11 Parseval's Theorem [5–8]

Parseval's theorem states that for a given signal, the total energy in the time domain equals the total energy in the frequency domain. This relation can be written as

$$\int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty} |X(f)|^2 df. \quad (3.56)$$

This relation can be proven as follows. From (3.54) or (3.55), we can get the following:

$$\int_{-\infty}^{\infty} h(t)x(t)e^{-j2\pi \lambda t} dt = \int_{-\infty}^{\infty} H(\lambda - f)X(f)df. \quad (3.57)$$

In this equation, both sides are functions of λ , a dummy variable. If $\lambda = 0$, this equation becomes

$$\int_{-\infty}^{\infty} h(t)x(t)dt = \int_{-\infty}^{\infty} H(-f)X(f)df. \quad (3.58)$$

Let us assume that $h(t) = x^*(t)$, where $*$ represents the complex conjugate. Then, based on the conjugation symmetry property, we get $H(-f)$ is equivalent to $X^*(f)$. Therefore

$$\begin{aligned} \int_{-\infty}^{\infty} h(t)x(t)dt &= \int_{-\infty}^{\infty} x^*(t)x(t)dt = \int_{-\infty}^{\infty} |x(t)|^2 dt \\ &= \int_{-\infty}^{\infty} H(-f)X(f)df = \int_{-\infty}^{\infty} X^*(f)X(f)df = \int_{-\infty}^{\infty} |X(f)|^2 df. \end{aligned} \quad (3.59)$$

This completes the proof of Parseval's theorem.

A summary of the continuous Fourier transform can be found in Table 3.1. Fourier transforms of some commonly used functions can be found in Table 3.2.

Table 3.1 Continuous Fourier transform properties.

Properties	Signals	Continuous Fourier transform
Linearity	$x(t)$	$X(f)$
	$y(t)$	$Y(f)$
	$\alpha x(t) + \beta y(t)$	$\alpha X(f) + \beta Y(f)$
Odd and even function	$x(t)$ is an even function	$X(f) = \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)dt$
	$x(t)$ is an odd function	$X(f) = -j \int_{-\infty}^{\infty} x(t)\sin(2\pi ft)dt$
Duality	$X(t)$	$x(-f)$
Conjugation and conjugation symmetry	$x^*(t)$	$X^*(-f)$
Scaling	$x\left(\frac{t}{a}\right)$	$aX(af)$
	$ax(at)$	$X\left(\frac{f}{a}\right)$
Time shift	$x(t - t_0)$	$e^{-j2\pi ft_0} X(f)$
Frequency shift	$e^{j2\pi f_0 t} x(t)$	$X(f - f_0)$
Derivative	$x'(t)$	$j2\pi f X(f)$
Integral	$\int_{-\infty}^t x(\tau)d\tau$	$\frac{X(f)}{j2\pi f}$
Convolution	$x(t) \otimes y(t)$	$X(f)Y(f)$
	$x(t)y(t)$	$X(f) \otimes Y(f)$
Parseval's theorem	$\int_{-\infty}^{\infty} x(t) ^2 dt = \int_{-\infty}^{\infty} X(f) ^2 df$	

Table 3.2 Continuous-time Fourier transform of commonly used functions.

Signals	Continuous Fourier transform
$rect(t/T) = A$ for $-T/2 < t < T/2$ = 0, otherwise	$A \frac{\sin(\pi f T)}{\pi f} = AT \text{sinc}(\pi f T)$
$\delta(t)$	1
$\cos(2\pi f_0 t)$	$\frac{1}{2} [\delta(f - f_0) + \delta(f + f_0)]$
$\sin(2\pi f_0 t)$	$-\frac{j}{2} [\delta(f - f_0) - \delta(f + f_0)]$
$\sum_{n=-\infty}^{\infty} \delta(t - nT)$	$\frac{1}{T} \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T}\right)$
$e^{j2\pi f_0 t}$	$\delta(f - f_0)$
$x(t) = \cos(2\pi f_0 t)$ for $-T/2 < t < T/2$ = 0, otherwise	$X(f) = \frac{A}{2} \left\{ \frac{\sin[\pi(f - f_0)T]}{\pi(f - f_0)} + \frac{\sin[\pi(f + f_0)T]}{\pi(f + f_0)} \right\}$
$x(t) = A^2 T^2 \left(1 - \frac{ t }{T}\right)$ for $ t < T$ = 0, otherwise	$X(f) = A^2 \frac{\sin^2(\pi f T)}{(\pi f)^2} = [AT \text{sinc}(\pi f T)]^2$
$x(t) = a + (1 - a)\cos\left(\frac{2\pi t}{T}\right)$ for $-T/2 < t < T/2$ = 0, otherwise	$X(f) = \frac{a \sin(\pi f T)}{\pi f} + \frac{1 - a}{2} \left\{ \frac{\sin[\pi(f - f_0)T]}{\pi(f - f_0)} + \frac{\sin[\pi(f + f_0)T]}{\pi(f + f_0)} \right\}$
$\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}}$	$e^{-2\pi^2 \sigma^2 f^2}$
Coherent RF pulse train (Example 3.5)	
$x(t) = \left\{ \left[\sum_{n=-\infty}^{\infty} \delta(t - nT) \right] \otimes s_1(t) \right\} s_2(t) \cos(2\pi f_0 t)$, where $s_1(t) = 1$ for $ t < \frac{\tau}{2}$ $s_2(t) = 1$ for $ t < \frac{NT}{2}$	$X(f) = \frac{\sin(\pi f \tau)}{\pi^2 f} \sum_{n=-\infty}^{\infty} \left\{ \frac{\sin\left[\pi\left(f - f_0 - \frac{n}{T}\right)NT\right]}{f - f_0 - \frac{n}{T}} + \frac{\sin\left[\pi\left(f + f_0 - \frac{n}{T}\right)NT\right]}{f + f_0 - \frac{n}{T}} \right\}$
$\frac{1}{\pi t}$	$-j \text{sgn}(f)$

3.5 Examples [3, 5, 8, 12, 13]

This section presents some examples often encountered in signal processing and wideband receivers. Most of the problems are not solved directly, but through the duality of convolution and Fourier transform.

Example 3.1

Find the Fourier transform of $e^{j2\pi f_0 t}$. This result can be obtained directly from the Fourier transform or from the Fourier transform of sine and cosine. We know that

$$e^{j2\pi f_0 t} = \cos(2\pi f_0 t) + j \sin(2\pi f_0 t). \quad (3.60)$$

The Fourier transform of $\cos(2\pi f_0 t)$ can be determined as follows:

$$\begin{aligned} \int_{-\infty}^{\infty} \cos(2\pi f_0 t) e^{-j2\pi f t} df &= \frac{1}{2} \int_{-\infty}^{\infty} [e^{-j2\pi(f-f_0)t} + e^{-j2\pi(f+f_0)t}] dt \\ &= \frac{1}{2} [\delta(f-f_0) + \delta(f+f_0)]. \end{aligned} \quad (3.61)$$

To obtain the final result, the relation in (3.18) is used. The Fourier transform of the cosine function is two impulse functions at $f = \pm f_0$. Similarly, we can find the Fourier transform of $\sin(2\pi f_0 t)$ as

$$\begin{aligned} \int_{-\infty}^{\infty} \sin(2\pi f_0 t) e^{-j2\pi f t} df &= \frac{1}{2j} \int_{-\infty}^{\infty} [e^{-j2\pi(f-f_0)t} - e^{-j2\pi(f+f_0)t}] dt \\ &= \frac{-j}{2} [\delta(f-f_0) - \delta(f+f_0)]. \end{aligned} \quad (3.62)$$

Using the linearity property of (3.30) and combining (3.61) and (3.62), we get

$$\begin{aligned} \mathcal{F}(e^{j2\pi f_0 t}) &= \mathcal{F}(\cos(2\pi f_0 t) + j \sin(2\pi f_0 t)) \\ &= \frac{1}{2} [\delta(f-f_0) + \delta(f+f_0) + \delta(f-f_0) - \delta(f+f_0)] = \delta(f-f_0), \end{aligned} \quad (3.63)$$

which is a single-sided response. This result is important in discussing I and Q channels of frequency downconverters in Chapter 6. This result is shown in Figure 3.5. It should be noted that Figure 3.5(a) is not in the time domain, but is in the complex plane, and the time is represented by the rotating vector.

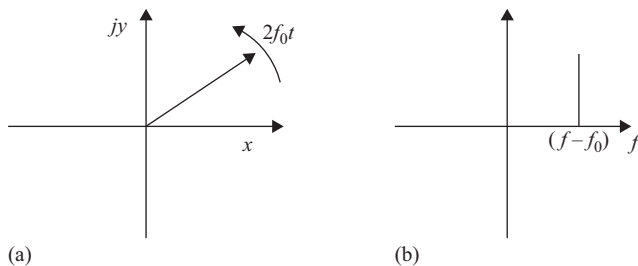


Figure 3.5 Fourier transform of $e^{j2\pi f_0 t}$: (a) complex plane and (b) frequency domain.

Example 3.2

Find the Fourier transform of a rectangular windowed cosine wave. This waveform is often used to represent the output of pulsed radar. If the window is from $-T/2$ to $T/2$ with amplitude A , the windowed cosine wave can be generated by multiplying the window function and a cosine wave as

$$\begin{aligned} x_1(t) &= \cos(2\pi f_0 t) \\ w(t) &= \begin{cases} A & \text{for } -\frac{T}{2} < t < \frac{T}{2}, \\ 0 & \text{elsewhere} \end{cases} \\ x(t) &= w(t)x_1(t), \end{aligned} \quad (3.64)$$

where f_0 is the frequency of the input pulse. Since multiplication in the time domain is equivalent to convolution in the frequency domain, the Fourier transforms of the rectangular window and cosine wave are needed. The Fourier transform of the window, which is a rectangular function, is $W(f) = A \sin(\pi f T) / (\pi f)$. The Fourier transform of a cosine wave, from Example 3.1 is $X_1(f) = [\delta(f - f_0) + \delta(f + f_0)]/2$. The Fourier transform of $x(t)$ is the convolution of $W(f)$ and $X_1(f)$:

$$\begin{aligned} X(f) &= W(f) \otimes X_1(f) = \int_{-\infty}^{\infty} \frac{A \sin(\pi \lambda T)}{\pi \lambda} \frac{[\delta(f - f_0 - \lambda) + \delta(f + f_0 - \lambda)]}{2} d\lambda \\ &= \frac{A}{2} \left\{ \frac{\sin[\pi(f - f_0)T]}{\pi(f - f_0)} + \frac{\sin[\pi(f + f_0)T]}{\pi(f + f_0)} \right\}. \end{aligned} \quad (3.65)$$

The results are shown in Figure 3.6. In this figure $A = 1$, $T = 1$, and $f_0 = 10$. Since the Fourier transform of a cosine function has two δ functions, the convolution with them will generate two outputs. One shifts by f_0 and the other one shifts by $-f_0$.

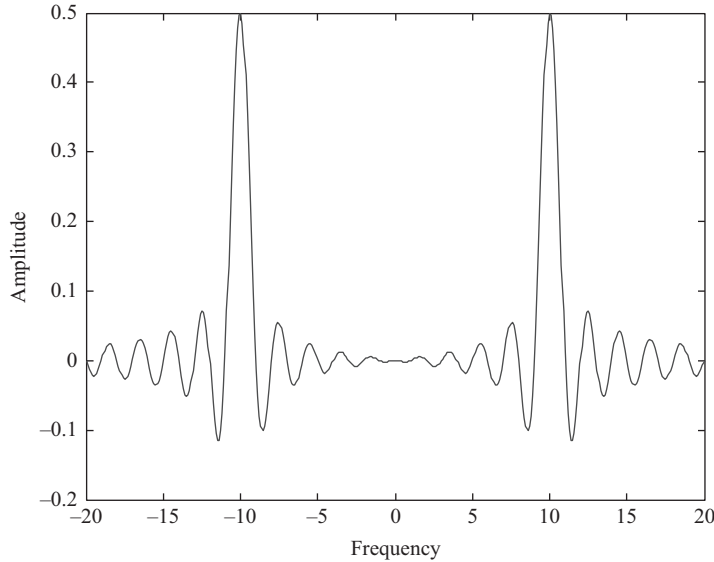


Figure 3.6 Fourier transform of a rectangular windowed cosine wave.

Example 3.3

Find the Fourier transform of

$$x(t) = \begin{cases} a + (1-a)\cos\left(\frac{2\pi t}{T}\right), & \text{for } -\frac{T}{2} < t < \frac{T}{2}, \\ 0, & \text{elsewhere} \end{cases} \quad (3.66)$$

where $a < 1$. The $x(t)$ is often referred to as a generalized cosine window function. Different window functions are often used in signal processing. When $a = 0.54$, $x(t)$ is called the Hamming window. The limited response in the time domain ($-T/2 < t < T/2$) can be achieved by multiplying $a + (1-a)\cos(2\pi t/T)$ by a rectangular function. The duality of convolution theorem can be used to obtain the desired transform. Let $x_1(t) = a + (1-a)\cos(2\pi t/T)$ and $x_2(t)$ represents the rectangular function. Their Fourier transforms are

$$X_1(f) = a\delta(f) + \frac{1-a}{2}[\delta(f-f_0) + \delta(f+f_0)], \quad \text{where } f_0 = \frac{1}{T} \quad (3.67)$$

and

$$X_2(f) = \frac{\sin(\pi f T)}{\pi f} \quad (3.68)$$

The Fourier transform of $x(t)$ equals the convolution of $X_1(f)$ and $X_2(f)$ as

$$\begin{aligned} X(f) &= X_2(f) \otimes X_1(f) = \frac{\sin(\pi f T)}{\pi f} \otimes \left\{ a\delta(f) + \frac{1-a}{2}[\delta(f-f_0) + \delta(f+f_0)] \right\} \\ &= \frac{a \sin(\pi f T)}{\pi f} + \frac{1-a}{2} \left\{ \frac{\sin[\pi(f-f_0)T]}{\pi(f-f_0)} + \frac{\sin[\pi(f+f_0)T]}{\pi(f+f_0)} \right\}. \end{aligned} \quad (3.69)$$

The time and frequency domains are shown in Figure 3.7. In this figure $a = 0.5$ and $T = 5$, thus $-2.5 < t < 2.5$ and f is from $-10/T < f < 10/T$.

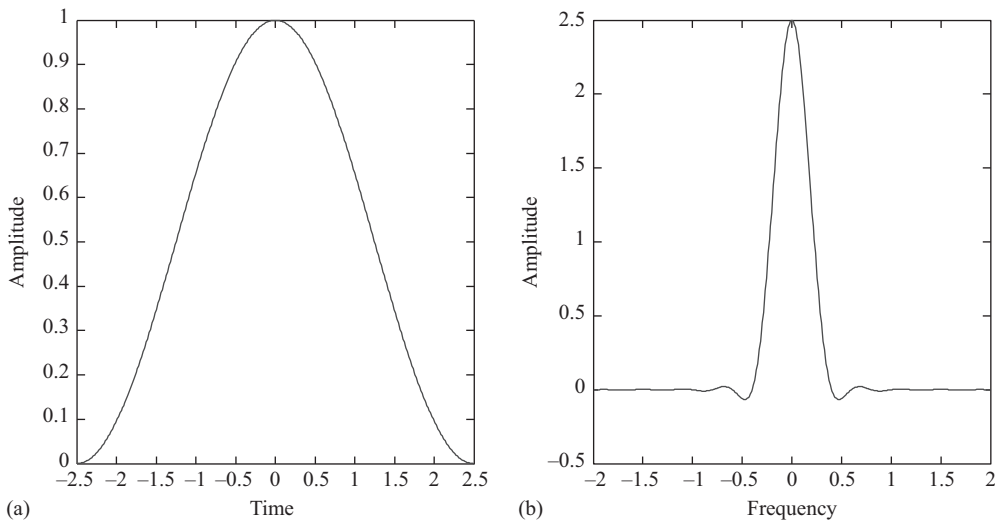


Figure 3.7 Fourier transform of a generalized cosine window with $a = 0.5$: (a) time domain and (b) frequency domain.

Example 3.4

Find the inverse Fourier transform for a quadrature phase shift. Before starting this problem, let us define a sign function as

$$\text{sgn}(f) = \begin{cases} 1, & f > 0 \\ -1, & f < 0 \end{cases}. \quad (3.70)$$

When $f > 0$, the function is $+1$ and when $f < 0$, the function is -1 . The quadrature phase shift is defined in terms of the sign function as

$$X(f) = -j \text{sgn}(f). \quad (3.71)$$

The phase relation is shown in Figure 3.8(a) and is referred to as a step function. The approach to find the inverse transform of this function is to use the derivative of $X(f)$. The derivative can be obtained in a similar way as shown in (3.42), and the result is

$$\mathcal{F}^{-1} \left[\frac{dX(f)}{df} \right] = -j2\pi t x(t). \quad (3.72)$$

The next step is to obtain the derivative of $X(f)$. The derivative of a step function is a δ function at the transition position multiplied by a constant. The constant depends on the amplitude change of the step function. In this special case, the amplitude change is $-2j$ (from $+j$ to $-j$). Thus the derivative is $-j2\delta(f)$. Substituting this relation in the above equation, the result is

$$x(t) = \frac{1}{-j2\pi t} \mathcal{F}^{-1} \left[\frac{dX(f)}{df} \right] = \frac{j}{2\pi t} \int_{-\infty}^{\infty} [-j2\delta(f)] e^{j2\pi ft} df = \frac{j}{2\pi t} (-j2) = \frac{1}{\pi t}. \quad (3.73)$$

This equation has a singularity at $t = 0$. This result will be used in the Hilbert transform in Chapter 6. The time and frequency domain plots are shown in Figure 3.8.

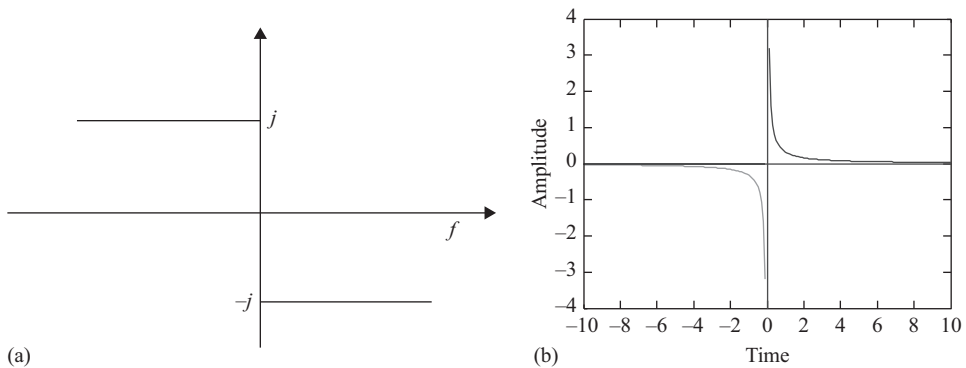


Figure 3.8 A quadrature phase shift: (a) frequency domain and (b) time domain.

Example 3.5

Find the Fourier transform of a radio frequency (RF) pulse train. The approach provided here is a step-by-step combination of desired signals to provide the pulse train. In each step the duality of convolution theorem is used. There are four steps to generate the pulse train in the time domain. In each step the time domain operations and the corresponding frequency domain operations are represented. The last step produces the desired spectrum distribution. To keep this discussion simple, let us assume the pulse train has unit amplitude. In order to keep this approach clear, the selected functions are represented by $c(t)$, $s_i(t)$, and $r(t)$ and the resulting functions are represented by $x_i(t)$, where i is an integer.

1. Choose a comb function $c(t)$ with period T as

$$c(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT). \quad (3.74)$$

Its Fourier transform can be determined as

$$C(f) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T}\right), \quad (3.75)$$

where T represents the pulse repetition interval (PRI) of the pulse train and n is an integer.

2. Choose a rectangular function $s_1(t)$ with width τ and unit amplitude to represent an individual pulse. The time and frequency domains are

$$s_1(t) = \begin{cases} 1, & \text{for } |t| < \frac{\tau}{2} \\ 0, & \text{elsewhere} \end{cases} \quad (3.76)$$

$$S_1(f) = \frac{\sin(\pi f \tau)}{\pi f}. \quad (3.77)$$

In order to create a pulse train $x_1(t)$, $s_1(t)$ can be used to convolve with the comb function $c(t)$, thus

$$x_1(t) = c(t) \otimes s_1(t) \quad (3.78)$$

$$X_1(f) = C(f)S_1(f) = \frac{\sin(\pi f \tau)}{\pi f} \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T}\right). \quad (3.79)$$

In the above operation, $f = n/T$ is substituted into $S_1(f)$ because the δ functions in $C(f)$ only exist at these values.

3. Choose a second rectangular function $s_2(t)$ with width NT and unit amplitude to represent the total length of the pulse train. Thus

$$s_2(t) = \begin{cases} 1, & \text{for } |t| < \frac{NT}{2} \\ 0, & \text{elsewhere} \end{cases} \quad (3.80)$$

$$S_2(f) = \frac{\sin(\pi f NT)}{\pi f}. \quad (3.81)$$

The function $s_2(t)$ is multiplied by $x_1(t)$ to create the pulse train, which is represented by $x_2(t)$. The frequency domain result is obtained by convolving $X_1(f)$ with $S_2(f)$ as

$$x_2(t) = x_1(t)s_2(t) \quad (3.82)$$

$$X_2(f) = X_1(f) \otimes S_2(f) = \frac{\sin(\pi f \tau)}{\pi^2 f} \sum_{n=-\infty}^{\infty} \frac{\sin[\pi(f - \frac{n}{T})NT]}{f - \frac{n}{T}}. \quad (3.83)$$

It should be noted that convolving with a δ function shifts the function by n/T .

4. The last step is to add the RF in the pulse train. Let us choose $r(t)$ to represent a cosine wave. Then

$$r(t) = \cos(2\pi f_0 t) \quad (3.84)$$

$$R(f) = \frac{1}{2}[\delta(f - f_0) + \delta(f + f_0)], \quad (3.85)$$

where f_0 is the RF of the pulse. The function $x_2(t)$ is multiplied by $r(t)$ to produce the RF pulse train. The final results are

$$x(t) = x_2(t)r(t) \quad (3.86)$$

$$X(f) = X_2(f) \otimes R(f)$$

$$= \underbrace{\frac{\sin(\pi f \tau)}{\pi^2 f}}_1 \sum_{n=-\infty}^{\infty} \left\{ \underbrace{\frac{\sin[\pi(f - f_0 - \frac{n}{T})NT]}{f - f_0 - \frac{n}{T}}}_2 + \underbrace{\frac{\sin[\pi(f + f_0 - \frac{n}{T})NT]}{f + f_0 - \frac{n}{T}}}_3 \right\} \quad (3.87)$$

The time domain signal generated through the four steps is shown in Figure 3.9(a). It is a series of RF pulses. This signal in radar is called coherent, because there is a certain phase relation between the individual pulses. In other words, a continuous sinusoidal wave that is gated by a pulse train generates a coherent pulse train, because every pulsed RF signal is part of the same sinusoidal wave.

Now let us discuss the result of (3.87). The first term represent the envelope of the spectrum. The first zero of the envelope can be found from

$$f = \frac{1}{\tau}. \quad (3.88)$$

The width of the mainlobe is $2/\tau$, and this result is shown in Figure 3.9(b). The second and third terms represent upper and lower band spectra. They are located at $\pm f_0$ and are symmetrical with respect to zero frequency. The upper group of spectrum lines occurs only at

$$f = f_0 + \frac{n}{T}. \quad (3.89)$$

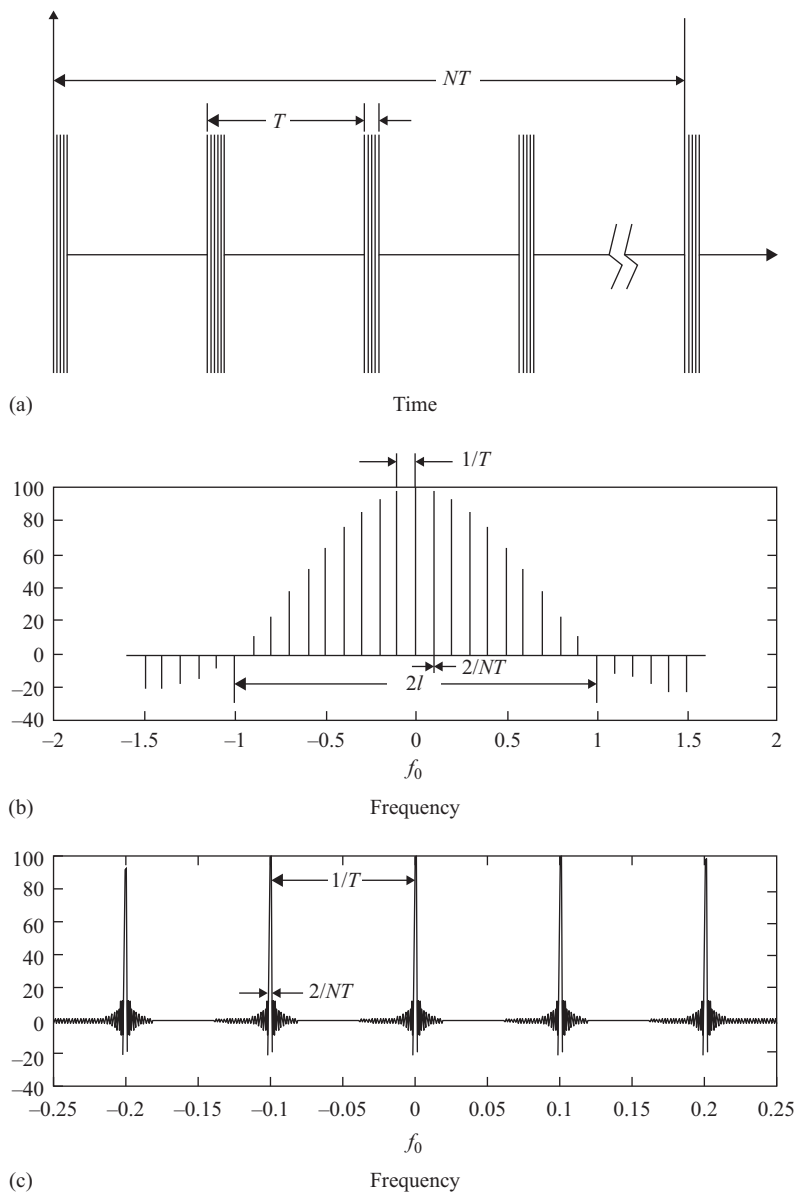


Figure 3.9 Fourier transform of an RF pulse train: (a) RF pulse train, (b) spectrum lines, and (c) details of individual lines.

The spectrum lines are separated by $1/T$. The fine structure of the individual line can be found from the second term in (3.87) as

$$X(f) = K \frac{\sin[\pi(f - f_0 - \frac{n}{T})NT]}{\pi(f - f_0 - \frac{n}{T})NT}, \quad (3.90)$$

where K is a constant. The first zero occurs at $NT = 1$, thus the width of the individual spectrum line is $2/NT$. This result is shown in Figure 3.9(c).

Some radars measure the velocity of a target through the Doppler frequency shift, f_d . In a radar system, this quantity can be written as

$$f_d = \frac{2f_0 V}{C}, \quad (3.91)$$

where f_0 is the operating frequency, V is the velocity of the target, and C is the speed of light [13]. The factor 2 in this equation is caused by the round-trip of the radar signal. This type of radar is referred to as pulse Doppler radar. In order to reduce the ambiguity in frequency measurement, the separation between frequency lines must be wide. This requires high PRF or a low PRI, which means a short T . In order to have fine spectrum line width, NT must be large, or N must be large. In other words, a large number of pulses must be integrated. Thus pulse Doppler radar can generate very high pulse density.

3.6 DFT

The continuous Fourier transform is a useful analytical tool; however, its direct applications to receiver design are limited for two reasons. First, the function in the time domain must be representable in a closed form so that the Fourier integral can be performed. Thus, unless the input function can be written in closed form, it is impossible to evaluate the integral. Second, even if the time function can be written in a closed form, it might also be difficult to find a closed form solution of the integral.

In a digital receiver, the input data are obtained from digitizing the input signal. In an electronic warfare (EW) environment, the input signal represented by a function in the time domain is usually unknown. Even if the input is known, such as simulated data generated from a sine or cosine function to test a signal processing algorithm, the signal is digitized before it is processed. When the input signal is in a digitized form, the DFT is used to implement the Fourier transform. Unlike the Fourier transform, the DFT can be performed on any kind of digitized input data, therefore its use is not limited.

It is important to know that the DFT does not provide the same result as the Fourier transform. It only provides an approximate solution. The DFT was developed to process sampled signal with a finite duration. Sometimes these results can be very close to the desired result, however, it rarely happens for short data length. At other times, the results generated from the DFT may be misleading. In order to better understand the DFT, the basic operation of the DFT will be discussed first, some of the important properties will be presented, and finally, the fast Fourier transform (FFT) will be presented.

3.7 Signal Digitization

A sine wave in the time domain t can be represented by

$$x(t) = A \sin(2\pi f_0 t), \quad (3.92)$$

where A is the amplitude and f_0 is the frequency. A set of digitized data can be generated from this signal by sampling some discrete points on the sine wave. The device that performs

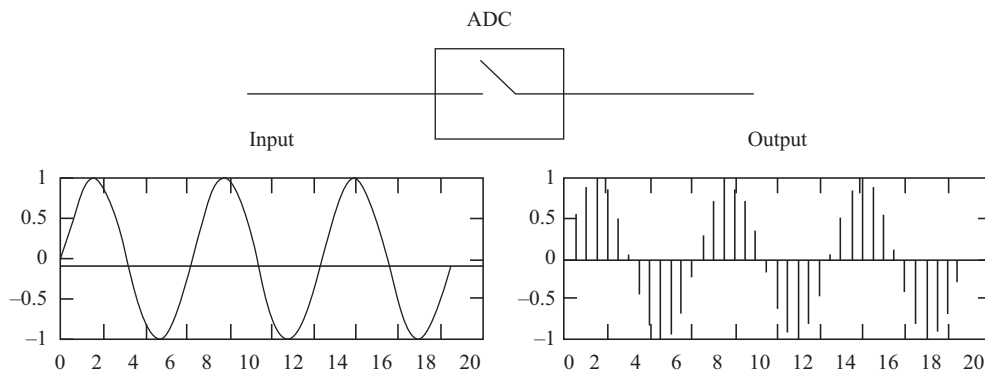


Figure 3.10 A digitized waveform.

this operation is called an analog-to-digital converter (ADC), which is discussed in Chapter 5. The digitizing operation can be represented by a switch (see Figure 3.10). It samples the input signal at a uniform rate. Of course, the input signal can also be sampled at a nonuniform rate. It is one method used in analog-to-information (A-to-I) but is not included in this book. In this book, only the uniform digitized signal will be discussed, because it is necessary for DFT operation.

In Figure 3.10, the input is continuous but the ADC output is a pulse train with envelope following the input. It is obvious that not all the information in the signal is transferred to the output. The information between the sampling points is lost. Thus the output is no longer exactly equal to the input signal. This loss of information is discussed in Section 3.9.

Mathematically, the digitizing operation can be represented by multiplying the input signal with a comb function in the time domain, which can be written as

$$x(nt_s) = x(t) \sum_{n=-\infty}^{\infty} \delta(t - nt_s), \quad (3.93)$$

where n is an integer, t_s is the sampling interval, and $x(nt_s)$ represents the digitized signal. The digitized signal has an output only at integer multiples of t_s , as shown in Figure 3.10.

3.8 Derivation of the DFT

The Fourier transform analyzes a continuous signal in the frequency domain and the Fourier transform generates a continuous spectrum in the frequency domain. The DFT can be considered as samples of the Fourier transform of a signal with finite length. Therefore DFT is not identical to the Fourier transform but is an approximation. We will explain DFT from two different viewpoints in this section.

3.8.1 Graphical Description

In this section the DFT is described through a graphical illustration and two aspects are discussed: the mathematical formula and the time and frequency domain responses. A step-by-step approach will be taken starting from signal digitization. The discussion is

somewhat similar to Example 3.5. In Example 3.5, the data in the frequency domain are still continuous. In a DFT, the data in the frequency domain are also discrete, and this approach can avoid an analytic integral. Let us use a simple example to demonstrate the operation.

1. We used $\text{sinc}^2 t$ as the input signal, $x(t)$, to explain the DFT. The signal $x(t)$ is a continuous signal and its Fourier transform $X(f)$ is an isosceles triangle. The results are shown in Figure 3.11(a).
2. The sampling process can be represented by multiplying $x(t)$ by a comb function $b(t)$ with interval t_s . The Fourier transform of this function, $B(f)$, is also a comb function with an amplitude of $1/t_s$ and a period of $1/t_s$. The results are shown in Figure 3.11(b).
3. The input signal $x(t)$ multiplied by $b(t)$ can be written as $y_1(t) = x(t)b(t)$. The corresponding frequency domain operation is $Y_1(f) = X(f) \otimes B(f)$. The results are shown in Figure 3.11(c). Notice that the sampling process generates discrete data whose Fourier transform result is a periodic and whose period is $1/t_s$.
4. To analyze the signal in real time, the input signal must be limited in length. This operation can be achieved by using a rectangular function $w(t)$. This function will cover N points whose indexes are from 0 to $N - 1$. The Fourier transform of $w(t)$ is a sinc function $W(f)$, as shown in Figure 3.11(d).
5. The signal $y_1(t)$ is multiplied by the rectangular function $w(t)$ to limit data length to N points. This operation can be written as $y_2(t) = y_1(t)w(t)$. The corresponding frequency domain operation is $Y_2(f) = Y_1(f) \otimes W(f)$. The results are shown in Figure 3.11(e). It should be noted that $Y_2(f)$ is periodic but is still continuous. To obtain $Y_2(f)$, the Fourier transform of $y_2(t)$ is required. As a result, the frequency information must still be obtained through an analytic operation. If one can obtain the frequency information as discrete data, the analytic operation can be avoided. The next two steps are required to eliminate the analytic operation and at the same time generate discrete components in the frequency domain. However, these operations also change the shape of the input signal.
6. In order to obtain discrete frequency components, a second comb function in the frequency domain $C(f)$ is needed. This comb function cannot be created with an arbitrary period. It has a period of $1/(Nt_s)$, which is determined by the data duration Nt_s . The inverse Fourier transform is $c(t)$ in the time domain with a period of Nt_s , which matches the length of the rectangular function $w(t)$. The results are shown in Figure 3.11(f).
7. The final step is to obtain the frequency information in discrete form. This operation is achieved by multiplying $Y_2(f)$ by $C(f)$, which can be written as $X(k) = Y_2(k)C(k)$. In this operation, the frequency can be written as $f = k/(Nt_s)$, where k is an integer and $1/(Nt_s)$ is the frequency resolution. In the time domain the corresponding operation is $x(nt_s) = y_2(t) \otimes c(t)$, where $x(nt_s)$ represents discrete time domain data. This operation generates an infinite replica of the digitized input signal. The results are shown in Figure 3.11(g). This is a very important step, because after this step the input signal in the time domain is no longer the digitized signal shown in Figure 3.11(e). The input signal becomes periodic with period Nt_s . If one performs all the above operations, the approximate frequency components of the input signal can be obtained.

The DFT procedure is summarized as follows. We first sample a continuous sign, thus creating a periodic frequency spectrum. Then we limit the signal data length by multiplying the sampled data with a rectangular window function, thus changing the frequency components. The last step is to sample of the frequency spectrum. This action creates a periodic sequence.

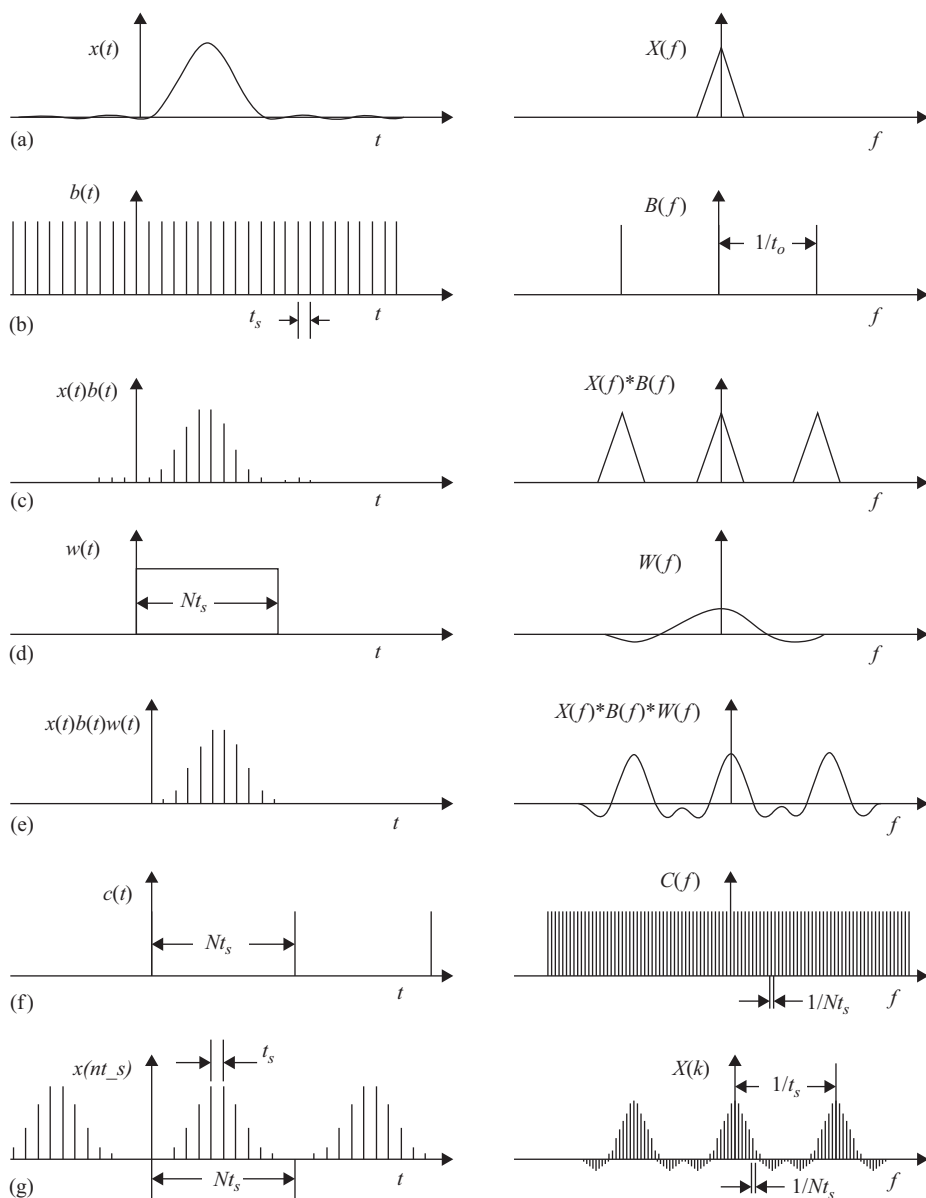


Figure 3.11 Graphic demonstration of a DFT: (a) input $x(t)$ and $X(f)$, (b) comb function $b(t)$ and $B(f)$, (c) $x(t)b(t)$ and $X(f) \otimes B(f)$, (d) rectangular function $w(t)$ and $W(f)$, (e) $x(t)b(t)w(t)$ and $X(f) \otimes B(f) \otimes W(f)$, (f) comb function $c(t)$ and $C(f)$, and (g) output $x(nt_s)$ and $X(k)$.

As shown in Figure 3.11, we start with a continuous aperiodic signal with a continuous aperiodic spectrum and end with a periodic digitized signal with a periodic discrete spectrum. Thus the DFT is a practical approach to approximate a Fourier transform; however, some information will be lost during this process. It is important to understand the difference between a Fourier transform and a DFT and what information is lost in this process. These will be considered in Section 3.9.

3.8.2 Analytic Approach

The analytic approach follows the graphical explanation discussed in the previous section. Assume the input signal is $x(t)$ and the comb function $b(t)$ is

$$b(t) = \sum_{n=-\infty}^{\infty} \delta(t - nt_s), \quad (3.94)$$

where t_s is the sampling interval. The function $y_1(t)$ is

$$y_1(t) = x(t)b(t) = x(t) \sum_{n=-\infty}^{\infty} \delta(t - nt_s) = \sum_{n=-\infty}^{\infty} x(t)\delta(t - nt_s). \quad (3.95)$$

The rectangular function $w(t)$ covers the δ function from $t = 0$ to $(N - 1)t_s$, which limits signal duration, thus

$$y_2(t) = y_1(t)w(t) = \sum_{n=0}^{N-1} x(t)\delta(t - nt_s). \quad (3.96)$$

The next step is to sample $Y_2(f)$ in the frequency domain, which is done by multiplying $Y_2(f)$ by a comb function $C(f)$, whose period is $1/(Nt_s)$. This operation is equivalent to convolving $y_2(t)$ with $c(t)$ and the result is

$$\begin{aligned} y(t) = y_2(t) \otimes c(t) &= \sum_{n=0}^{N-1} x(nt_s)\delta(t - nt_s) \otimes \sum_{r=-\infty}^{\infty} \delta(t - rNt_s) \\ &= \sum_{r=-\infty}^{\infty} \sum_{n=0}^{N-1} x(nt_s)\delta(t - nt_s - rNt_s). \end{aligned} \quad (3.97)$$

The last step is to find the Fourier transform of $y(t)$. This step can be written as

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} \sum_{r=-\infty}^{\infty} \sum_{n=0}^{N-1} x(nt_s)\delta(t - nt_s - rNt_s)e^{-j2\pi ft} dt \\ &= \sum_{r=-\infty}^{\infty} \sum_{n=0}^{N-1} x(nt_s) \int_{-\infty}^{\infty} \delta(t - nt_s - rNt_s)e^{-j2\pi \frac{k}{N}t} dt \\ &= \sum_{r=-\infty}^{\infty} \sum_{n=0}^{N-1} x(nt_s)e^{\frac{-j2\pi kn}{N} - j2\pi kr} \equiv X(k). \end{aligned} \quad (3.98)$$

In (3.98) the relation $f = k/(Nt_s)$ is used. In the frequency domain, the output only has values at $f = k/(Nt_s)$ because of the sampling property of the δ function. The notation $X(k)$ is used to represent discrete components of the Fourier transform.

From the above result, the DFT has an infinite number of frequency components. However, because of the periodic property of $e^{-j2\pi kr}$, where k and r are integers, the output spectrum repeats the data from $k = 0$ to $N - 1$. Usually only one such period is used to represent the output. If one considers $r = 0$ in the above equation, the result is

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}. \quad (3.99)$$

This is the well-known relation of the Fourier transform in discrete form. In this equation, $t_s = 1$ is used. It can be easily proved that $X(k) = X(k + N)$, therefore the $X(k)$ is a periodic sequence in the frequency domain, as shown in Figure 3.11(g).

The inverse DFT can be written as

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N} \quad (3.100)$$

and proved as follows. By substituting the result in (3.100) into (3.99), the result of (3.99) can be obtained. It can be easily proved that $x(n) = x(n + N)$, therefore the $x(n)$ is a periodic sequence in the time domain, as shown in Figure 3.11(g). Equation (3.100) can be written as

$$\begin{aligned} X(k) &= \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} = \sum_{n=0}^{N-1} \left[\frac{1}{N} \sum_{k'=0}^{N-1} X(k') e^{j2\pi k'n/N} \right] e^{-j2\pi kn/N} \\ &= \sum_{k'=0}^{N-1} X(k') \left[\frac{1}{N} \sum_{n=0}^{N-1} e^{j2\pi (k'-k)n/N} \right] = X(k). \end{aligned} \quad (3.101)$$

In (3.101), the relation

$$\sum_{n=0}^{N-1} e^{j2\pi (k'-k)n/N} = \begin{cases} N, & k' = k \\ 0, & k' \neq k \end{cases} \quad (3.102)$$

is used. Equations (3.99) and (3.100) are the well-known relations of the DFT and the inverse discrete Fourier transform (IDFT).

3.9 More Discussions on the DFT

The Fourier transform of a continuous signal needs not be limited in length in the time domain. The output in the frequency domain is not band limited either. However, the input signal in a DFT is limited in length in the time domain and becomes periodic due to the sampling effect in the frequency domain. The output in the frequency domain is also periodic due to the sampling effect in the time domain. Thus the DFT is only an approximation of the continuous Fourier transform. One must carefully evaluate the output of the DFT, because the result may be erroneous if the input signal does not fulfill certain requirements.

In this section, some issues of the DFT are covered to help the reader avoid misreading DFT results.

3.9.1 Limited Frequency Bandwidth

Figures 3.11(b) and 3.11(g) show that the output in the frequency domain has a period of $f_s = 1/t_s$, where f_s is the sampling frequency and t_s is the sampling interval. Since the frequency output from a real-valued signal has two sidebands, the signal frequency must be less than $f_s/2$. If the signal frequency is higher than $f_s/2$, the periodic extensions of the output will overlap, as shown in Figure 3.12. Figure 3.12(a) shows the signal $x(t)$ and its Fourier transform. Figure 3.12(b) shows the digitizing comb function $c(t)$ and $C(f)$. Figure 3.12(c) shows the product of $x(t)$ and $c(t)$ and the corresponding frequency domain output $X(f) \otimes C(f)$. The true spectrum is shown by the dotted lines. It is obvious that the output spectrum does not match the input spectrum because of the overlap. This phenomenon is called aliasing. Therefore, in general, the input signal bandwidth is kept less than $f_s/2$. This is referred to as the Nyquist sampling theorem. If the input signal has a bandwidth higher than $f_s/2$, the spectrum will alias in the baseband. Sometimes the DFT can be used to solve the problem of several narrowband signals spreading over a bandwidth greater than $f_s/2$, such as in an EW receiver. In order to use the DFT for this purpose, special procedures are required in the processing. This problem is discussed in Chapter 4.

If the sampling interval t_s is decreased to a very small value, the unambiguous bandwidth will widen. When all the continuous time data in the input signal are used, which is equivalent to making $t_s \rightarrow 0$, the bandwidth of the equivalent sampled response becomes infinity. Therefore one can consider that the limitation on the bandwidth is due to information being lost between the sampled points.

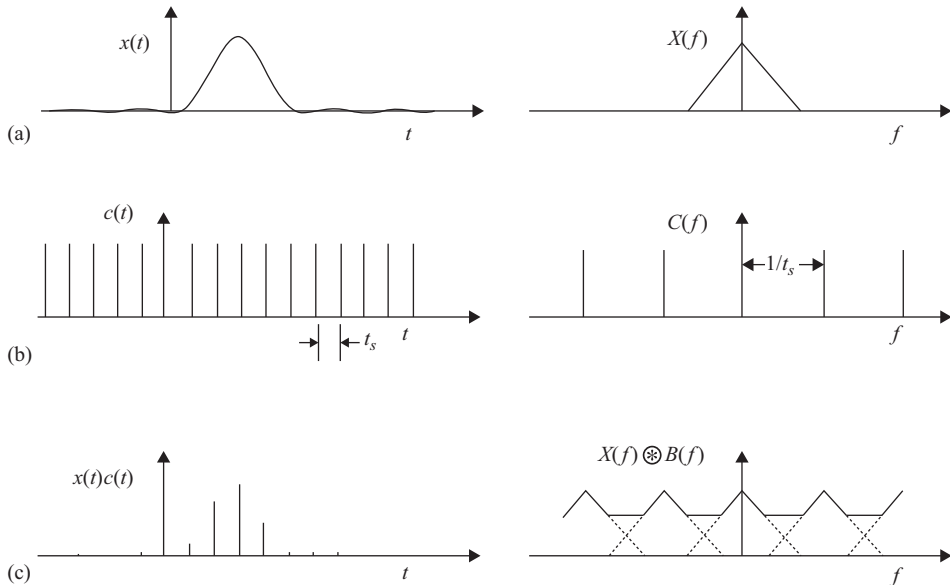


Figure 3.12 Aliasing effect: (a) $x(t)$ and $X(f)$, (b) $c(t)$ and $C(f)$, and (c) $x(t)c(t)$ and $X(f) \otimes C(f)$.

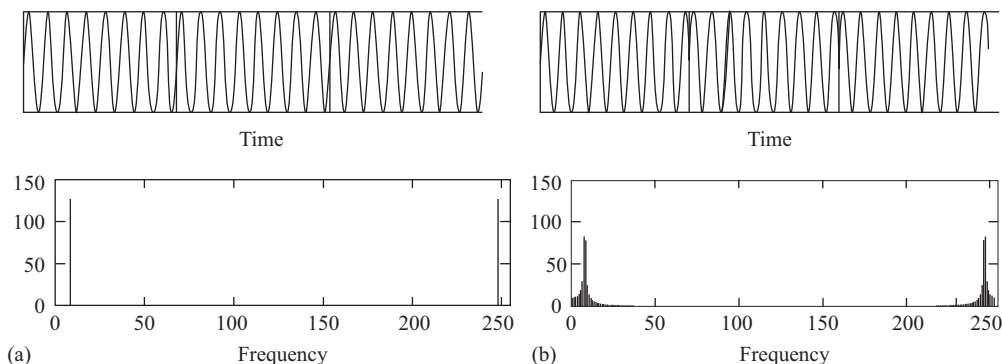


Figure 3.13 Leakage effect of a sine wave: (a) window matches and (b) window does not match the period of the input signal.

3.9.2 Unmatched Time Interval

The input signal in the time domain is windowed by a rectangular function that contains N data points from $n = 0$ to $N - 1$, as discussed in the previous section. If the input is a periodic function such as $\sin(2\pi f_0 t)$, the window may not match the period of the sine function. Since the input signal is always periodic with respect to N , the input signal will not be a continuous sine wave when the window does not match the period of the signal. Figure 3.13 shows the results of such an operation. Figure 3.13(a) shows that the window matches the period of the input signal. Under this condition, the signal appears continuous and its DFT has two peaks. The positions of the peaks represent the frequency of the signal. Figure 3.13(b) shows the case when the time window does not match the period of the input signal. Under this condition, the sine wave is no longer continuous, but breaks into many discontinuous sections. As a result, there are many spectrum lines in the frequency domain. The additional spectrum lines are caused by the sharp discontinuity in the input signal. Another way to explain this phenomenon is that the DFT performs sampling in the frequency domain. If the signal frequency is not on the sampling points of the DFT (i.e., the time window does not match the signal period), signal energy in the frequency domain is leaked from one main component to many components. This phenomenon is often referred to as the leakage effect. To guarantee no leakage effect, the DFT frequency resolution $1/(Nt_s)$ needs to approach zero, which means N (and window length) need to approach infinity. The leakage phenomenon is not easy to avoid. Special window functions discussed in the next section will help reduce the discontinuity.

3.9.3 Overlapping Aliasing Effect on Real-Valued Data

First, let us consider the Fourier transform in continuous form. If the data are real-valued, the frequency domain output will have conjugate symmetric positive and negative components. For example, if the input is $\sin(2\pi f_0 t)$, the Fourier transform result will be at $\pm f_0$. When the input frequency is low, the two frequency components are close to the y -axis. When the input frequency increases, the two spectra will separate far apart.

Let us assume that real data are used as the input. In the DFT, the output spectra are only shown from 0 to $N - 1$. The negative frequency does not show up on the negative side of

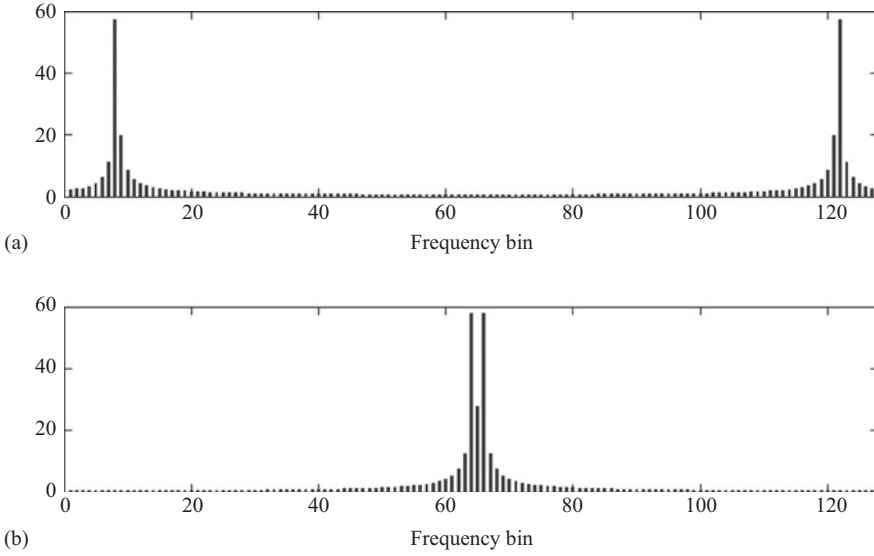


Figure 3.14 Positive and negative spectra: (a) low frequency and (b) high frequency.

the y -axis. Both the spectra are shown in the 0 to $N - 1$ range. They are symmetrical with respect to $N/2$. It should be noted that because of the periodic nature of the output, the output at $X(0)$, the lowest frequency bin, and $X(N - 1)$, the highest frequency bin, should be next to each other. Figure 3.14 shows the power spectra of a sine function. In Figure 3.14(a), the input frequency is low and the two spectra are slightly overlapped at the low end (close to $X(0)$) and the high end (close to $X(N - 1)$). This result still resembles the true frequency response.

When the input frequency increases, in the DFT case, the positive spectrum move away from $X(0)$ toward the right. The negative spectrum should move away from $X(0)$ toward the left, but there is no negative index of X . Since $X(0)$ is next to $X(N - 1)$, moving away from $X(0)$ toward the right is equivalent to moving away from $X(N - 1)$ toward the left. Thus the negative spectrum moves away from $X(N - 1)$ toward $X(0)$, as shown in Figure 3.14(b). When the input frequency increases approaching $f_s/2$, the two spectra are close together at near $X(N/2)$. In general, since each spectrum has sidelobes, these two spectra will interfere with each other. When the input frequency equals half the sampling frequency ($f_s/2$), the two responses will overlap and highly interfere with each other. In general, when the input signal frequency is close to

$$f_i \approx \frac{nf_s}{2}, \quad (3.103)$$

where n is an integer including zero, the positive and negative spectra will interfere with each other. The amount of interference depends on the leakage effect and the bandwidth of the input signal and the sidelobes. Even if the input frequency bandwidth is less than $f_s/2$, that is, the Nyquist sampling rate is satisfied, the positive and negative spectra may still interfere with each other. If we convert the real-valued sine signal to a complex-valued signal, this overlapping effect can be avoided.

3.9.4 Circular and Linear Convolution [14, 15]

The convolution of two continuous signals $x(t)$ and $y(t)$ is defined as

$$x(t) \otimes y(t) = \int_{-\infty}^{\infty} x(\tau)y(t - \tau)d\tau. \quad (3.104)$$

It was shown in Section 3.4 that convolution in the time domain is equivalent to multiplication in the frequency domain, thus (3.52) is rewritten as

$$x(t) \otimes y(t) \Leftrightarrow X(f)Y(f). \quad (3.105)$$

The direct conversion of (3.104) to the discrete domain is

$$x(n) \otimes y(n) = \sum_{m=-\infty}^{\infty} x(m)y(n - m). \quad (3.106)$$

The convolution designed in (3.106) is referred to as the linear convolution of the sequence $x[n]$ and $y[n]$. However, when we multiply their DFT results, $X(k)$ and $Y(k)$, and take the IDFT of the corresponding product, the resulting time domain result is not a linear convolution, but a circular convolution, as designed in (3.107):

$$\sum_{m=0}^{N-1} x(m)y(n - m) \Leftrightarrow X(k)Y(k). \quad (3.107)$$

It should be noted that the time shift in $y(n - m)$ is circular because the IDFT creates a periodic sequence. Equation (3.107) can be proved as follows. By taking the discrete Fourier transform of (3.105) one obtains $Z(k)$ as

$$\begin{aligned} Z(k) &= \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} x(m)y(n - m)e^{\frac{-j2\pi kn}{N}} \\ &= \sum_{m=0}^{N-1} x(m) \left[\sum_{n=0}^{N-1} y(n - m)e^{\frac{-j2\pi k(n-m)}{N}} \right] e^{\frac{-j2\pi km}{N}} \\ &= Y(k) \left(\sum_{m=0}^{N-1} x(m)e^{\frac{-j2\pi km}{N}} \right) = Y(k)X(k). \end{aligned} \quad (3.108)$$

Equation (3.108) is referred to as the circular convolution. It does not provide the expected result of a linear convolution. The reason is because of the periodic nature of the DFT and IDFT.

Before we explain this effect, let us use a simple example to demonstrate this effect. Let $x(n) = \{1, 1, 1, 1\}$; $y(n) = \{0, 1, 2, 3\}$; for $n = 0, 1, 2, 3$, respectively. The linear convolution is $z(n) = x(n) \otimes y(n) = \{0, 1, 3, 6, 6, 5, 3\}$; for $n = 0$ to 6. The result is shown in Figure 3.15. Figures 3.15(a) and 3.15(b) show $x(n)$ and $y(n)$, respectively. Figure 3.15(c) shows $x(n) \otimes y(n)$. Figure 3.15(d) shows that $y(n - m)$ shifts seven ($4 + 4 - 1 = 7$) times

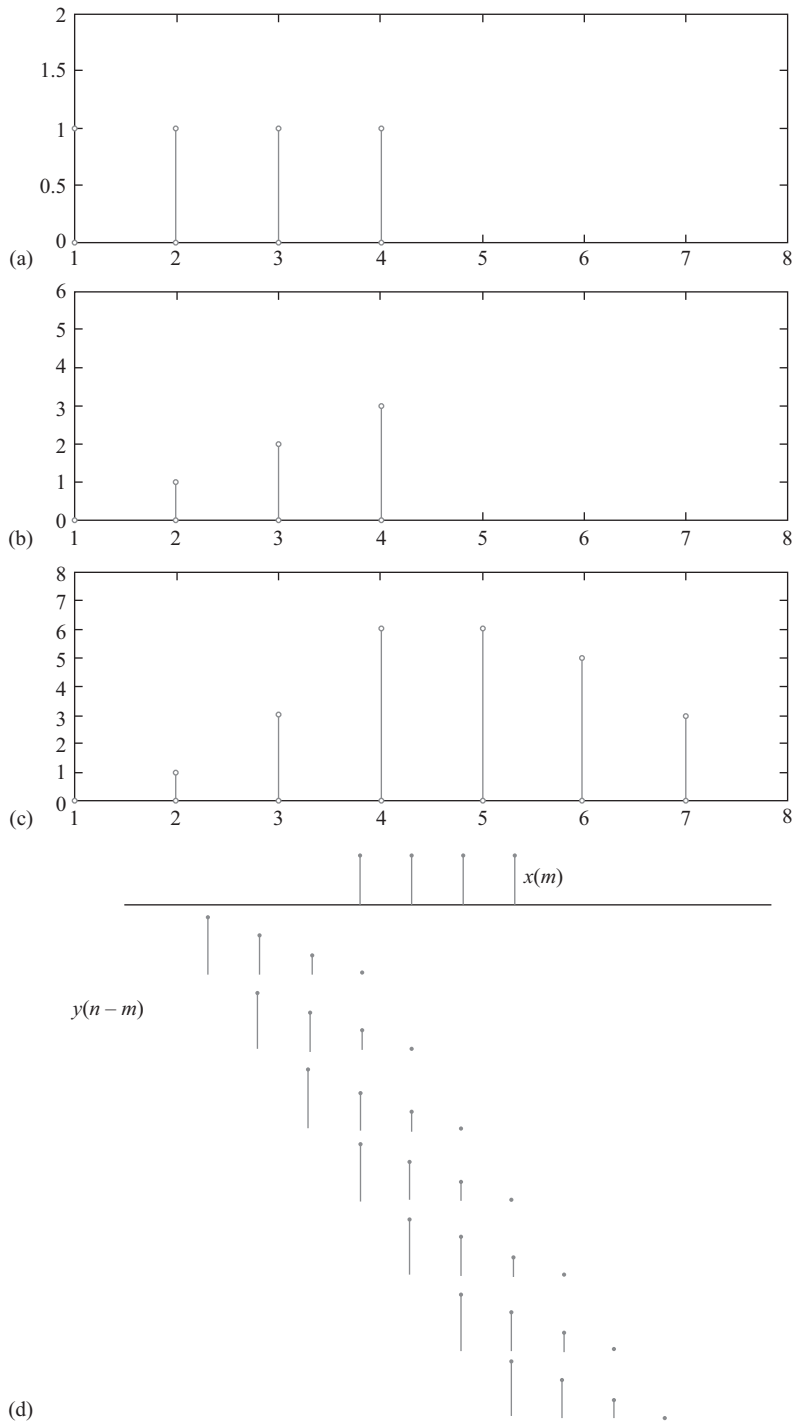


Figure 3.15 Results of linear convolution: (a) $x(n)$, (b) $y(n)$, (c) $z(n) = x(n) \otimes y(n)$, and (d) sliding of $y(n - m)$ against $x(m)$.

to the right. After the seventh shift ($n = 6$), $x(m)$ and $y(n - m)$ no longer overlap. If the dot product of $x(m)$ and $y(n - m)$ is calculated at each shift, the result in Figure 3.15(c) will be obtained.

If one takes the DFT of $x(n)$ and $y(n)$ to obtain $X(k)$ and $Y(k)$, then takes the IDFT of $X(f)Y(f)$, the result is $z(n) = \{6, 6, 6, 6\}$ and it has only four outputs. This is the result of the circular convolution. The circular convolution result that is obtained is shown in Figure 3.16. Due to the periodicity inherent in the DFT and IDFT, both $x(n)$ and $y(n)$ are periodic in nature. When $y(-m)$ slides one unit toward to right, the entire sequence moves one unit. As a result, the pattern repeats itself after four shifts. For this example, the output is the same at each step and the result is always $6 (= 3 + 2 + 1)$, which is the same as that obtained from the DFT operation. When searching for signals in a global positioning system (GPS) using circular correlation, circular convolution is very useful [16].

In many signal processing applications, one would like to avoid circular convolution and instead obtain linear convolution using the DFT. One way to accomplish this goal is padding the original sequences with zeros before conducting the DFT. The question is how many zeros are needed. Let us discuss this problem from a more general approach. It should be noted that to perform convolution, the data length of x and y need not be the same. From Figure 3.15(d), it is easy to see that the total length of the convolution is $N \pm N - 1$. If $x(n)$ has M data points and $y(n)$ has N data points, the linear convolution output has $M + N - 1$ points, that is, one of the inputs must be shifted $M + N - 1$ times during convolution.

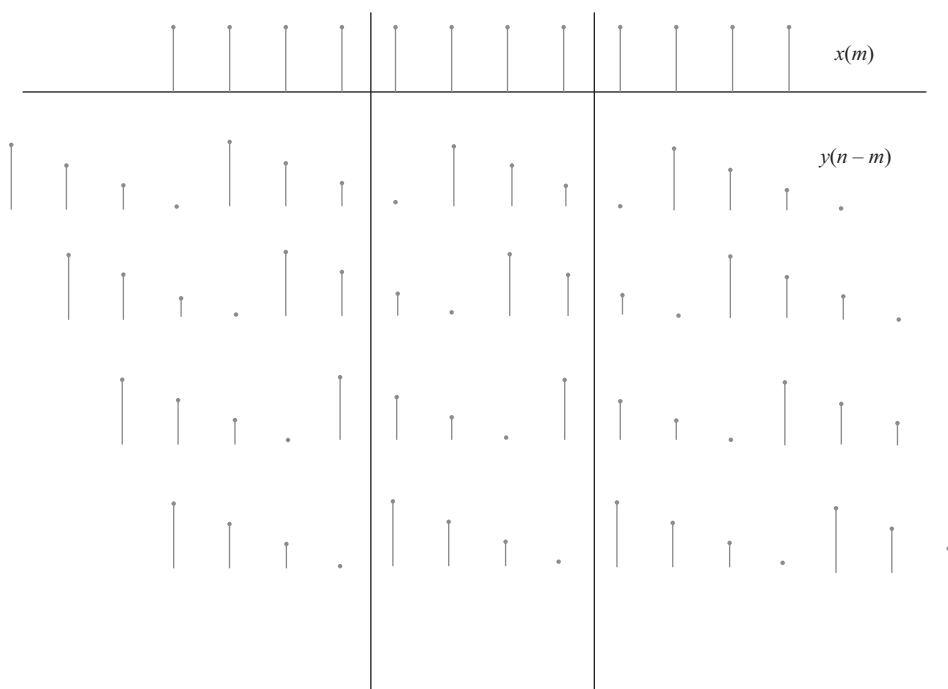


Figure 3.16 Circular convolution.

Afterwards, the two sequences will not overlap and the sum of the dot product will be zero. Therefore both $x(n)$ and $y(n)$ must be padded with a sufficient number of zeros so that the minimum length is $M + N - 1$ points. This creates enough room to perform the linear convolution and avoid circular convolution.

As a demonstration, the example at the beginning of this section will be solved with zero padding. Let us pad $x(n)$ and $y(n)$ each with a minimum of three zeros. Thus $x(n) = \{1, 1, 1, 1, 0, 0, 0\}$ and $y(n) = \{0, 1, 2, 3, 0, 0, 0\}$. If one now takes the DFT of $x(n)$ and $y(n)$ to obtain $X(k)$ and $Y(k)$, then takes the IDFT of $X(k)Y(k)$, the correct answer will be obtained.

In practice, $x(n)$ can be a long signal in the time domain and $y(n)$ may be the filter impulse response with relatively short time response. Under this condition, it is impractical to pad the data with zeros. First, the signal could be very long and the required DFT too computationally intensive. Second, in this operation it is necessary to collect all the data of the signal. If the signal is very long, it is impossible to collect all the data before performing the linear convolution. The circular convolution can be used to obtain the linear convolution in this case. There are two approaches, overlap-add and overlap-save, to achieve this goal and they can be found in Rabiner [14] and Oppenheim and Schaffer [15] and not discussed here.

3.10 Window Functions [17]

In the last section, it was shown that when the rectangular window in the time domain does not match an integer number of cycles in the input signal, discontinuity will occur at the window edges. This discontinuity will generate spurious spectrum lines that do not exist in the true Fourier transform. The window size in the time domain is often prefixed. Therefore it is highly unlikely that the window length will match an integer number of periods of the input signal.

It can be clearly seen from Figure 3.13(b) that the mismatch occurs at the ends of the rectangular window. One obvious way to reduce the discontinuity is to reduce the amplitudes at the end of the time window to minimize or eliminate the discontinuity. The basic idea is to create some special windows other than the rectangular one discussed previously. Figure 3.17(a) shows the discontinuity at the ends of a rectangular window and 3.17(b)

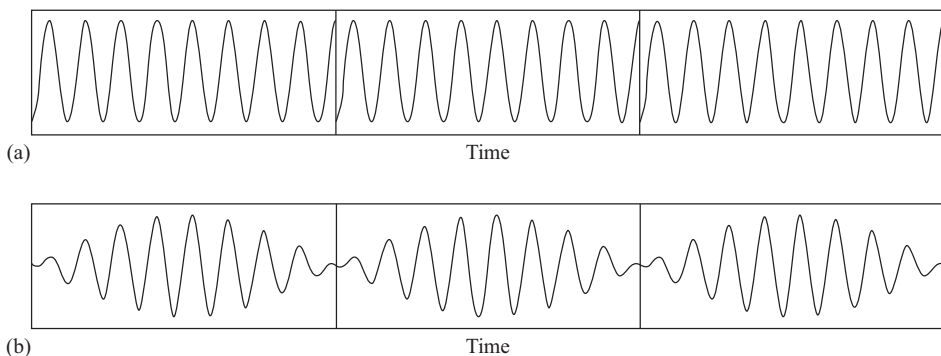


Figure 3.17 Discontinuity at the ends of the windows: (a) rectangular window and (b) cosine window.

shows the results of a cosine window. Although in both cases the signals are discontinuous at the edges of the windows, the cosine window reduces the severity of the discontinuity.

A more commonly used approach to explain the leakage effect is through convolution. As discussed in Section 3.8, selecting a certain length of data is equivalent to multiplying the data by a rectangular window. This operation is equivalent to convolving the signal with a sinc function in the frequency domain. Since the sinc function contains many sidelobes, these sidelobes will appear in the frequency domain. These sidelobes only disappear when the rectangular window length matches the period of the input sinusoidal signal, because the period of the comb function used in the frequency domain sampling matches the zeros in the sinc function.

Under general sampling conditions, in order to reduce the output sidelobes in the frequency domain, a window with low sidelobes in the frequency domain will be used. The general approach to designing a window is to minimize the amplitude of the sidelobes. However, lowering the sidelobes will increase the width of the mainlobe, which is an undesirable effect. One way to evaluate the performance of the window is to measure the width of the main spectrum lobe and the amplitudes of the sidelobes. Low sidelobes result in a high dynamic range for the EW receiver, while a wide mainlobe width reflects poor frequency resolution. Therefore, selecting a window is normally based on the trade-off between frequency resolution and dynamic range.

Usually the frequency domain response determines the performance of the window. The time domain is also very important since it determines the transient effect. The time and frequency responses of the window functions are related by the Fourier transform. Some common window functions are discussed below.

3.10.1 Rectangular Window

A rectangular window is often used as a reference to evaluate other types of window functions. The frequency domain response can be obtained from the Fourier transform of a rectangular function. The width of the mainlobe, that is, from two nodes of the mainlobe, equals to $N/2$. In comparisons of windows, this width is often considered as unity. Figure 3.18 shows the frequency response of a rectangular window. In order to obtain fine resolution in the frequency domain, that is, the sidelobes, the figure is obtained through zero padding, which is discussed in Chapter 4. In this plot, let us define $N = 128$ as the window width and $31 \times N (= 3968)$ of zeros are added after the data, as shown in Figure 3.18(a). The peak of the rectangular window in the frequency domain is at zero, that is, around $X(0)$ and $X(32N - 1)$ as shown in Figure 3.18(b). Since the peak appears at the two ends, it is difficult to see the window shape. The peak can be shifted toward the center of the figure so that the entire peak is displayed undivided. In order to shift the peak to the center of the figure at $X(16N)$, the points from $X(16N)$ to $X(32N - 1)$ are plotted then followed by $X(0)$ to $X(16N - 1)$. The result is shown in Figure 3.18(c). Figure 3.18(d) plots only the center portion around the peak. The width of the mainlobe is $N/2$, and this is the narrowest frequency response in comparison with other windows. The highest sidelobe is about 13 dB down from the mainlobe. Other windows are designed to reduce the sidelobe magnitude.

The zero padding scheme and shifting of the frequency domain data are applied to all the windows in the following discussions in order to produce a better visual display of the shapes of the windows. Many different windows are listed in Harris [17].

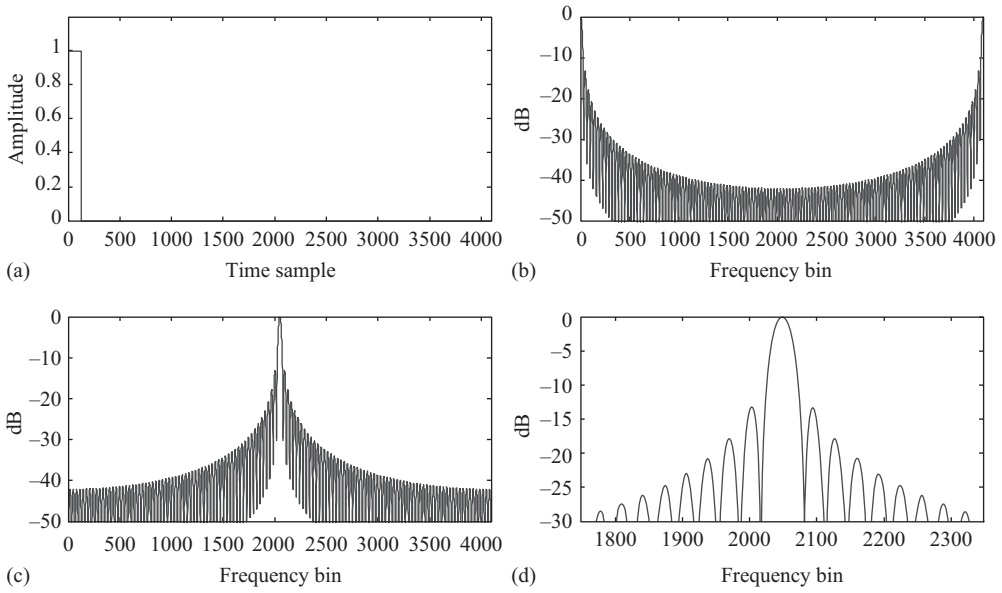


Figure 3.18 Response of a rectangular window: (a) time domain response, (b) frequency domain response, (c) frequency response after shift, and (d) frequency response near the peak.

3.10.2 Gaussian Window

A Gaussian is written as

$$w(n) = e^{-\frac{1}{2} \left[\frac{2a \left(n - \frac{N}{2} \right)}{N} \right]^2}, \quad (3.109)$$

where a is a constant and N is the total number of points in the window. It can be proved that the Fourier transform of a Gaussian function (with an infinite duration) in the time domain is a Gaussian function in the frequency domain. Theoretically a Gaussian window should not produce any sidelobes. However, in actual applications the Gaussian window must be truncated to a certain length for the DFT operation. This truncation will generate sidelobes in the frequency domain. When $a = 2.5$, the width of the mainlobe is about 1.33 times the rectangular window. The highest sidelobe is 42 dB down from the mainlobe. This result is shown in Figure 3.19. It should be noted that although the data in the time domain are zero padded, the zeros are not shown in Figure 3.19(a).

3.10.3 Cosine Window Raised to the a th Order

This window function is written as

$$w(n) = \cos^a \left[\frac{\pi \left(n - \frac{N}{2} \right)}{N} \right]. \quad (3.110)$$

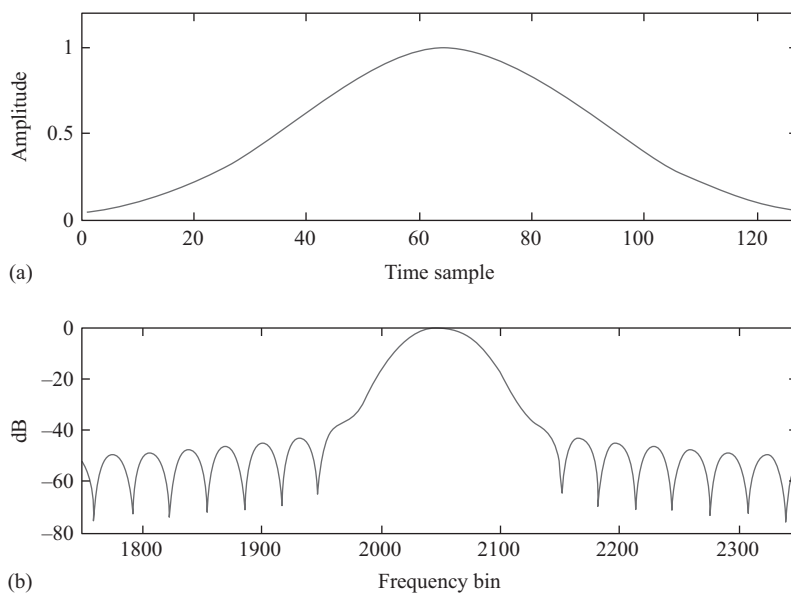


Figure 3.19 Response of a Gaussian window: (a) time domain and (b) frequency domain.

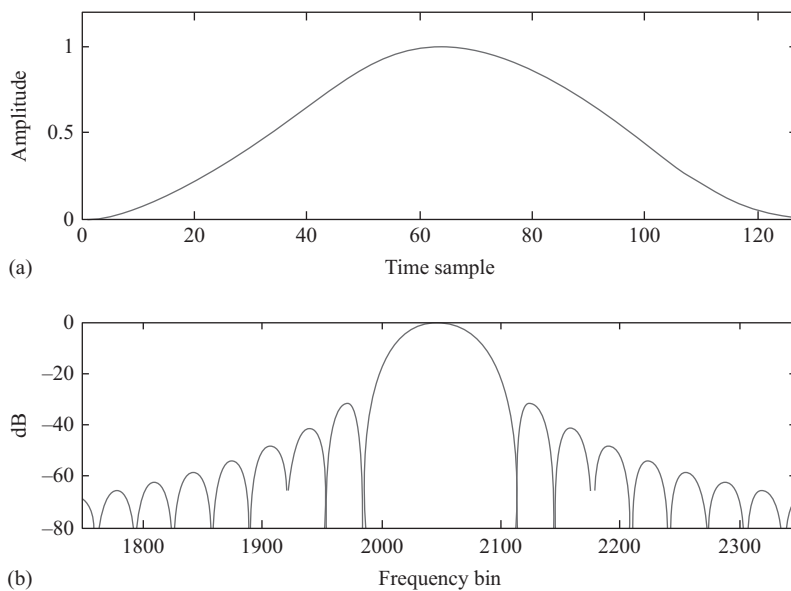


Figure 3.20 Cosine window raised to the a th order ($a = 2$): (a) time domain and (b) frequency domain.

When $a = 2$, the window function is called the Hanning window (named after Julius von Hann). The width of the mainlobe is 1.5 times the rectangular window and the highest sidelobe generated by the Hanning window is -32 dB. Figure 3.20 shows the time and frequency domain responses.

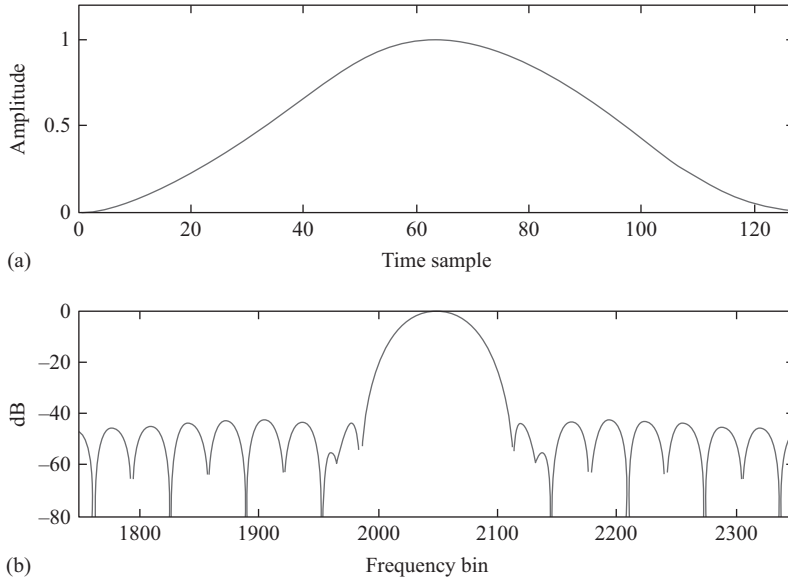


Figure 3.21 (a) Time domain and (b) frequency domain responses of a Hamming window.

3.10.4 Generalized Hamming window

This window function is written as

$$w(n) = a + (1 - a) \cos \left[\frac{2\pi(n - \frac{N}{2})}{N} \right], \quad (3.111)$$

where $a < 1$. When $a = 0.54$, the window is referred to as the Hamming window (named after Richard W. Hamming.). The mainlobe width is 1.36 and the highest sidelobe is -43 dB. The time and frequency domain responses of the Hamming window are illustrated in Figure 3.21. It is interesting to note that all the sidelobes shown have approximately the same amplitude.

When $a = 0.5$, the generalized Hamming window becomes a Hanning window, since

$$\cos^2 \left[\frac{\pi(n - \frac{N}{2})}{N} \right] = \frac{1}{2} \left\{ 1 + \cos \left[\frac{2\pi(n - \frac{N}{2})}{N} \right] \right\}. \quad (3.112)$$

Note that not only the names of the windows (Hamming and Hanning) are close, but the functions can also be related under special conditions.

The Hamming window has lower sidelobes than the Hanning window. Assume that both the Hamming and Hanning windows have N points. Since the endpoints of the Hanning window are zero, in reality the Hanning window has only $N - 2$ points. This is the reason that Hanning windows have higher sidelobes, because they are shorter than Hamming windows. In fact, this is the reason why the Hamming window is also called the raised cosine window.

3.11 FFT [8, 14, 15, 18–41]

In general, the DFT is computationally intensive. For example, to compute the DFT for N data points, (3.99) will require N multiplications and summations to calculate one component

in the frequency domain. This implies that N^2 operations are needed in the complete DFT. In 1965, Cooley and Tukey discovered a very efficient way to calculate the DFT based on the symmetry of the $e^{-j2\pi nk}$ kernel. This discovery has allowed the DFT to become the preferred method of analyzing the spectrum of signals. This method was later referred to as the FFT, because of the improvement in calculation speed. The FFT can be simply considered as an efficient way to calculate the DFT. The results obtained from the FFT are identical to those obtained from the DFT for 2^m data points. All the characteristics and properties of the DFT are identical in the FFT.

3.11.1 Derivation of the FFT

The following derivation is based on the derivation in Chassaing and Reay [41]. First, we introduced a new variable W_N :

$$W_N = e^{\frac{-j2\pi}{N}}. \quad (3.113)$$

The following two properties of W_N will be used to derive the FFT:

$$W_N^{k+N} = W_N^k \quad (3.114)$$

and

$$W_N^{k+\frac{N}{2}} = -W_N^k. \quad (3.115)$$

N should be powers of 2. The basic concept of the FFT is “divide and conquer.” Instead of conducting one N -point DFT, we would like to conduct two $N/2$ -point DFTs. Instead of conducting two $N/2$ -point DFTs, we would like to conduct four $N/4$ -point DFTs. This process continues until a 2-point DFT is the only operation that needs to be carried out.

The DFT in (3.99) can then be rewritten using W_N as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \quad 0 \leq k \leq N-1. \quad (3.116)$$

Equation (3.116) can then be written as

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} x(n) W_N^{kn} + \sum_{n=\frac{N}{2}}^N x(n) W_N^{kn} = \sum_{n=0}^{\frac{N}{2}-1} x(n) W_N^{kn} + W_N^{\frac{kN}{2}} \sum_{n=0}^{\frac{N}{2}-1} x(n) W_N^{kn}. \quad (3.117)$$

Because

$$W_N^{\frac{N}{2}} = e^{-j\pi} = -1, \quad (3.118)$$

(3.117) can be written as

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} \left[x(n) + (-1)^k x\left(n + \frac{N}{2}\right) \right] W_N^{kn}. \quad (3.119)$$

When k is an even number ($k = 2k'$), (3.119) becomes

$$X(2k') = \sum_{n=0}^{\frac{N}{2}-1} \left[x(n) + x\left(n + \frac{N}{2}\right) \right] W_N^{2k'n}. \quad (3.120)$$

When k is an odd number ($k = 2k' + 1$), (3.119) becomes

$$X(2k' + 1) = \sum_{n=0}^{\frac{N}{2}-1} \left[x(n) - x\left(n + \frac{N}{2}\right) \right] W_N^n W_N^{2k'n}. \quad (3.121)$$

Since

$$W_N^2 = e^{(-\frac{j2\pi}{N})^2} = e^{(-\frac{j2\pi}{N})} = W_{N/2}, \quad (3.122)$$

(3.120) and (3.121) can be written as

$$X(2k') = \sum_{n=0}^{\frac{N}{2}-1} \left[x(n) + x\left(n + \frac{N}{2}\right) \right] W_{N/2}^{k'n} \quad (3.123)$$

and

$$X(2k' + 1) = \sum_{n=0}^{\frac{N}{2}-1} \left[x(n) - x\left(n + \frac{N}{2}\right) \right] W_{N/2}^{k'n}. \quad (3.124)$$

Check (3.123) and (3.124) and note that by rearranging the first half of the input data ($0 \leq n \leq \frac{N}{2} - 1$) and the second half of the input data ($\frac{N}{2} \leq n \leq N - 1$), the DFT result with even indexes can be calculated as an $N/2$ -point DFT and the DFT result with odd indexes can be calculated as another $N/2$ -point DFT. An 8-point DFT consisting of two 4-point DFTs is illustrated in Figure 3.22(a). If we keep this “divide and conquer” approach, we end up with a 2-point DFT operation:

$$X(k) = \sum_{n=0}^1 x(n)(-1)^n, \quad 0 \leq k \leq 1 \Rightarrow \begin{cases} X(0) = x(0) + x(1) \\ X(1) = x(0) - x(1) \end{cases} \quad (3.125)$$

The 2-point DFT symbol is illustrated in Figure 3.22(b), which is often referred to as the radix 2 FFT or the FFT butterfly. For an 8-point FFT, there are three layers of FFT butterflies and the last operation shown is the reordering. The reordering process is referred to as “bit reversal,” because if we write the FFT output position as a binary number as $b_N b_{N-1} \dots b_0$ (top output as $00 \dots 0$), then its corresponding FFT element index is $b_0 b_1 \dots b_N$.

To implement an N -point DFT ($N = 2^m$) using (3.99), the number of required operations is N^2 . The FFT reduces the number of operations to $(N/2)\log_2(N)$ complex multiplications, $(N/2)\log_2(N)$ additions and $(N/2)\log_2(N)$ subtractions. This reduction can be found in Figure 3.23.

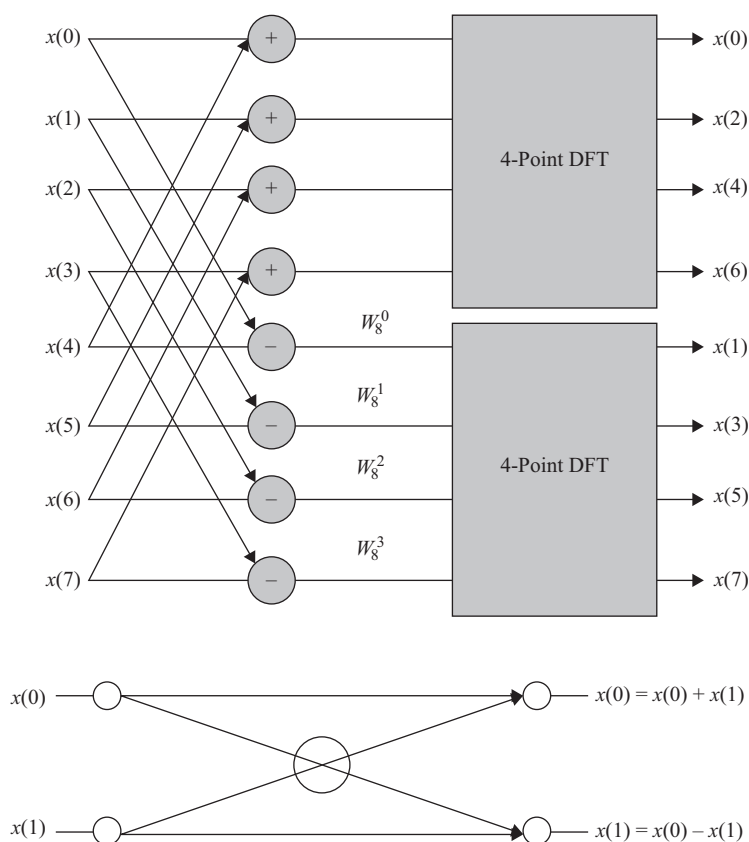


Figure 3.22 FFT butterfly: (a) an 8-point DFT consisting of two 4-point DFTs and (b) a 2-point FFT butterfly.

From this figure, one can see that the savings in the number of operations is very dramatic. The number of multiplications and additions for FFT operations can be found in Duhamel [40].

This same approach can be applied to the inverse FFT (IFFT). The only difference is changing $-j$ in (3.113) to $+j$; the rest of the discussion stays the same.

The number of input data must be a power of 2. Since the discovery of the FFT, many different approaches similar to the original Cooley–Tukey approach have been proposed. FFTs with input data points other than a base 2 number have been developed. In some designs the output is in the correct order and no bit reversal is required; however, in those approaches the input data points must be rearranged [8, 41]. Thus there is no significant advantage. Today, it appears that the most popular FFT algorithm is still the original Cooley–Tukey approach. The most promising approach for microwave receiver applications might still be FFT chips built in the hardware. If their operating speeds can match the ADC, one ADC can be followed by one FFT chip. If the FFT speed is less than the ADC, a number of FFT chips can be connected and operated in parallel, or some other approach such as the decimation technique can be used.

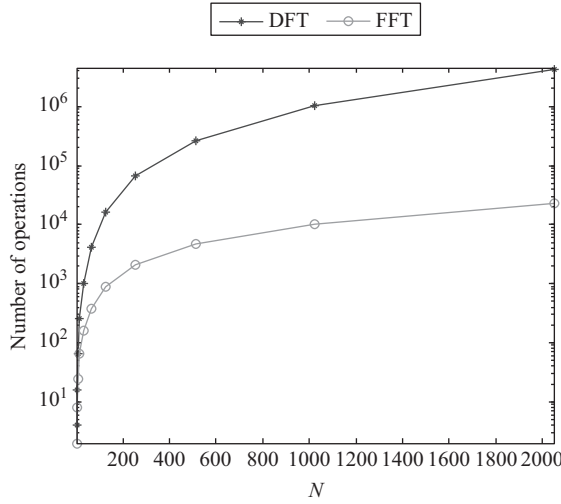


Figure 3.23 The number of operations required for computing an FFT and DFT.

3.11.2 Real Input Computed by a Complex FFT Operator [14]

Usually an FFT operator is designed to process complex data with a real and an imaginary part. However, in many implementations the input is available only in real form, because it is difficult to collect complex data with well-balanced in-phase (I) and quadrature-phase (Q) outputs. The I and Q channels are discussed in detail in Chapter 6. The real-valued data can be used as the input to the real port of the FFT operator and zeros can be used as input to the imaginary port; however, this arrangement does not use the resources efficiently.

This section will discuss the use of one FFT operator to process two sets of real-valued data simultaneously. If $x(n)$ is real-valued data, the FFT of the k th component is

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi nk/N} \quad (3.126)$$

and the $(N - k)$ th component can be written as

$$X(N - k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi n(N-k)/N} = \sum_{n=0}^{N-1} x(n)e^{j2\pi nk/N} = \sum_{n=0}^{N-1} x^*(n) \left[e^{j2\pi nk/N} \right]^*, \quad (3.127)$$

because $x(n) = x(n)^*$, where $*$ is the complex conjugate. It is obvious that

$$X(k) = X(N - k)^*$$

or

$$\begin{aligned} \operatorname{Re}[X(k)] &= \operatorname{Re}[X(N - k)] \\ \operatorname{Im}[X(k)] &= -\operatorname{Im}[X(N - k)], \end{aligned} \quad (3.128)$$

where Re and Im represent the real and imaginary parts of the function, respectively. The real part of the FFT is symmetric and the imaginary is antisymmetric with respect to the index $(N - 1)/2$.

If $y(n)$ is another real function and has the same number of data points as $x(n)$, one can form a complex function $z(n)$ such that

$$\begin{aligned} z(n) &= x(n) + jy(n) \\ Z(k) &= X(k) + jY(k) = Re[X(k)] + jIm[X(k)] + j\{Re[Y(k)] + jIm[Y(k)]\} \\ &= Re[X(k)] - Im[Y(k)] + jIm[X(k)] + jRe[Y(k)]. \end{aligned} \quad (3.129)$$

This equation can be written as

$$\begin{aligned} Re[Z(k)] &= Re[X(k)] - Im[Y(k)] \\ Im[Z(k)] &= Im[X(k)] + Re[Y(k)]. \end{aligned} \quad (3.130)$$

Since $x(n)$ and $y(n)$ are real-valued functions, the real part of $X(k)$ and $Y(k)$ are symmetric with respect to $(N - 1)/2$ and the imaginary parts are antisymmetric. The second half of the k components can be written in terms of the first half by using (3.130) and (3.128) as

$$\begin{aligned} Re[Z(N - k)] &= Re[X(k)] + Im[Y(k)] \\ Im[Z(N - k)] &= -Im[X(k)] + Re[Y(k)] \end{aligned} \quad (3.131)$$

From (3.130) and (3.131), it is easily seen that

$$\begin{aligned} Re[X(k)] &= \frac{Re[Z(k)] + Re[Z(N - k)]}{2} \\ Im[Y(k)] &= \frac{Re[Z(N - k)] - Re[Z(k)]}{2} \\ Re[Y(k)] &= \frac{Im[Z(k)] + Im[Z(N - k)]}{2} \\ Im[X(k)] &= \frac{Im[Z(k)] - Im[Z(N - k)]}{2}. \end{aligned} \quad (3.132)$$

This is the desired result.

One can divide the real-valued data $x(n)$ into two sets: $x_1(n)$ and $x_2(n)$ and apply $x_1(n)$ to the real input port of the FFT and $x_2(n)$ to the imaginary port. The output of the FFT is represented by $Z(k)$. The real and imaginary parts of $X_1(k)$ and $X_2(k)$ can be found from (3.132).

3.12 Possible Advantages of DFT over FFT in Receiver Applications [40, 42]

Although FFT requires fewer operations than DFT, for wideband microwave receiver applications it is possible that DFT may have some advantages over FFT under certain conditions. One should note that in Figure 3.23, when $N = 2048$ there is a large advantage of FFT over DFT. The savings in computation can be expressed as the ratio of $N/\log_2 N$. At $N = 32, 64$, and 128 , these savings are 6.4, 10.67, and 18.29, respectively.

If the microwave receivers are for EW applications, at least in the near future, the number of input data will probably be 32, 64, or 128 points. The input data length is limited by two factors: the operating speed and the minimum pulse width. If the collected data length is much longer than the input pulse width, the additional data contain only noise. The data points without signal will degrade the sensitivity of the receiver. With the present ADC operating speed, it is likely that on a short pulse only tens or a few hundred data points can be collected. Therefore the DFT and FFT operations are compared for these short data points. The two possible advantages of DFT over FFT are discussed below.

3.12.1 Initial Data Accumulation

It should be noted that the initial data accumulation in the time domain is different for the DFT and FFT. When an N -point FFT is calculated, all N points of data must be available. In contrast, from (3.99), the data can be collected and processed as they arrive. Each input data point $x(n)$ is multiplied by N terms. For example, if $N = 64$, when the first data point $x(0)$ is collected, 64 intermediate data points $X_i(n, k)$ can be generated from (3.99) as

$$X_i(n, k) = x(0)e^{\frac{-j2\pi k0}{N}} = x(0), \quad (3.133)$$

where $n = 0$ and $k = 0$ to 63. Similar operations can be applied to all the input data. For the n th data $x(n)$, the intermediate data $X_i(n, k)$ is

$$X_i(n, k) = x(n)e^{\frac{-j2\pi kn}{N}}, \quad (3.134)$$

where $k = 0$ to 63. When the last data point is collected, there is a total of N^2 intermediate data points. To find the k th components in the frequency domain, all the intermediate data with the same k value summed together will produce the required value.

Of course, there are N^2 operations in the above approach. But as soon as the last input data are collected, a summation will produce the desired result. For the FFT approach, it must wait until all the data are collected before the operation can be started. Thus for small N values, the DFT approach might be easier to build as a real-time processor.

3.12.2 Sliding DFT

In an EW receiver, the input data may or may not contain signal. One common way is to divide the input data from index 0 to index $N - 1$, then from index N to index $2N - 1$, and so on. This approach saves processing time, but the signal may be divided into two data groups, thus the sensitivity and frequency resolution of the receiver are low. One method of improving this approach is to process the data with some degree of overlap. For a 50% overlap, the first group will contain points 0 to $N - 1$ and the second group $N/2$ to $(3N/2) - 1$. This approach will double the required processing, but will improve receiver performance. The overlapping of data is further discussed in Chapter 10.

The most sophisticated approach is to slide the DFT one point at a time. For example, the first group contains data whose index is from 0 to $N - 1$, the second group contains data whose index is from 1 to N , and the third group contains data whose index is from 2 to $N + 1$, as shown in Figure 3.24. Although this approach may provide the best receiver performance, it is processing intensive. If the FFT is used for this approach, each set of input

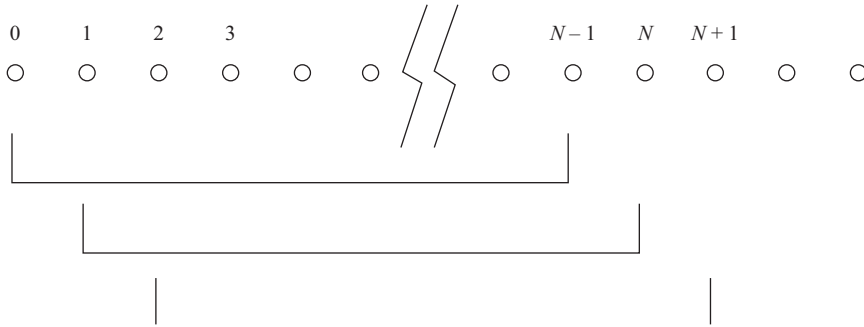


Figure 3.24 A sliding DFT.

data will be calculated independently. If the DFT is used for this type of processing, the new information can be obtained from the previous operation. To demonstrate this operation, let us rewrite (3.99) as

$$X_0(k) = x(0) + x(1)e^{\frac{-j2\pi k}{N}} + x(2)e^{\frac{-j2\pi k2}{N}} + \cdots + x(N-1)e^{\frac{-j2\pi k(N-1)}{N}}, \quad (3.135)$$

where $X_0(k)$ is used to represent the DFT of the first set of data. This is exactly the same equation except in slightly different form. The second set of data will generate a DFT as

$$X_1(k) = x(1) + x(2)e^{\frac{-j2\pi k}{N}} + x(3)e^{\frac{-j2\pi k2}{N}} + \cdots + x(N)e^{\frac{-j2\pi k(N-1)}{N}}. \quad (3.136)$$

The only difference between (3.135) and (3.136) is the set of data points. The exponential terms of the two equations are identical. One can see that (3.135) can be modified slightly to obtain (3.136). This relation can be written as

$$X_1(k) = [X_0(k) - x(0)]e^{\frac{j2\pi k}{N}} + x(N)e^{\frac{-j2\pi k(N-1)}{N}} = [X_0(k) - x(0) + x(N)]e^{\frac{j2\pi k}{N}}. \quad (3.137)$$

This approach can be generalized as

$$X_{m+1}(k) = [X_m(k) - x(m) + x(N+m)]e^{\frac{j2\pi k}{N}}. \quad (3.138)$$

It appears that this approach might be more efficient than the FFT approach. If this operation can be performed efficiently, a real-time sliding FFT is possible.

3.13 Summary

In this chapter both the continuous Fourier transform and DFT were introduced. The continuous Fourier transform provides a foundation to analyze signals in the frequency domain and the DFT provides a practical approach to apply Fourier theory in receiver design. The DFT can be considered as an approximation of Fourier transform and the differences between them should not be ignored. We discussed some of these differences in this chapter. One way to reduce leakage effects caused by the DFT is by using different windows when sampling data, and several different windows were introduced in the chapter. The FFT is an efficient algorithm for calculating the DFT in real-time applications, which was also described in this chapter. The Fourier transform-related operation for EW receivers is covered in the next chapter.

References

- [1] Campbell GA, Foster RM. *Fourier Integrals for Practical Applications*. New York: Van Nostrand Reinhold; 1948.
- [2] Erdelyi A (ed.). *Table of Integral Transforms*, Vol. 1. New York: McGraw-Hill; 1954.
- [3] Dwight HB. *Tables of Integrals and Other Mathematical Data*, 4th ed. New York: MacMillan; 1961.
- [4] Robinson EA. 'A historical perspective of spectrum estimation'. *Proceedings of the IEEE* 1982;**70**(9):885–907.
- [5] Papoulis A. *The Fourier Integral and Its Applications*. New York: McGraw-Hill; 1962.
- [6] Papoulis A. *Probability, Random Variables, and Stochastic Processes*. New York: McGraw-Hill; 1965.
- [7] Bracewell R. *The Fourier Transform and Its Applications*. New York: McGraw-Hill; 1965.
- [8] Brigham EO. *The Fast Fourier Transform*. Englewood Cliffs, NJ: Prentice-Hall; 1973.
- [9] Carlson AB. *Communication Systems: An Introduction to Signals and Noise in Electrical Communication*. New York: McGraw-Hill; 1975.
- [10] Stremmer FG. *Introduction to Communication Systems*, 3rd ed. Reading, MA: Addison-Wesley; 1990.
- [11] Ziemer RE, Tranter WH. *Principles of Communications: Systems, Modulation, and Noise*, 3rd ed. Boston: Houghton Mifflin; 1990.
- [12] Stimson GW. *Introduction to Airborne Radar*. El Segundo, CA: Hughes Aircraft Co.; 1983.
- [13] Skolnik MI. *Introduction to Radar Systems*. New York: McGraw-Hill; 1962.
- [14] Rabiner LR. *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1975.
- [15] Oppenheim AV, Schaffer RW. *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1975.
- [16] Tsui JB. *Fundamentals of Global Positioning System Receivers*, 2nd ed. Hoboken, NJ: John Wiley & Sons; 2005.
- [17] Harris FJ. 'On the use of windows for harmonic analysis with the discrete Fourier transform'. *Proceedings of the IEEE* 1978;**66**(1):51–83.
- [18] Kay SM. *Modern Spectral Estimation Theory and Application*. Englewood Cliffs, NJ: Prentice Hall; 1988.
- [19] Marple SL Jr. *Digital Spectral Analysis with Applications*. Englewood Cliffs, NJ: Prentice Hall; 1987.
- [20] Elliott DF (ed.). *Handbook of Digital Signal Processing: Engineering Applications*. San Diego: Academic Press; 1987.
- [21] Tsui JBY. *Digital Microwave Receivers: Theory and Concepts*. Norwood, MA: Artech House; 1989.
- [22] Tukey JW. 'An introduction to the calculations of numerical spectrum analysis', in Harris B (ed.), *Spectral Analysis of Time Series*. New York: John Wiley & Sons; 1967.
- [23] Blackman RB, Tukey JW. 'The measurement of power spectra from the point of view of communications engineering'. *Bell System Technical Journal* 1958;**37**(1):185–282.

- [24] Cooley JW, Tukey JW. ‘An algorithm for the machine calculation of complex Fourier series’. *Mathematics of Computation* 1965;**19**(90):297–301.
- [25] Welch PD. ‘The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms’. *IEEE Transactions on Audio and Electroacoustics* 1967;**15**(2):70–73.
- [26] Cooley JW, Lewis PAW, Welch PD. ‘Historical notes on the fast Fourier transform’. *Proceedings of the IEEE* 1967;**55**(10):1675–1677.
- [27] Cochran WT, Cooley JW, Favon DL, *et al.* ‘What is the fast Fourier transform?’. *Proceedings of the IEEE* 1967;**55**(10):1664–1674.
- [28] Brigham EO, Morrow RE. ‘The fast Fourier transform’. *IEEE Spectrum* 1967;**4**(12):63–70.
- [29] Cooley JW, Lewis PAW, Welch PD. ‘The finite Fourier transform’. *IEEE Transactions on Audio and Electroacoustics* 1969;**17**(2):77–85.
- [30] Bergland GD. ‘A guided tour of the fast Fourier transform’. *IEEE Spectrum* 1969;**6**(7):41–52.
- [31] Sanderson RB. ‘Instrumental techniques’, in Rao RN, Matthews CW (eds.), *Molecular Spectroscopy: Modern Research*. San Diego: Academic Press; 1972.
- [32] Palmer LC. ‘Coarse frequency estimation using the discrete Fourier transform’. *IEEE Transactions on Information Theory* 1974;**20**(1):104–109.
- [33] Allen JB. ‘Short term spectral analysis, synthesis, and modification by discrete Fourier transform’. *IEEE Transactions on Acoustics, Speech and Signal Processing* 1977;**25**(3):235–238.
- [34] Sorensen HV, Jones DL, Heideman M, Burrus CS. ‘Real-valued fast Fourier transform algorithm’. *IEEE Transactions on Acoustics, Speech and Signal Processing* 1987;**35**(6):849–863.
- [35] Thong T. ‘Practical consideration for a continuous time digital spectrum analyzer’. *Proceedings of the IEEE International Symposium on Circuits and Systems*. New York: IEEE; 1989:1047–1050.
- [36] Jenq Y-C. ‘Digital spectra of nonuniformly sampled signals: a robust sampling time offset estimation algorithm for ultra high-speed waveform digitizers using interleaving’. *IEEE Transactions on Instrumentation and Measurement* 1990;**39**(1):71–75.
- [37] Agoston M, Henricksen R. ‘Using digitizing signal analyzers for frequency domain analysis’. *Microwave Journal* 1990;**33**(9):181–189.
- [38] Fine B. ‘DSPs address real-world problems’. *Microprocessor* 1990;72–74.
- [39] Sayegh SI. ‘A pipeline processor for mixed-size FFTs’. *IEEE Transactions on Signal Processing* 1992;**40**(8):1892–1900.
- [40] Duhamel P. ‘Implementation of “Split-radix” FFT algorithms for complex, real, and real-symmetric data’. *IEEE Transactions on Acoustics, Speech and Signal Processing* 1986;**34**(2):285–295.
- [41] Chassaing R, Reay D. *Digital Signal Processing and Applications with the TMS320C6713 and TMS320C6416 DSK*, 2nd ed. Hoboken, NJ: John Wiley & Sons; 2008.
- [42] Springer T. ‘Sliding FFT computes frequency spectra in real time’. *Electronic Design News* 1988; **September**:161–170.

Fourier Transform-Related Operations

4.1 Introduction

In this chapter, the operations related to the Fourier transform are discussed. Some of these operations are useful to electronic warfare (EW) receiver design and some of these techniques can improve the results of the fast Fourier transform (FFT). The topics include periodograms, zero padding, better localization of FFT peaks, autocorrelation, a phase sampling scheme to increase the bandwidth of the receiver, and a decimation scheme of the discrete Fourier transform (DFT) for EW receivers.

4.2 Periodogram [1–7]

Once the FFT or DFT of the digitized input signal is completed, the next step is to find the spectrum or the frequency components. The straightforward way is to find the power spectrum. Since the FFT output is complex-valued, the power spectrum can be written as

$$P(k) = \frac{1}{N} |X(k)|^2, \quad (4.1)$$

where $X(k)$ is the FFT of $x(n)$. This is referred to as the periodogram.

For real-valued signals, its power spectrum has an even symmetry. This can be shown as follows. The frequency component $X(k)$ and $X(N - k)$ of a real-valued signal can be written as

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-\frac{j2\pi nk}{N}} \quad (4.2)$$

$$X(N - k) = \sum_{n=0}^{N-1} x(n) e^{-\frac{j2\pi n(N-k)}{N}} = \sum_{n=0}^{N-1} x(n) e^{\frac{j2\pi nk}{N}} = X(k)^*. \quad (4.3)$$

Therefore

$$|X(k)|^2 = |X(N - k)|^2. \quad (4.4)$$

When $k = N/2$, $X(k)$ equals $X(N - k)$, therefore the output is symmetrical with respect to $X(N/2)$. Thus if there are N points of real-valued data, after the FFT, the power spectrum has N frequency components. However, among the N frequency components, only half of them provide useful information. The second-half of the power spectrum components equals the first half. As a result, if there are N points of real-valued input data, even though after the FFT $X(k)$ has N components, it is common to plot only $N/2$ components.

4.2.1 Averaged Periodogram [4, 5, 8, 9]

If the input signal contains only noise with variance σ^2 , one might expect that when the input data length increases, the variance of the power spectrum approaches zero. However, this is not true. When the data length increases, the variability in the power spectrum does not decrease. This phenomenon is illustrated in Figure 4.1. In this figure, 1024 points of data are collected and the data contain only noise. The data are divided into eight subgroups and each contains 128 points. Figure 4.1(a) shows the square root of the periodogram of all 1024 data points. Figure 4.1(b) shows the square root of the periodogram of the first 128 data points. The remaining seven sets of data points will show similar results. It is clearly shown that more data points used in the calculation do not decrease the noise variability in the frequency domain.

In order to reduce the variance of noise in the output in the above example, one can calculate the periodograms of all eight subgroups of data and take the average of the results. The operation of this special case can be written as

$$P_{av}(k) = \frac{1}{8} \sum_{i=0}^7 X_i(k), \quad (4.5)$$

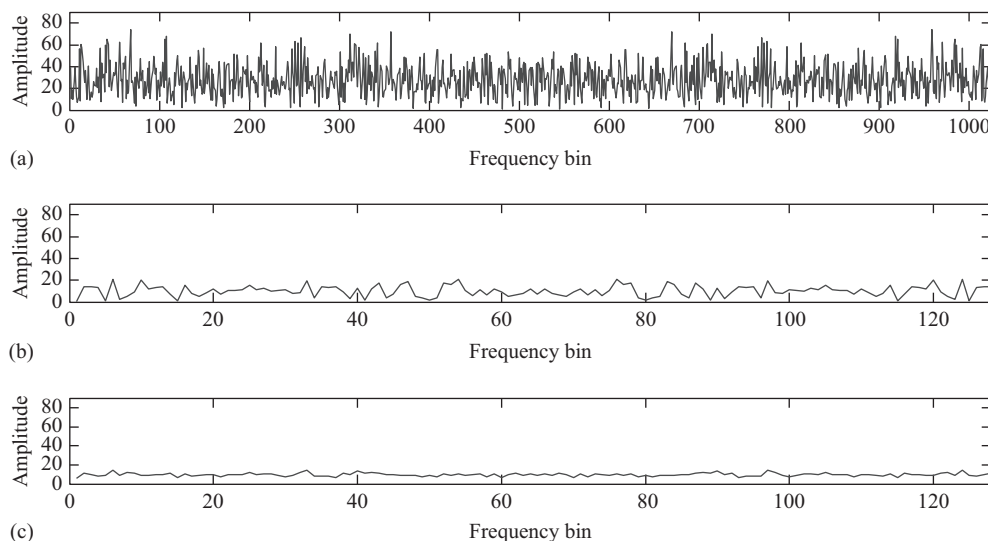


Figure 4.1 Periodogram of noise-only input: (a) 1024 data points, (b) 128 data points, and (c) 1024 data points divided into eight subgroups.

where

$$X_i(k) = \frac{1}{8} \sum_{n=0}^{127} x(128i + n) e^{-\frac{j2\pi nk}{N}}, \quad i = 0, 1, \dots, 7.$$

Figure 4.1(c) shows the square root of the averaged periodogram. Comparing with Figures 4.1(a) and 4.1(b), the variance of the noise spectrum is reduced through this averaging process. Because the eight sets of data can be considered independent, the sum will increase the noise by $\sqrt{8}$. When this result is divided by 8, the variance of the noise will decrease.

Now let us change the input data to a sine wave contaminated with noise. A total of 1024 data points are collected. The same three approaches are used to process the input data. First, the periodogram is obtained by using all the data points. Second, the periodogram of the first 128 data points is obtained. Finally, the input data are divided into eight subgroups and their periodograms are obtained and averaged. The square roots of the periodograms are shown in Figure 4.2. In Figure 4.2(a), the signal frequency is clearly visible for the 1024-point FFT. In Figure 4.2(b), the signal frequency is difficult to recognize for the 128-point FFT. In Figure 4.2(c) the signal frequency is clearly shown for the averaged case, but the frequency resolution is not as good as that in Figure 4.2(a).

Although averaging the periodogram can decrease the noise contribution, it sacrifices frequency resolution. In general, if the desired frequency resolution determined by the FFT length is reached, additional sets of data can be collected and, through averaging, the signal-to-noise (S/N) ratio can be improved.

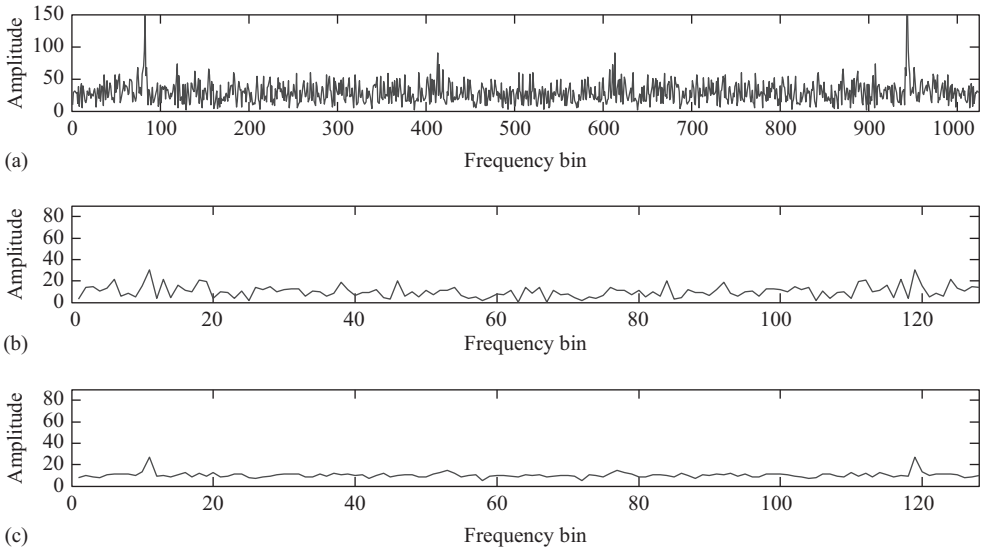


Figure 4.2 Periodogram of data contaminated with noise (a) 1024 data points (b) 128 data points (c) 1024 data points divided into 8 subgroups.

4.3 Zero Padding [4, 5]

Zero padding is the addition of zeros at the end of a digital data string before the FFT is performed. For example, if 64 points of data are collected, usually a 64-point FFT is performed to obtain the frequency. However, one can add 64 zeros at the end of the data string and perform a 128-point FFT. For the 64 data points with 64 zeros, the number of output frequency components is doubled. Notice that zero padding does not add any new information and the resultant FFT form does not change.

In general, suppose there are N data points $x(0)$ to $x(N - 1)$, and we add N zero points from $x(N)$ to $x(2N - 1)$ and perform a $2N$ -point DFT. The result will be

$$X(k) = \sum_{n=0}^{2N-1} x(n) e^{-j2\pi kn/2N}. \quad (4.6)$$

Note that the summation is from 0 to $2N - 1$ and the kernel is $e^{-j2\pi kn/2N}$ rather than $e^{-j2\pi kn/N}$. Since $x(N) = x(N + 1) = \dots = x(2N - 1) = 0$, this equation can be written as

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/2N}. \quad (4.7)$$

The only change from (4.6) is in the limit of the summation. It should be noted that the value of k in $X(k)$ is from $k = 0$ to $2N - 1$. Thus the number of spectrum lines is $2N$, which doubles the result of an N -point FFT.

In (4.6), if we consider only the even number of spectrum lines, one can write $k = 2k'$. Substituting this relation in (4.7), the result is

$$X(k') = \sum_{n=0}^{N-1} x(n) e^{-j2\pi k'n/N}, \quad (4.8)$$

which is identical to an N -point FFT. Thus, in this example, zero padding does not change the amplitude and phase of the even-numbered components of the spectrum, but provides interpolated values at the odd-number components.

The above discussion can be generalized to pad the input data with LN of zeros, with L as a positive integer. In the above discussion $L = 1$. In general, in order to keep the total number of data points a power of 2 number, the L value is

$$L = 2^i - 1, \quad \text{where } i = 0, 1, 2, \dots \quad (4.9)$$

An example is shown in Figure 4.3. A total of 32 points of data are collected and Figure 4.3(a) shows the 32-point FFT. Figure 4.3(b) shows the result of a 64-point FFT padded with 32 zeros. One can see that the even components in this figure have the same amplitude as in Figure 4.3(a). Figure 4.3(c) shows the result of a 128-point FFT padded with 96 zeros. One can see that every fourth component is identical to the result in Figure 4.3(a). Figure 4.3(d) shows the result of a 1024-point FFT padded with 992 zeros. Therefore it is obvious that the zero padding does not change the spectrum shape, but only interpolates between the original N -point FFT. In these figures, the frequency range is from 0 to $f_s/2$, where f_s is the sampling frequency. If the input data are complex, the frequency is from 0 to f_s . The frequency index is based on the number of the FFT points.

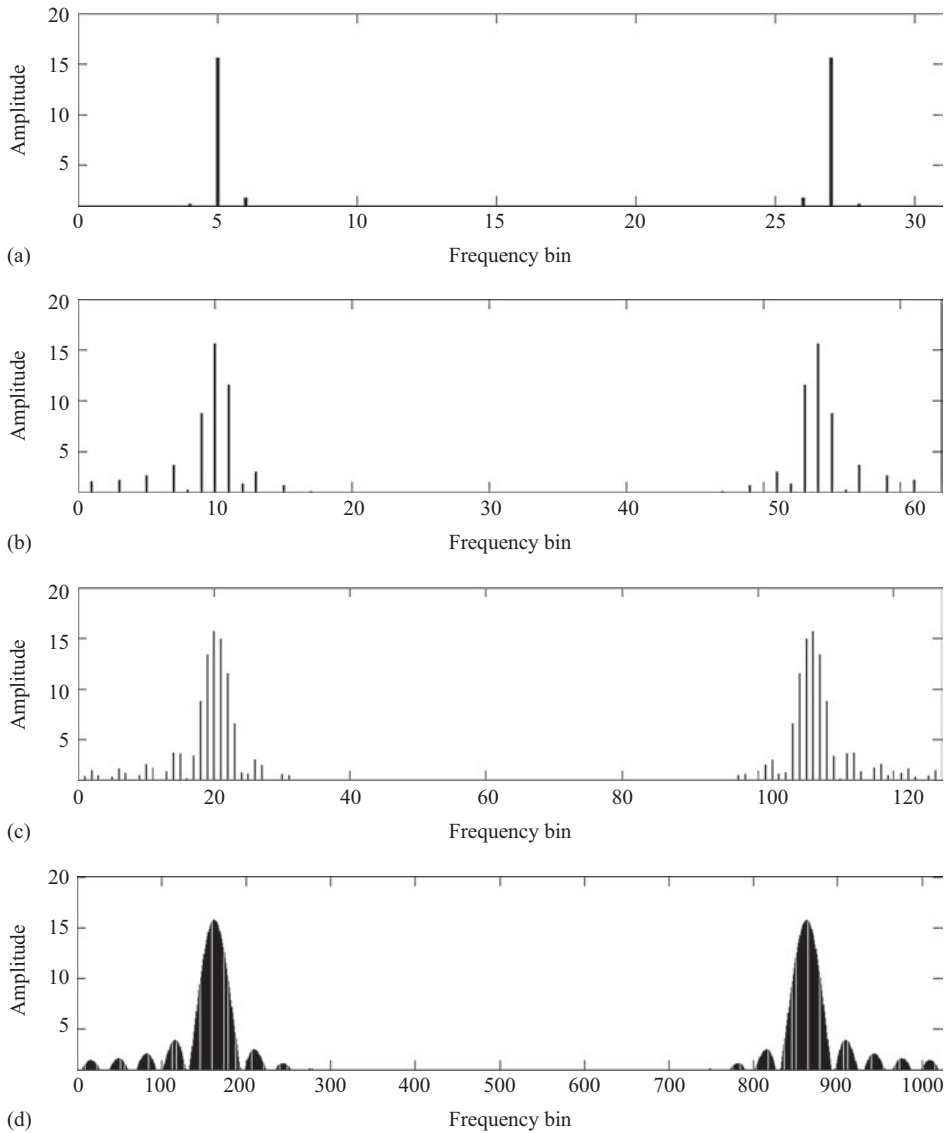


Figure 4.3 Effect of zero padding: (a) 32-point FFT, (b) 64-point FFT, (c) 128-point FFT, and (d) 1024-point FFT.

From this example, one can see that in the 32-point FFT it might be difficult to choose the peak of the power spectrum. The fine structure of the sidelobes is even more difficult to observe. With zero padding, the location of the peak of the power spectrum appears more clearly and the sidelobes are clearly shown. Thus the additional processing provided by zero padding can improve the capability of interpolating the peak of the power spectrum. The price paid is increased processing time. If zero padding is not used, the fine structure in the frequency domain cannot be obtained. Many of the plots generated in this book use zero padding.

A very important point is that zero padding does not increase the fundamental resolution of the FFT. In other words, the width of the mainlobe of the FFT does not change due to zero padding. The frequency resolution depends only on the true data length. Zero padding can only improve the capability of selecting the peak frequency component in the mainlobe.

4.4 Peak Position Estimation for Different Windows

This section discusses a simple way to extrapolate the peak of the frequency component from FFT outputs. The zero padding operation discussed in Section 4.3 can be used to better locate the peak, but the length of the FFT must be increased. In this method, the size of the FFT can be maintained and the peak can be estimated using two points from the FFT output.

4.4.1 Peak Position Estimation for Rectangular Window [10–16]

First, let us consider the continuous case and assume a rectangular window. This case was discussed in Section 3.3 and the results are presented here again. If the input is a rectangular window in the time domain with unity amplitude and width T as shown in Figure 4.4(a), its Fourier transform is a sinc function in the frequency domain (from Section 3.3) as

$$X(f) = \frac{\sin(\pi f T)}{\pi f}. \quad (4.10)$$

The result is shown in Figure 4.4(b); its peak occurs at $f = 0$ and the first minima are at $\pm 1/T$.

However, if its spectrum is calculated via the FFT, the frequency components are discrete and will follow the contour of the sinc function as shown in Figure 4.5. In this figure the maximum frequency component of the power spectrum might not coincide with the maximum of the sinc function. The purpose of this discussion is to determine the true position of the maximum from these FFT components. It should be noted that there are equal separations between these components and the distance equals $1/T$. Let us represent the amplitude of the frequency component by X_i ; the highest one by X_0 , the second highest one by X_1 , and so on. From Figure 4.5, one can see that two of the highest amplitudes are in the mainlobe and the third one is in the first sidelobe.

Assume that k is the distance between the true peak and X_0 . Since the distance between adjacent frequency components is $1/T$, this quantity can be used as the unit of distance. The positions of X_1 and X_2 are at $k - 1/T$ and $k + 1/T$, respectively. Their corresponding amplitudes can be written as

$$\begin{aligned} X_0 &= \frac{\sin(\pi T k)}{\pi k} \\ X_1 &= \frac{\sin[\pi T(k - \frac{1}{T})]}{\pi(k - \frac{1}{T})} = \frac{-\sin(\pi T k)}{\pi(k - \frac{1}{T})} \\ X_2 &= \frac{\sin[\pi T(k + \frac{1}{T})]}{\pi(k + \frac{1}{T})} = \frac{\sin(\pi k)}{\pi(k + \frac{1}{T})}. \end{aligned} \quad (4.11)$$

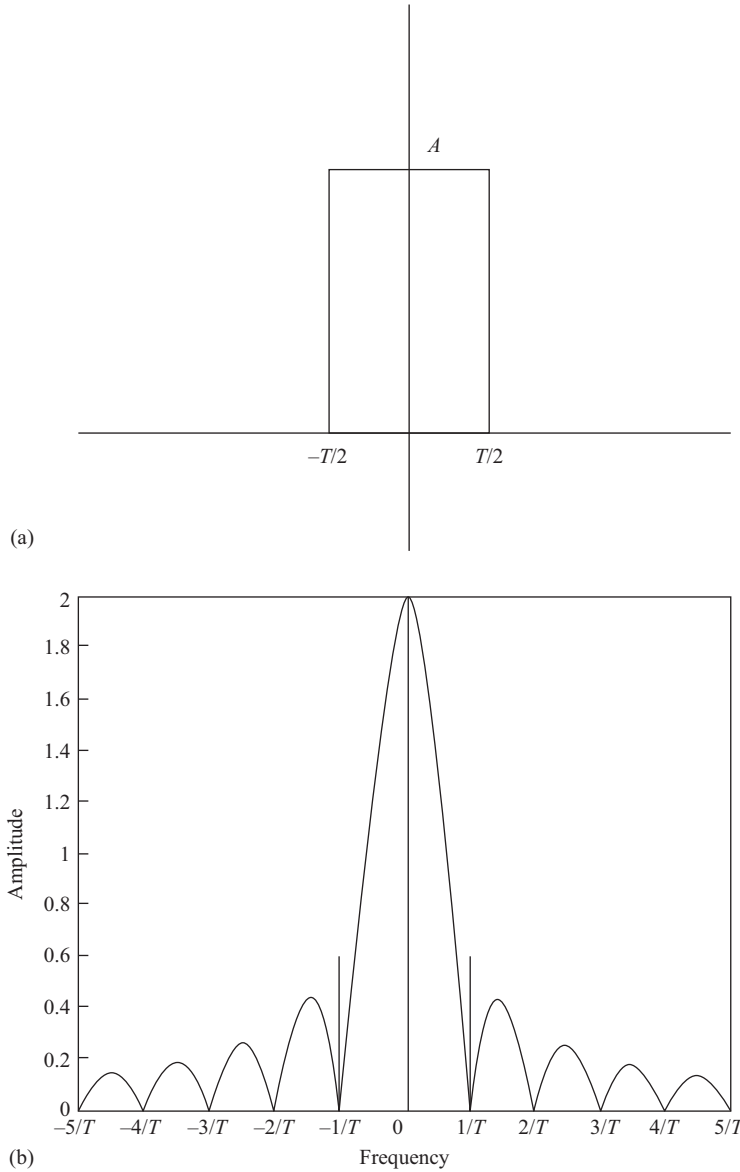


Figure 4.4 A rectangular function: (a) time domain and (b) frequency domain.

From these equations one can find

$$\frac{X_1}{X_0} = \frac{k}{1-k} \quad \text{or} \quad k = \frac{X_1}{X_0 + X_1}. \quad (4.12)$$

In (4.12), the relation $1/T = 1$ is used because the window width is assumed to be unity. Similarly, from X_1 and X_2 one can obtain

$$\frac{X_2}{X_1} = \frac{1-k}{1+k} \quad \text{or} \quad k = \frac{X_1 - X_2}{X_1 + X_2}. \quad (4.13)$$

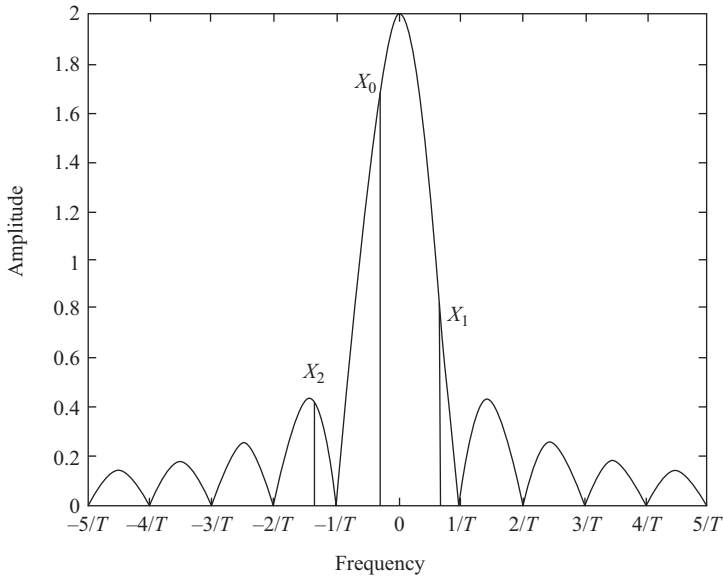


Figure 4.5 Frequency components from FFT of a rectangular window.

Either (4.12) or (4.13) can be used to find the center of the spectrum, and each equation only uses two output components from the frequency domain. If the signal is noisy, both equations can be used to obtain k and the averaged value of the two can be used as the value of k .

If the frequency at X_0 and X_1 are k_0 and k_1 , respectively, once k is found, the center frequency f_c can be considered as

$$f_c = k_0 \pm k. \quad (4.14)$$

The positive sign is used if $k_1 > k_0$ and the negative sign is used if $k_1 < k_0$. Note that the center frequency f_c is no longer an integer number.

The one important operation to obtain the above relations is the capability to eliminate the sine function by taking the ratio of the results in (4.11). Once the sine function is eliminated, the solution of k is easily found.

4.4.2 Peak Positions Estimation for Hanning Windows [10–16]

If a certain window is applied to the input data, the above approach can still be applicable. Let us use a Hanning window to demonstrate this idea. The Hanning window in the time domain can be written as

$$w(t) = \frac{1}{2} + \frac{1}{2} \cos\left(\frac{2\pi t}{T}\right), \quad (4.15)$$

where T is the total data length in time. The frequency response is shown in Figure 4.6.

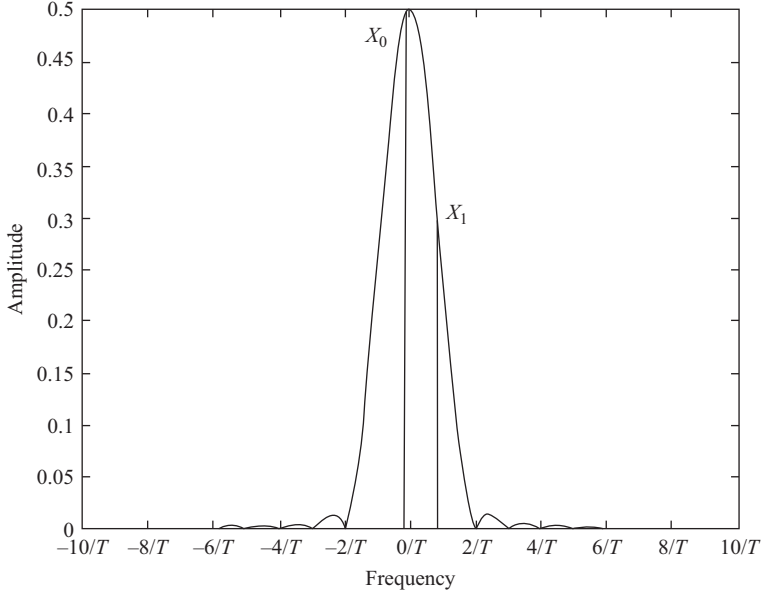


Figure 4.6 Frequency components from FFT of a Hanning window.

The frequency response can be obtained from the Fourier transform of the constant and the cosine terms, which are discussed in Chapter 3. If the highest amplitude is k from the true maximum where $f = 0$, the amplitude can be written from (3.69) as

$$\begin{aligned}
 X_0 &= \frac{\sin(\pi Tk)}{2\pi k} + \frac{1}{4} \left\{ \frac{\sin[\pi T(k + \frac{1}{T})]}{\pi(k + \frac{1}{T})} + \frac{\sin[\pi T(k - \frac{1}{T})]}{\pi(k - \frac{1}{T})} \right\} \\
 &= \sin(\pi Tk) \left\{ \frac{1}{2\pi k} - \frac{1}{4\pi(\frac{1}{k+T})} - \frac{1}{4\pi(k - \frac{1}{T})} \right\}. \tag{4.16}
 \end{aligned}$$

The second highest amplitude is $1/T$ away from k and can be written as

$$\begin{aligned}
 X_1 &= \frac{\sin[\pi T(k - \frac{1}{T})]}{2\pi(k - \frac{1}{T})} + \frac{1}{4} \left\{ \frac{\sin(\pi Tk)}{\pi k} + \frac{\sin[\pi T(k - \frac{2}{T})]}{\pi(k - \frac{2}{T})} \right\} \\
 &= \sin(\pi Tk) \left\{ \frac{-1}{2\pi(k - \frac{1}{T})} + \frac{1}{4\pi k} + \frac{1}{4\pi(k - \frac{2}{T})} \right\}. \tag{4.17}
 \end{aligned}$$

Similarly, if one finds the ratio of X_1/X_0 , the sine function can be cancelled. If one assumes that $1/T = 1$, the result is

$$\frac{X_1}{X_0} = \frac{\frac{-1}{2(k-1)} + \frac{1}{4k} + \frac{1}{4(k-2)}}{\frac{1}{2k} - \frac{1}{4(k+1)} - \frac{1}{4(k-1)}} = \frac{k+1}{2-k}$$

or

$$k = \frac{2X_1 - X_0}{X_0 + X_1}. \quad (4.18)$$

Once the k value is found, the center frequency can be estimated by (4.14).

The above relations in (4.12) and (4.18) are also included in Rife and Boorstyn [14]. In this reference, some other windows are also included.

The spectrum estimation method covered in this section is similar to zero padding. Neither method changes the power spectrum shape, they only provide a better estimation of the peak location of the power spectrum. When there are multiple signals in the mainlobe, these two methods can cause errors. The two methods mentioned in this section are sensitive to noise, especially when X_0 is very close to the true peak. Under this condition, X_1 and X_2 are very close to the minima and noise may reverse their amplitudes. If the amplitudes of X_1 and X_2 are reversed, the above equation will move the peak in the wrong direction and cause more error in the frequency reading.

4.5 Peak Position Estimation Through Iteration [16]

This approach generates X_0 and X_1 through iterations from the input data. The idea is to make $X_0 = X_1$, as shown in Figure 4.7. If one can find two frequencies with the same amplitudes close to the main peak, the center peak frequency will be between these two values. In order to keep this discussion simple, let us use an example with a rectangular window. When two frequencies are separated by $1/T$ and have the same amplitude, their amplitudes can be found as

$$\left| \frac{\sin(\pi T f)}{\pi f} \right|_{f=\pm\frac{1}{2T}} = \frac{2T}{\pi}, \quad (4.19)$$

which is $2/\pi$ below the true peak. This value is very close to the peak and will not be affected much by noise. Thus this approach is less noise sensitive, but more computation is required.

The following procedure is used to find the two frequency components with approximately equal amplitudes. In order to keep this discussion clear, all the steps (including previously derived equations) will be included. Assume the input signal is $x(n)$ and X_0 and X_1 are the highest two components in the frequency domain. From these two components one can find k from (4.20) as

$$k^{(1)} = \frac{X_1}{X_0 + X_1}. \quad (4.20)$$

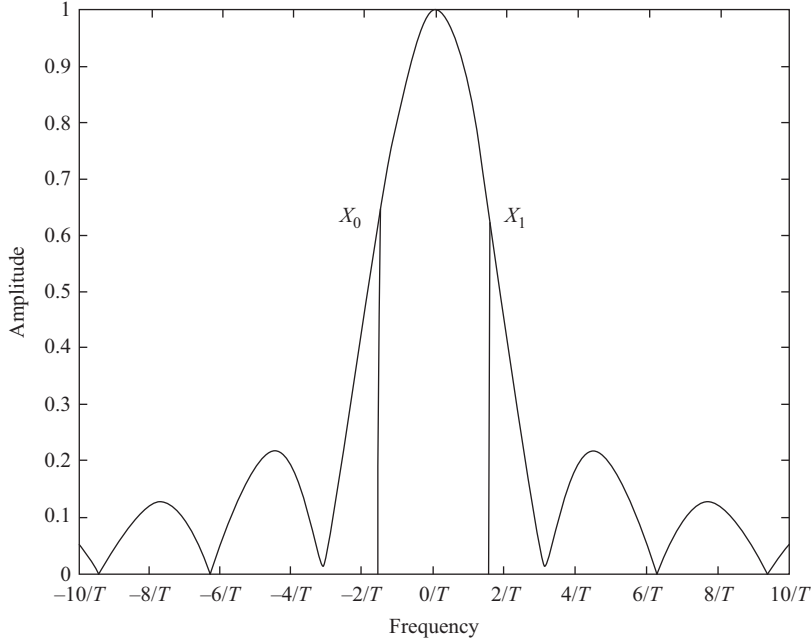


Figure 4.7 Desired relation of X_0 and X_1 .

The center frequency from (4.21) is

$$f_c^{(1)} = k_0 \pm k^{(1)}. \quad (4.21)$$

In these equations, superscripts are used to represent the order of iteration. From this frequency, one component on each side is selected. The separation between them is chosen to be $1/T$ in order to use the relation in (4.21). If the separation is smaller than $1/T$, a linear approximation may be used to find the center frequency. However, there is a possibility that the two components are on the same side of the mainlobe and make the converging process slow. The two frequencies are

$$k_0^{(1)} = f_c^{(1)} - \frac{1}{2T}$$

and

$$k_1^{(1)} = f_c^{(1)} + \frac{1}{2T}. \quad (4.22)$$

From these two frequencies, their corresponding amplitudes can be found as

$$\begin{aligned} X_0^{(1)} &= \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi n k_0^{(1)}}{N}} \\ X_1^{(1)} &= \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi n k_1^{(1)}}{N}}. \end{aligned} \quad (4.23)$$

It should be noted that these equations only calculate one frequency component at $k_0^{(1)}$ and $k_1^{(1)}$. Once $X_0^{(1)}$ and $X_1^{(1)}$ are found, they can be used to find $k^{(2)}$ from (4.20), and the same procedure repeats.

From simulated results, it appears that this process converges in about three cycles. In other words, once $X_0^{(3)}$ and $X_1^{(3)}$ are known, they can be used to find $k_0^{(4)}$ and $k_1^{(4)}$. The final frequency can be found as

$$f_c = \frac{k_0^{(4)} + k_1^{(4)}}{2}. \quad (4.24)$$

Comparing this approach with the simple approach in (4.12), it appears that this method is less sensitive to noise.

This method can be applied to other windows. The only change required is (4.20). For simplicity, sometimes a linear relation can be used.

4.6 Actual Frequency Determination by FFT [4, 5]

In this section we will find the actual frequency (in hertz) of the input signal after the FFT is performed. Parameters affecting this operation are the digitizing speed and the total number of data points. After the FFT, the frequency components should be in units of hertz. Let us start with the DFTs. They are rewritten in the following forms:

$$\begin{aligned} X(k) &= \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} \\ x(n) &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N}. \end{aligned} \quad (4.25)$$

Let the input signal be

$$x(t) = e^{j2\pi f_0 t}. \quad (4.26)$$

After digitization, the input is

$$x(n) = e^{j2\pi f_0 n t_s}, \quad (4.27)$$

where t_s is the sampling period. When this equation is substituted into (4.25), the result is

$$\begin{aligned} X(k) &= \sum_{n=0}^{N-1} e^{j2\pi f_0 n t_s} e^{-j2\pi kn/N} \\ &= \sum_{n=0}^{N-1} e^{j2\pi n(f_0 t_s - k)/N} = \frac{1 - e^{j2\pi(Tf_0 - k)}}{1 - e^{j2\pi(Tf_0 - k)/N}}, \end{aligned} \quad (4.28)$$

where $T = Nt_s$ and T is the total signal duration. It is interesting to note that the sampling interval t_s does not appear in the above equation explicitly, but only the total length of data T appears. The amplitude of the above equation can be written as

$$|X(k)| = \frac{\sin[\pi(Tf_0 - k)]}{\sin\left[\frac{2\pi(Tf_0 - k)}{N}\right]}. \quad (4.29)$$

The peak of this equation occurs at $Tf_0 = k$. However, k can only be an integer. Thus the frequency of the input signal is

$$f_0 \approx \frac{k}{T}. \quad (4.30)$$

Let us use a numerical example to illustrate the result. Assume the input is a sine wave and the simulation setting is listed below:

Signal frequency, $f_0 = 200 \text{ MHz} = 2 \times 10^8 \text{ Hz}$
 Sampling rate = 1 GHz
 Sampling period, $t_s = 10^{-9} \text{ s}$,
 FFT frame size, $N = 64$, then $T = 64 \times 10^{-9} \text{ s}$.

It is interesting to note that the length of data from $x(0)$ to $x(63)$ only covers $63t_s$; however, the input to the FFT is periodic and the period is from $x(0)$ to $x(64)$. Thus the length of data should be considered as $64t_s$. The signal can be written as

$$x(n) = \sin\left(\frac{2\pi f_0 n t_s}{N}\right). \quad (4.31)$$

The result of applying the FFT is shown in Figure 4.8. Since the input signal is real, the unambiguous band will be 500 MHz ($1/2t_s$) and only half of the spectrum components from 0 to 31 are plotted. Each frequency bin is $1/T = 15.625 \times 10^6 \text{ Hz}$ wide. The peak of the power spectrum is at $k = 13$, where the true frequency should be at $Tf_0 = 12.8$ rather than 13. The corresponding frequency can be determined as $k/T = 13 \times 15.6 \times 10^6 = 203 \times 10^6 \text{ Hz}$, which is close to the input frequency.

4.7 Autocorrelation [2–5]

In this section, the definitions of autocovariance and autocorrelation will be discussed since they can be used in spectrum estimation. The difference between autocorrelation and convolution will also be discussed.

Autocorrelation is often used in spectrum estimation. Autocorrelation is quite similar to convolution. If there are N points of input data represented by $x(n)$, where n is from 0 to $N - 1$, the autocorrelation is defined as

$$R(m) = \frac{1}{N} \sum_{n=0}^{N-m-1} x(n)x(n+m), \quad (4.32)$$

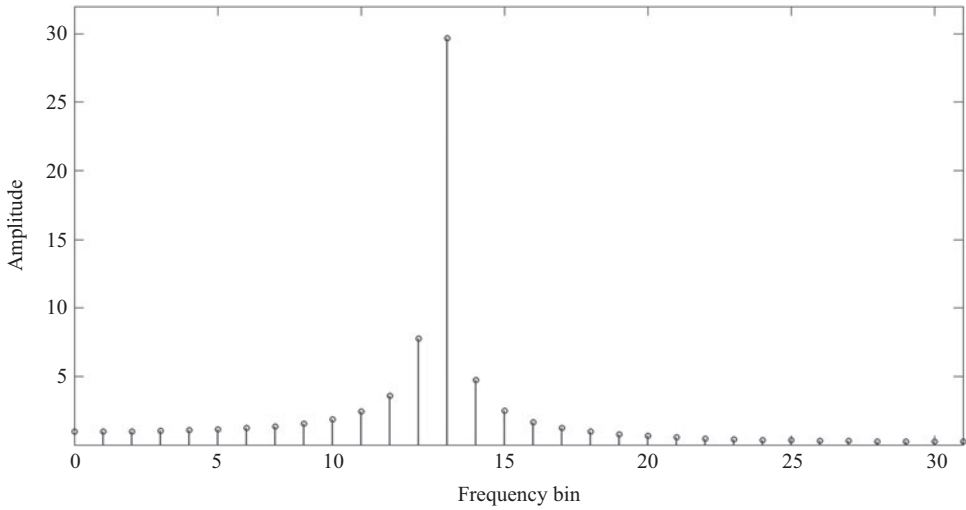


Figure 4.8 Frequency components from the FFT of a sine wave.

$$\begin{array}{ccccccc}
 x(0) & & x(1) & & x(2) & & \dots & & x(N-m-1) \\
 x(m) & & x(m+1) & & x(m+2) & & \dots & & x(N-1)
 \end{array}$$

$$NR(m) = x(0)x(m) + x(1)x(m+1) + x(2)x(m+2) + \dots + x(N-m-1)x(N-1)$$

Figure 4.9 Representation of $R(m)$ calculation.

where m is referred to as the lag of the autocorrelation. Strictly speaking, this quantity should be referred to as the “sample” autocorrelation, which approximates $E[x(n)x(n+m)]$, where $E[\]$ represents the expectation value. The value of m can be either positive or negative. When the argument m is negative, the autocorrelation can be related to the one with positive m as

$$R(-m) = R(m)^* \quad (4.33)$$

if $R(k)$ is complex. If $R(m)$ is real, then $R(-m) = R(m)$.

The autocorrelation can be explained with a simple example. If the lag is m , the input data from 0 to $N-1$ is divided into two groups of the same length: one from 0 to $N-m-1$, the other one from m to $N-1$. The elements in these two groups are multiplied term by term as shown in Figure 4.9. The sum of all the product terms is equal to N times the autocorrelation $R(m)$.

The autocorrelation can be considered as the measurement of the similarity between the two groups of data. If the two groups are alike, the autocorrelation value is high; otherwise, it is low. When $m = 0$, the two groups of data are identical; therefore $R(0)$ produces the highest value among the autocorrelations with different lags. When the m value is large, only a few terms can be obtained in the summation, but the denominator N in (4.32) is a fixed constant. Therefore the magnitude of $R(m)$ is usually small because the summation is divided

by N ; however, theoretically this value could be large. The autocorrelation defined in (4.32) is referred to as the biased form.

One can define the unbiased autocorrelation as

$$R_u(m) = \frac{1}{N-m} \sum_{n=0}^{N-m-1} x(n)x(n+m). \quad (4.34)$$

In this definition, when m is large, the denominator is small and it is equal to the number of summation. This form is seldom used in spectrum estimation because it may lead to a negative power spectrum [4]. If $R(m)$ is small, it has less effect on the calculated power spectrum. It is desired to keep the value of $R(m)$ small because it contains only a few data points, which should have less effect on the power spectrum.

Another quantity that is very close to autocorrelation is called autocovariance. Autocovariance can be written as

$$\Gamma(m) = \frac{1}{N} \sum_{n=0}^{N-m-1} [x(n) - \mu][x(n+m) - \mu], \quad (4.35)$$

where μ is the “sample” mean of the input data $x(n)$ and can be written as

$$\mu = \frac{1}{N} \sum_{n=0}^{N-1} x(n). \quad (4.36)$$

Strictly speaking, the result in (4.35) should be called “sample” autocovariance. The only difference between the autocovariance and the autocorrelation is that in the autocovariance the mean is subtracted from the input data. If the mean is zero, then these two quantities are identical.

The convolution operation discussed in Section 3.4 is mathematically similar to the autocorrelation, but they have different meanings. If an input signal $x(n)$ passes through a linear system, the output signal can be obtained from convolution. In autocorrelation the result represents the similarity of two data sets, while in convolution the result represents the output of a linear system.

The mathematical representation of convolution in digital form can be written as

$$\begin{aligned} y(m) &= h \otimes x = \sum_{n=0}^{N-m-1} h(n)x(m-n) \\ &= \sum_{n=0}^{N-m-1} h(m-n)x(n), \end{aligned} \quad (4.37)$$

where $x(n)$ is the input data, $h(n)$ is the impulse response of the linear function, and $y(m)$ is the output. The major difference between this equation and (4.32) is that there is a minus sign in either $x(n)$ or $h(n)$, but there is no minus sign in (4.32). Therefore the autocorrelation can be calculated as a convolution of a signal and its flipped version, and the convolution can be implemented using the FFT operation.

4.8 Autocorrelation (Blackman–Tukey) Spectrum Estimation [2–5, 17]

In this section, autocorrelation is used to find the power spectrum of the input signals. The power spectrum can be estimated using two approaches. First, it can be estimated through the FFT of the input data in the time domain and then squaring the result. This approach is called the periodogram, and is discussed in Section 4.2. This result can be obtained directly from the FFT operation. The power spectrum can also be obtained from the autocorrelation of the input data. It can be shown that [17]

$$\lim_{T \rightarrow \infty} \frac{1}{T} E \left[\left| \int_{-T/2}^{T/2} x(t) e^{-j2\pi f t} dt \right|^2 \right] = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} R(\tau) e^{-j2\pi f \tau} d\tau, \quad (4.38)$$

where $E[\]$ represents the expectation value.

The power spectrum $P(k)$ from limited data can be obtained from the above equation as

$$P(k) = \sum_{m=-M}^M R(m) e^{-j2\pi k m t_s}, \quad (4.39)$$

where the m value is from $-M$ to M , k is the frequency component, and t_s is the unit of sampling period. This equation is usually referred to as the Blackman–Tukey method. The total number in the summation is $2M + 1$ since it includes $m = 0$. If the biased autocorrelation is used in the above equation, the result is equal to the periodogram, as discussed in Kay [4].

In general, the biased autocorrelation $R(m)$ is used in the above equation in order to produce a positive power spectrum. The biased $R(m)$ can be considered as a windowed function. When m is large, $R(m)$ is usually low. Sometimes the summation of m is limited to approximately $m = -N/10$ to $N/10$, and a maximum value of $-N/5$ to $N/5$ is often recommended. However, a window function can also be added to the biased autocorrelation to further reduce the sidelobes. With an additional window, the power spectrum can be written as

$$P(k) = \sum_{m=-M}^M W(m) R(m) e^{-j2\pi k m}, \quad (4.40)$$

where $W(m)$ is the window function. This window must be symmetrical in order to produce an even function of $P(k)$.

The expected value of $R(m)$ can be found as follows. Since $R(m) = R(-m)$, $R(m)$ can be written as

$$\begin{aligned} R(m) &= \frac{1}{N} \sum_{n=0}^{N-m-1} x(n) x(n+m) \\ &= \begin{cases} \frac{1}{N} \sum_{n=0}^{N-|m|-1} x(n) x(n+m) & \text{for } |m| \leq N-1, \\ 0 & 0 \text{ otherwise} \end{cases} \end{aligned} \quad (4.41)$$

The expectation value of $R(m)$ can be written as

$$\begin{aligned} E[R(m)] &= \frac{1}{N} \sum_{n=0}^{N-|m|-1} E[x(n)x(n+m)] = \frac{1}{N} \sum_{n=0}^{N-|m|-1} R(m) \\ &= \frac{R(m)}{N} \sum_{n=0}^{N-|m|-1} 1 \end{aligned} \quad (4.42)$$

since $R(m)$ is independent of n . The summation equals $N - |m|$, thus the above equation can be written as

$$E[R(m)] = \frac{N - |m|}{N} R(m) \xrightarrow{\text{as } N \rightarrow \infty} R(m). \quad (4.43)$$

The term $(N - |m|)/N$ represents a triangular window. Therefore one can argue that the Blackman–Tukey method is inherently limited by an effective window and the value of $R(m)$ with $m > N$ is arbitrarily assumed to be zero.

From the DFT and periodogram discussed previously, the frequency resolution equals f_s/N , where f_s is the sampling frequency and N is the total number of data points. If two frequencies f_{i1} and f_{i2} are separated by less than the frequency resolution cell (i.e., $|f_{i1} - f_{i2}| \leq f_s/N$), the periodogram and the Blackman–Tukey methods cannot distinguish them. Under this condition, zero padding will not help either. The only way to distinguish them through these two methods is to increase the actual data duration ($T = Nt_s$).

4.9 Application of FFT to Spectrum Estimation from the Autocorrelation Function [2]

In this section we will use the FFT to calculate the result from (4.39). Equation (4.39) is similar to the DFT, but the k value can be arbitrarily assigned. For example, any given k value can be calculated from the above equation. If k is restricted to exactly match the DFT, the FFT algorithm can be used to save calculation time. In order to adopt the FFT, (4.39) must be changed into the proper form. A DFT can be written as

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}, \quad (4.44)$$

where $X(k)$ is the frequency domain response and $x(n)$ is the time domain sampled data point. In this equation, both n and k are discrete and N is usually a power of 2 number.

The following procedures are needed to change (4.39) into the proper form.

1. The notation of t_s must be changed. By comparing (4.39) and (4.44), one can determine that

$$t_s = \frac{1}{N}, \quad (4.45)$$

where N is a number of the power of 2. Comparing with (4.28), this implies that T is assumed to be unity.

2. Equation (4.39) has to sum $2M + 1$ terms, which cannot be a power of 2 number. In order to use the FFT, the total terms under the summation sign must be changed to a power of 2 number. To accomplish this, (4.39) must be padded with zeros.

3. Equation (4.39) sums values from $-M$ to M , but in (4.44), the summation starts from zero, therefore the summation in (4.39) must be rearranged to start from zero.

In order to accomplish step 1, choose the N value as

$$2M + 1 < N < 2(2M + 1), \quad (4.46)$$

but N must be a power of 2 number.

Separate the summation of (4.39) as

$$\begin{aligned} P(k) &= \sum_{m=-M}^M R(m) e^{\frac{-j2\pi mk}{N}} \\ &= \sum_{m=0}^M R(m) e^{\frac{-j2\pi mk}{N}} + \sum_{m=-M}^{-1} R(m) e^{\frac{-j2\pi mk}{N}}. \end{aligned} \quad (4.47)$$

On the second summation, make the change of variable

$$m = m' + N, \quad (4.48)$$

then (4.47) can be written as

$$\begin{aligned} P(k) &= \sum_{m=0}^M R(m) e^{\frac{-j2\pi km}{N}} + \sum_{m'=N-M}^{N-1} R(m' + N) e^{\frac{-j2\pi k(m' + N)}{N}} \\ &= \sum_{m=0}^M R(m) e^{\frac{-j2\pi km}{N}} + \sum_{m'=N-M}^{N-1} R(m' + N) e^{\frac{-j2\pi km'}{N}}. \end{aligned} \quad (4.49)$$

In the above equation, the relation is

$$e^{-j2\pi k} = 1, \quad (4.50)$$

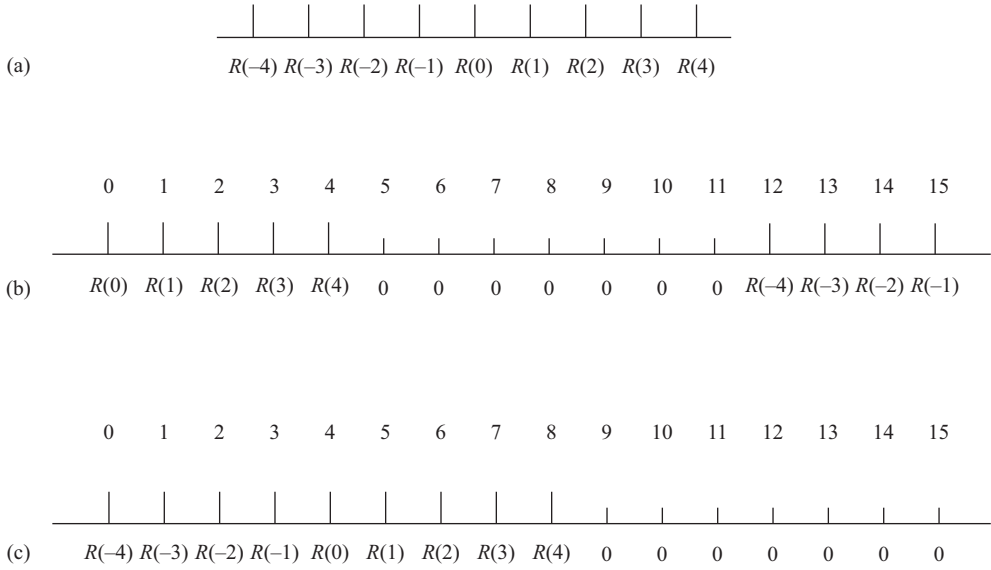
where k is an integer.

Since both m and m' are dummy variables, they can be replaced by n . The result can be written as

$$P(k) = \sum_{n=0}^M R(n) e^{\frac{-j2\pi kn}{N}} + \sum_{n=M+1}^{N-M-1} 0 + \sum_{n=N-M}^{N-1} R(n + N) e^{\frac{-j2\pi kn}{N}}. \quad (4.51)$$

In this equation, two important facts should be noted. First, zeros are added from $n = M + 1$ to $N - M - 1$. The total terms on the right equal N , a number that is the power of 2. Second, the $R(n)$ must be properly arranged. The $R(n)$ for $n = 0$ to M stay the same. Under the third summation sign, the $R(n)$ should be changed from the range of $n = -M$ to -1 in (4.47) to the range from $n = 2N - M$ to $2N - 1$, thus the above equation can be written as

$$\begin{aligned} P(k) &= \sum_{n=0}^M R(n) e^{\frac{-j2\pi kn}{N}} + \sum_{n=M+1}^{N-M-1} 0 + \sum_{n=N-M}^{N-1} R(n + N) e^{\frac{-j2\pi kn}{N}} \\ &= \sum_{n=0}^{N-1} R(n) e^{\frac{-j2\pi kn}{N}}, \end{aligned} \quad (4.52)$$

Figure 4.10 Rearranged $R(n)$.

which is exactly the same form as (4.44). Therefore FFT can be used to perform the calculation.

Now, let us use an example to demonstrate this operation. Let $R(n)$ be from -4 to 4 ($M = 4$), as shown in Figure 4.10(a). There are total of nine terms. From (4.46), choose N between 9 and 18 (and N should be a number that is the power of 2), thus $N = 16$. The rearranged $R(n)$ is shown in Figure 4.10(b). After this rearrangement, FFT can be used to calculate $P(k)$.

In the above discussion, (4.39) is directly transformed into the form of (4.44), therefore the correct phase relation is kept. The result obtained for $P(k)$ is usually a complex value. If the power spectrum is desired, the absolute value will be used. The $R(n)$ can be rearranged slightly differently and the same results will be obtained. This approach is to shift the entire $R(n)$ from $n = -M$ to M to $n = 0$ to $2M$ and pad with zeros at the end of the data string as shown in Figure 4.10(c). This arrangement will not generate the same result as (4.44) and there is a phase difference. However, the absolute value of $P(k)$ is the same, which is important for power spectrum estimation. This new way of shifting $R(n)$ might be somewhat easier to accomplish.

4.10 Basic Idea of a Sub-Nyquist Sampling Scheme [18–24]

In the next three sections, the concept of using sub-Nyquist sampling to design an EW receiver will be discussed. This section discusses the basic idea. Section 4.11 discusses the difference in phase relation between an analog and a digital system, as this is important in the sub-Nyquist sampling method. In Section 4.12, some potential problems and solutions that are unique in the sub-Nyquist approach are presented.

The sub-Nyquist method can have two impacts that might be of interest to EW receiver designers. First, this method can increase the instantaneous bandwidth of the receiver. Second, for a given bandwidth, the FFT processing speed need not match the instantaneous bandwidth. Actually these two impacts are closely related. The basic sub-Nyquist sampling scheme is very similar to the instantaneous frequency measurement (IFM) receiver briefly discussed in Chapter 2.

The main part of an IFM receiver is the correlator. The input to the correlator consists of a delayed and an undelayed copy of the same signal. The outputs from the correlator are

$$E = \sin(2\pi f\tau)$$

and

$$F = \cos(2\pi f\tau), \quad (4.53)$$

where f is the frequency of the input signal and τ is the delay time of the delayed path. In the above equation, the amplitude of the signal is neglected for simplicity. Since the delay time is known, the frequency can be obtained from

$$\theta = \tan^{-1}\left(\frac{E}{F}\right) = 2\pi f\tau. \quad (4.54)$$

The only restriction in the above equation is that $\theta < 2\pi$. If $\theta > 2\pi$, there is an ambiguity problem. The maximum unambiguous bandwidth obtained from this equation is

$$\Delta B = \frac{2\pi}{2\pi\tau} = \frac{1}{\tau}. \quad (4.55)$$

This relation only limits the bandwidth, not the center frequency, therefore the center frequency can be any value. For example, if $\tau = 0.5$ ns, the unambiguous bandwidth is 2 GHz. The frequency range can be either from 0 to 2 GHz or from 2 to 4 GHz, or any other values as long as the bandwidth is 2 GHz.

The sub-Nyquist sampling scheme can be considered as an IFM receiver implemented through digital techniques. An important difference is that a conventional IFM receiver can process only one signal at a time, but a digital IFM receiver can process simultaneous signals because of the FFT operation.

In the sub-Nyquist approach discussed in this chapter, the input signal is divided into two paths—an undelayed one and a delayed one—and analog-to-digital converters (ADCs) are used to digitize the signal, as shown in Figure 4.11. In actual design, the delay can be introduced in the clock pulse rather than in the radio frequency (RF) circuit.

The digitized outputs can be processed through an FFT operation. The FFT will generate real and imaginary parts in the frequency domain. Let $X_{ru}(k)$ and $X_{iu}(k)$ represent the real and imaginary components for the undelayed case and $X_{rd}(k)$ and $X_{id}(k)$ represent the real and imaginary components for the delayed case. The amplitude of the FFT output can be calculated from the undelayed output as

$$X_u(k) = \left[X_{ru}(k)^2 + X_{iu}(k)^2 \right]^{\frac{1}{2}}. \quad (4.56)$$

The delayed path has the same amplitude components. Let $X_u(k_m)$ represent the maximum amplitude of the frequency component from the undelayed path, thus $X_u(k_m)$ can represent

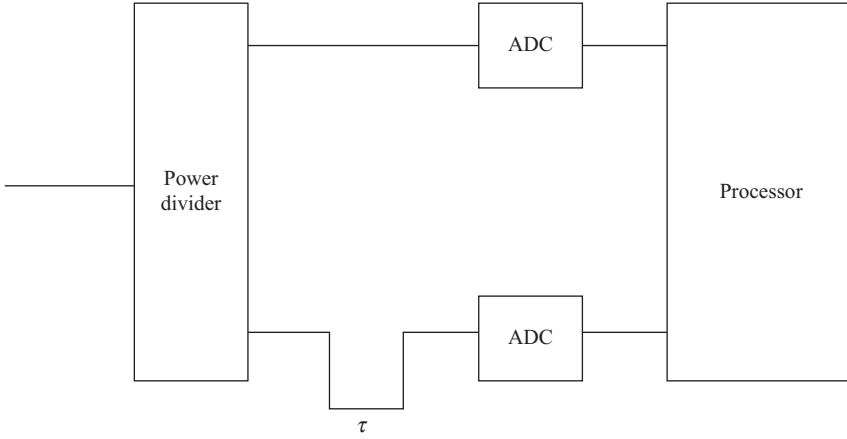


Figure 4.11 Basic sub-Nyquist sampling.

the input frequency. It should be noted that, as in the FFT operation, the input frequency need not be exactly on any frequency bin. The phase difference between the delayed and undelayed path can be written as

$$\theta = \theta_d - \theta_u = 2\pi f\tau, \quad (4.57)$$

where

$$\begin{aligned} \theta_d &= \tan^{-1} \left[\frac{X_{id}(k_m)}{X_{rd}(k_m)} \right] \\ \theta_u &= \tan^{-1} \left[\frac{X_{iu}(k_m)}{X_{ru}(k_m)} \right]. \end{aligned} \quad (4.58)$$

From this phase difference θ , the frequency of the input signal can be obtained since τ is known.

As long as the input frequencies (or multiple FFT peaks) are sufficiently separated, the input frequencies can be identified by observing those frequency bins whose magnitude exceeds a threshold. This is why this approach can process simultaneous signals.

In general, if the signal is sampled below the Nyquist rate, there is ambiguity in the FFT output. This ambiguity can be resolved by the phase difference in (4.57). Although this approach is called sub-Nyquist because the uniform sampling rate is lower than the Nyquist rate, it does not violate the Nyquist sampling theorem. The closest sampling time is the delay time τ . When the input bandwidth is less than $1/\tau$, the Nyquist sampling criterion is not violated.

Let us use an example to illustrate this approach. If an ADC can only operate at 250 MHz, the maximum unambiguous bandwidth is 125 MHz. If the desired input bandwidth is 1000 MHz and this ADC is used to collect data, the entire input bandwidth will fold into a 125-MHz output band as shown in Figure 4.12. In this figure, $f_s = 250$ MHz. This means there are eight ambiguity zones. The FFT can only determine the input frequency within 125 MHz without ambiguity. If the delay line τ is chosen to be less than 1 ns

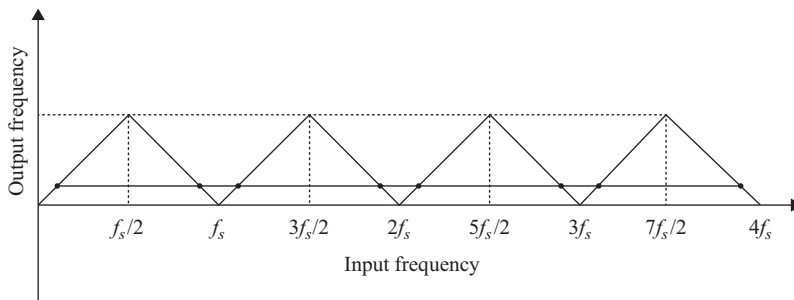


Figure 4.12 Input band versus output band.

(say 0.8 ns) from (4.55), the corresponding unambiguous frequency band is over 1250 MHz. Thus any frequency within the bandwidth of 1000 MHz can be determined uniquely by the phase difference in (4.57). For example, if the measured frequency is 40 MHz from the FFT operation, one does not know the input frequency because it can be in any one of the eight zones. The input could be 40, 210 ($250 - 40$), 290 ($250 + 40$), 460 ($500 - 40$), 540 ($500 + 40$), 710 ($750 - 40$), 790 ($750 + 40$), or 960 ($1000 - 40$) MHz as shown in Figure 4.12. The frequency obtained from the phase difference at this peak value (close to 40 MHz) can determine which zone this signal is in. From the example, if the frequency measured by the phase difference is close to 460 MHz, the true input frequency will be 460 MHz. Thus the fine frequency resolution can be determined by the FFT and the phase difference can be used to resolve the ambiguity zone.

From the above discussion, it may appear that (4.56) can be used to eliminate ambiguity frequency problems. If the phase is obtained from a continuous system, such as in an analog IFM receiver, this is true. However, the phase is not continuous in the sampled case. As a result, when the input frequency is close to the multiple of $f_s/2$, the phase measurement could be erroneous. Therefore there are regions where the frequency of the input signal cannot be obtained. In order to eliminate these regions, additional hardware is required.

4.11 Phase Relation in a Sub-Nyquist Sampling System [18, 19]

In this section, the phase difference in analog and digital systems is compared. The phase difference in a digital system will cause some receiver design changes from a conventional analog system.

In a conventional IFM receiver, the phase relation is obtained from the in-phase (I) and quadrature (Q) channel outputs. The phase versus frequency is continuous, thus the phase difference between the delayed and undelayed phase is continuous. The phase versus frequency of an IFM receiver is shown in Figure 4.13, where f_1 and f_2 are the low and high frequencies, respectively.

The difference phase will increase monotonically, except that an abrupt 2π phase change (from 2π drops to 0) will occur at a certain frequency. For simplicity, let us limit the phase shift within 2π as shown in Figure 4.13.

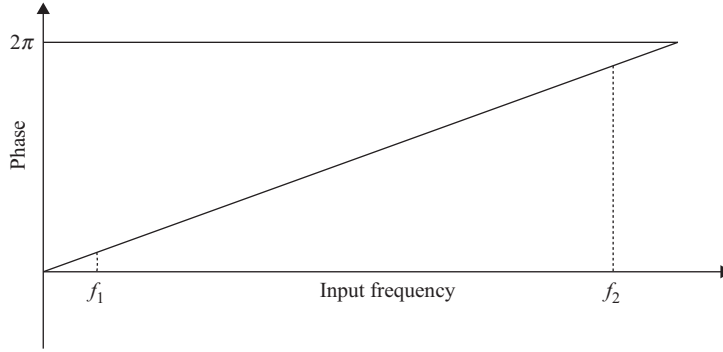


Figure 4.13 Phase versus frequency from an IFM correlator.

This idea is further illustrated mathematically by using a sine function. In the analog case, a sinusoidal wave can be written as

$$s(t) = \cos[2\pi f_o(t - \tau) - \phi], \quad (4.59)$$

where f_o is the input frequency, τ is a time delay, and ϕ is the initial phase angle. As discussed in Chapter 3, the Fourier transform of this signal can be represented by a pair of δ functions at $+f_o$ and $-f_o$ with positive and negative phase shifts as

$$S(f) = \frac{1}{2} [\delta(f - f_o)e^{j\theta} + \delta(f + f_o)e^{-j\theta}], \quad (4.60)$$

where

$$\theta = \phi + 2\pi f_o \tau. \quad (4.61)$$

Note that in the above equation, the phase for the two δ functions has opposite signs.

If the signal is truncated in time using a window function $w(t)$, the δ functions are convolved with the function $W(f)$ and the transform becomes

$$S(f) = \frac{1}{2} [W(f - f_o)e^{j\theta} + W(f + f_o)e^{-j\theta}]. \quad (4.62)$$

The window function $W(f)$ may modify the phase of the input signal; however, the phase of the positive and negative spectra still has the same amplitude but opposite signs.

In a digital system, the sine signal is sampled at frequency f_s and the Fourier integral is replaced by the DFT. The output $S_p(k)$ in the frequency domain is periodically replicated as

$$S_p(k) = S(k) \otimes \sum_k \delta(k - nf_s), \quad (4.63)$$

where k is an integer. An infinite number of zones appear, each of width f_s and containing a replica (or alias) of either the positive or negative power spectrum. If one considers the positive and negative frequencies separately, the zone width becomes $f_s/2$.

If the signal in a zone from nf_s to $(n + 1/2)f_s$ is the positive frequency, the signal in an adjacent zone from $(n + 1/2)f_s$ to $(N + 1)f_s$ is the negative frequency. Thus their phases are

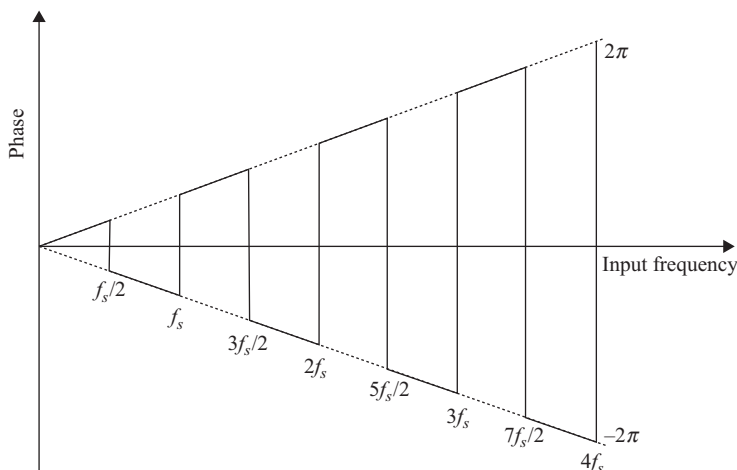


Figure 4.14 Phase versus frequency of a sub-Nyquist sampling scheme.

opposite in sign. As the input frequency increases to pass a zone boundary, the phase changes sign. The phase versus frequency of the sampled data is shown in Figure 4.14. At every multiple of $f_s/2$, there is a phase reversal.

If the phase in Figure 4.14 covers the entire range from -2π to 2π , the phase measurement can be ambiguous. The explanation is as follows. Since the real and the imaginary parts are calculated from FFT, the θ angle calculated from (4.57) is within $-\pi$ to π . The phase difference calculated from this relation has two possible values.

Let us use two examples to demonstrate the calculation of phase shifts. First, let $\theta_d = \pi/6$ and $\theta_u = 3\pi/4$. The phase difference can be either $\theta_1 = \theta_d - \theta_u = -7\pi/12$ or $\theta'_1 = \theta_d - \theta_u = (\pi/6 + 2\pi) - 3\pi/4 = 17\pi/12$ since both the calculated angles are within $\pm 2\pi$. Second, if $\theta_d = \pi/6$ and $\theta_u = -3\pi/4$, the phase difference can be either $\theta_1 = \theta_d - \theta_u = 11\pi/12$ or $\theta'_1 = \theta_d - \theta_u = \pi/6 - (-3\pi/4 + 2\pi) = -13\pi/12$. In each example, there are two different phase angles and one cannot determine which one is the true phase difference.

From these examples, it is obvious that the two possible phase shifts have the following relations:

$$|\theta_1| + |\theta'_1| = 2\pi. \quad (4.64)$$

Both θ_1 and θ'_1 have an absolute value of less than 2π , but they can be either positive or negative.

If one desires to use the entire phase shift from -2π to 2π for a digital receiver design, there are ambiguous frequency ranges. There are two ways to eliminate these ambiguous range pairs. One is to limit the input bandwidth from $\theta = -\pi$ to π , and this approach is obvious. For instance, in the previous example, to cover a 1000-MHz bandwidth by using a 250-MHz ADC, one can choose the delay time to be 0.4 ns rather than 0.8 ns. With this short delay, the overall phase difference can be restricted within $\pm\pi$ and there is no ambiguity.

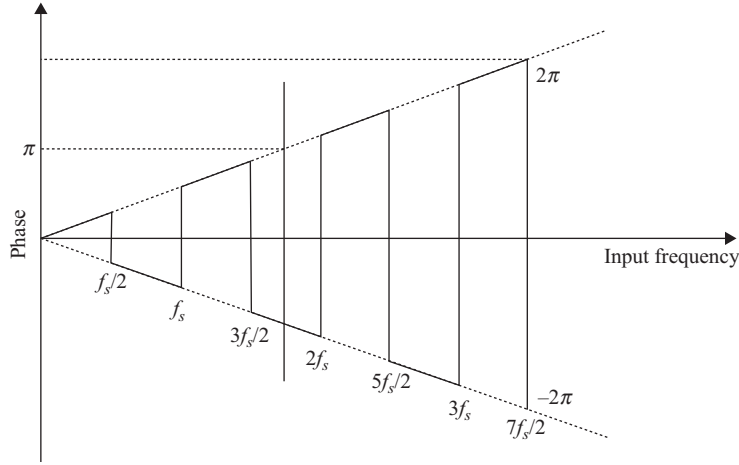


Figure 4.15 τ is chosen to have an odd number of $f_s/2$.

The other approach is to restrict the choice of the sampling frequency. One can choose the delay time with the following relation:

$$\left(n + \frac{1}{2}\right)f_s\tau = 1, \quad (4.65)$$

where n is an integer. This approach divides the input bandwidth ($1/\tau$) into an odd number of sampling bandwidth ($f_s/2$) intervals. Under this condition the phase angles θ_1 and θ'_1 have the same sign with respect to the vertical line passing π , as shown in Figure 4.15. If both phases have the same sign, the condition in (4.64) can no longer be fulfilled. For example, in the range 0 to $f_s/2$, a small positive-valued θ is the correct angle. The corresponding θ' ($\theta - 2\pi$, a negative value) is in the range $3f_s$ to $7f_s/2$ because of the symmetry. However, θ' must have a positive value in this frequency range, as shown in Figure 4.15. Thus θ' is not an acceptable answer and θ is the only answer. Therefore there is no ambiguous range if the relation in (4.65) is used.

4.12 Problems and Potential Solutions in the Sub-Nyquist Sampling Scheme

If the signal frequency is close to the multiple of $f_s/2$, the subsampling approach fails. The reason and some possible solutions will be discussed in this section.

As discussed in the previous section, if the input is real, the spectrum will appear in pairs: one at f_i , the other at $f_s - f_i$. Because of the limited data length in the time domain, the spectra have sidelobes. If the input signal frequency is close to $nf_s/2$, both spectra are close to $nf_s/2$ as shown in Figure 4.16. The sidelobe of one spectrum will interfere with the other one. This interference will distort the phase measurement. The wrong phase information may cause frequency measured to be assigned to the wrong subband, and a catastrophic error

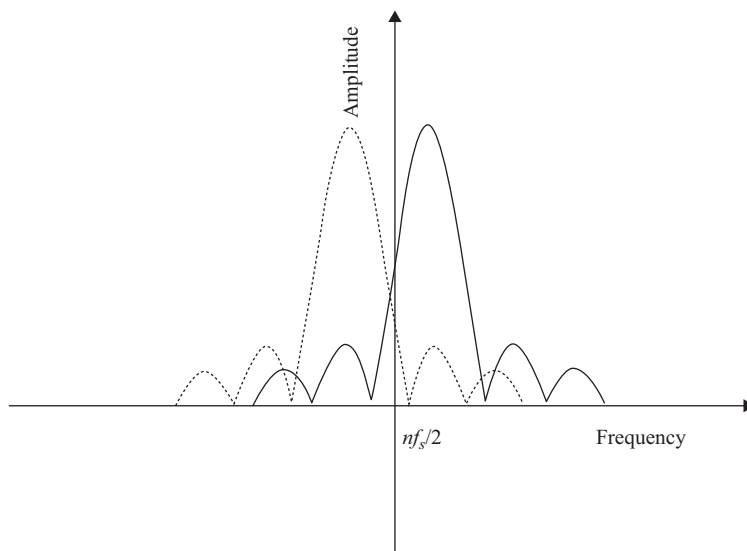


Figure 4.16 Spectra close to $nf_s/2$.

in the frequency reading would result. There are three possible ways to remedy this problem, as follows.

1. One straightforward approach is to reduce the amplitude of the sidelobes. As mentioned in the previous chapter, a proper windowing function will increase the width of the mainlobe but reduce the sidelobes. When the sidelobes are low, the interference from neighboring zones will be reduced. Although windowing can narrow the regions of interference, it cannot eliminate them. A receiver using this approach will have “holes” near the multiple of $nf_s/2$.
2. In order to reduce these “holes,” a second channel can be built. The second channel has a sampling rate f'_s , which is relatively prime to the first sampling rate f_s . The arrangement is shown in Figure 4.17. When the input signal is near the multiple of $f_s/2$, it should be far away from the multiple of $f'_s/2$. Thus when the first channel generates wrong frequency information, the second channel will produce the correct information. When the signal is close to a multiple of $f'_s/2$, the first channel should be used to read the frequency. In other words, between the two channels, one of the frequency readings must be correct.
3. Another way to eliminate the “holes” in the sub-Nyquist sampling scheme is to use the I and Q channels. The I and Q channels are used in both the delayed and undelayed paths, as shown in Figure 4.18. All four ADCs operate at the same sampling speed f_s . In both Figures 4.17 and 4.18, the delay is introduced in the clock rather than in the RF chain.

This arrangement generates a complex signal. As mentioned in Chapter 3, when the input signal is complex, the frequency component exists only on one side of the frequency axis—either on the positive or on the negative side. Let us assume that the spectrum is positive. Under this condition, when the spectrum is at the multiple number of $f_s/2$, there is no negative component to interfere with it. The phase relation measured through this approach is always correct and there is no frequency error at multiples of $f_s/2$.

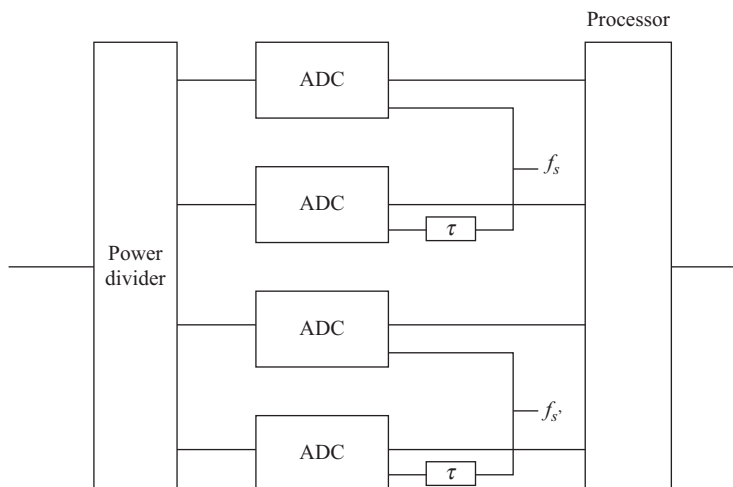


Figure 4.17 Sub-Nyquist scheme with two sampling rates.

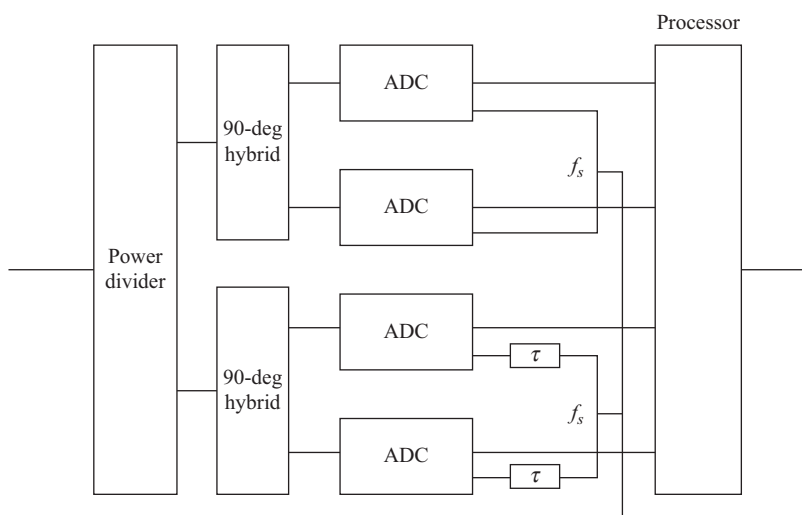


Figure 4.18 Sub-Nyquist sampling with I and Q channels.

Although it appears that the I and Q channels approach has obvious advantages, it has a practical limitation (i.e., the channel imbalance), especially for a wide frequency band system. If the two channels are perfectly balanced in both amplitude and phase, this is the preferred approach. If the two channels cannot be perfectly balanced, the negative frequency cannot be totally cancelled and the amplitude of the negative frequency depends on how well the two channels are balanced. The negative frequency limits the dynamic range of the receiver, and this is discussed in detail in Chapter 6.

If two signals are separated by f_s , they will fold into the same peak and the sub-Nyquist sampling scheme cannot solve them easily. The sub-Nyquist sampling approach can be considered as a special way to fold a wideband input into a narrow output band. This concept is used often in analog EW receivers. A price one pays is that the noise in the input bandwidth will fold into the output band. For example, folding eight input bands into one will increase the noise by 9 dB ($10\log 8$). This increase in noise will reduce the sensitivity of the receiver.

4.13 DFT Through Decimation [25, 26]

This section discusses the decimation scheme of the DFT. The discussion is based on Cooley *et al.* [26] and only the final results will be presented here. It is anticipated that with modification this approach might be applicable to real-time signal processing. This scheme uses many parallel FFT operators to perform many individual FFTs and combines them into a single FFT output. The basic approach is discussed here.

If there are N total points, the data can be divided into r subgroups and each group contains s data points. The FFT is performed on each group and the results are combined to obtain the desired results. The original data points can be represented by $x(n)$, where n is from $n = 0$ to $N - 1$. After regrouping, there are r groups, where each one can be represented as $x_i(n)$, where i is from 0 to $r - 1$. The data point n can be written as

$$n = lr + i \quad i = 0, 1, \dots, r - 1; \quad l = 0, 1, \dots, s - 1. \quad (4.66)$$

This processing is referred to as decimation, or the input data is decimated by r .

Let us use an example to illustrate this operation. Assume $N = 128$ and the data are divided into four groups ($r = 4$), and each group contains 32 data points ($s = 32$). Under this condition, $i = 0, 1, 2, 3$ and $l = 0, 1, 2, \dots, 31$. If $i = 0$, the data points consist of $n = 0, 4, 8, 12, \dots, 124$, which corresponds to $l = 0$ to 31. These results mean that $x_0(0) = x(0)$, $x_0(1) = x(4)$, $x_0(2) = x(8)$, $\dots$, $x_0(31) = x(124)$. A similar argument can be applied to the following: for $i = 1$, the data points consist of $x(n)$, where $n = 1, 5, 9, 13, \dots, 125$; for $i = 2$, $n = 2, 6, 10, 14, \dots, 126$; and for $i = 3$, $n = 3, 7, 11, 15, \dots, 127$. Figure 4.19 shows only the first few points of the data.

From the four subgroups, four DFTs will be performed. The result is

$$X_i(k) = \sum_{n=0}^{s-1} x_i(n) e^{-j2\pi kn/s}, \quad (4.67)$$

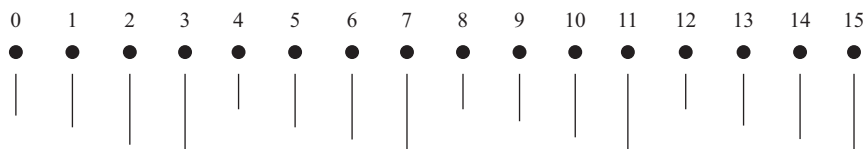


Figure 4.19 Data divided into four subgroups.

where n is given by (4.66). In the above equation, each X_i contains s (for this example $s = 32$) frequency components or k ranges from 0 to $s - 1$ (0 to 31). The final DFT of the N points can be obtained by combining the individual results from (4.67) as

$$X(k) = \frac{1}{r} \sum_{i=0}^{r-1} X_i(k \bmod s) e^{-j2\pi ki/N}, \quad (4.68)$$

where $(k \bmod s)$ means the remainder of k divided by s . If $k = 68$, since $s = 32$, $k/s = 68/32 = 2 + 4/32$, and the remainder is 4.

Since $X(k)$ is the DFT of all the points $x(n)$, there should be N frequency components or the k value of $X(k)$ is from 0 to 127 (for this example). However, the value of k in $X_i(k)$ is from 0 to 31, thus the $(k \bmod s)$ operation is used to change the k value. Let us use the above example to demonstrate this point. If $k = 2$, then $(k \bmod s) = (2 \bmod 32) = 2$, thus

$$X(2) = \frac{1}{4} \sum_{i=0}^3 X_i(2) e^{-j2\pi 2i/N}. \quad (4.69)$$

It takes four complex multiplications to obtain the above result. For a component at $k = 34$, $(k \bmod s) = (34 \bmod 32) = 2$, thus

$$X(34) = \frac{1}{4} \sum_{i=0}^3 X_i(2) e^{-j2\pi 34i/N}. \quad (4.70)$$

It should be noted that in the calculation at $k = 34$, the same $X_i(2)$ is used because there is no $X_i(34)$ available from (4.68). The only difference between (4.69) and (4.70) is that as a kernel function, the former uses $k = 2$, whereas the latter one uses $k = 34$.

If one wants to calculate all 128 points from this method, it appears that the last calculation as shown in (4.68) will take $N \times r$ complex multiplications. In this example it will take 512 (4×128) operations, which is more complicated than the straightforward FFT approach, which takes approximately $(128/2)\log_2(128) = 448$ operations. In addition to the last-step operation, there are four separate FFT operations to find the four $X_i(k)$ operations. Thus if one uses this approach to find all the components of the DFT, it may not be very attractive. Although the input can be processed in parallel, to combine all four individual operations is also rather complex. However, the operation in (4.68) should be further investigated to see whether it can be simplified.

4.14 Applications of the Decimation Method in EW Receivers

The basic idea in this section is to calculate all the FFTs in the subgroups and find the peaks. From these peak values one needs only to calculate the frequencies of interest. This operation uses less calculation than to find all the frequency components. Although the above decimated method to obtain FFT may not be very attractive to generate the FFT data, this modification might be applicable to EW receivers. Since an EW receiver is required to find only a few signals in a short time interval, as discussed in Chapter 2, all the frequency components from the FFT may not be needed.

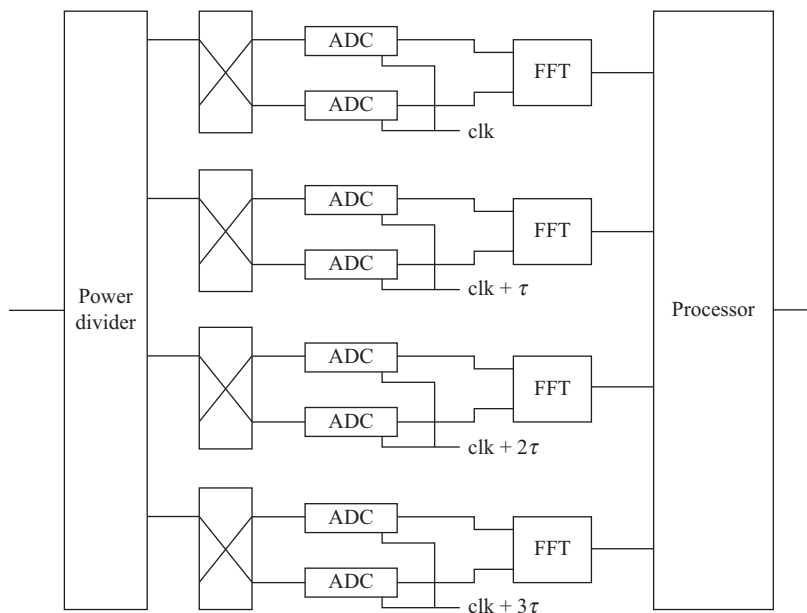


Figure 4.20 Input signals divided into four parallel channels.

Let us assume that the input signal is downconverted to complex outputs and four pairs of ADCs are used to digitize the output. If each ADC is operating at 250 MHz and staggered by 1 ns, this arrangement is equivalent to sampling at 1 GHz. If 128 data points are used for the calculation, the total data collection time is 128 ns. The probability of intercepting a large number of simultaneous signals is very low. Therefore the receiver can be designed to process only a few signals.

Let us use an example to illustrate this idea. Figure 4.20 shows the arrangement. In this example, each output has I and Q outputs. The ADC operates at 250 MHz and the delay time τ is 1 ns. If one considers only one pair of ADCs, the sampling rate is 250 MHz and the corresponding bandwidth is 250 MHz because of the I and Q channels.

In this arrangement, $N = 128$, $r = 4$, and $s = 32$. If there are two sinusoidal signals contained in the data, the power spectrum obtained from any one of the four FFT outputs should have two distinct peaks. First, all four FFTs are used to calculate the four sets of outputs $X_i(k)$ for $i = 0$ to 3. Let us represent these two peaks by the amplitudes of $X_0(k_1)$ and $X_0(k_2)$. It should be noted that the peaks in all the four outputs of $X_i(k)$ for $i = 0$ to 3 are at the same k values and have the same amplitudes. In other words, once k_1 and k_2 are found from the amplitude of $|X_0(k)|$, the remaining three $X_i(k)$ (for $i = 1, 2, 3$) with these k values will be used in (4.68) to find the correct frequency.

For each peak (or each k value) in $X_i(k)$ there are four possible values in the overall FFT output: $X(k)$, $X(k + s)$, $X(k + 2s)$, and $X(k + 3s)$, where $s = 32$ in this example. One needs to find these four components to determine the true peak. To find one of the $X(k + is)$ ($i = 0, 1, 2$, and 3) values from (4.68) requires four complex multiplications, and a total of 16 operations are required. Two signals require 32 operations. The total operations including

the four sets of FFTs are $4(32/2)\log_2(32) + 32 = 352$, which is less than the 448 operations required for straightforward 128-point FFTs.

When the two signals are separated by an integer multiple of 250 MHz, the two input frequencies will fold into one peak. Under this condition, the peaks from each $X_i(k)$ are different in amplitude because of the different phases in the two signals. This is an important sign that two signals are folding into one peak. The highest peak can be used to choose the k value. Once the k value is selected, $X(k)$, $X(k + s)$, $X(k + 2s)$, and $X(k + 3s)$ are calculated. Two of these $X(k + is)$ values should have peaks, and they represent the true frequencies. This requires only 16 complex multiplications.

In general, if there are p signals, the total number of complex multiplications (N_c) can be approximated as

$$N_c = r\left(\frac{s}{2}\right)\log_2 s + pr^2, \quad (4.71)$$

where r is total number of groups and s is the number of points in each subgroup. When this number is equal to the operations required for a straightforward FFT, the advantage of this method will disappear. This condition is

$$\frac{rs}{2}\log_2(rs) = \frac{rs}{2}\log_2 s + pr^2$$

or

$$p = \frac{s}{2r}\log_2 r. \quad (4.72)$$

For the above example, p can be as large as 32.

Of course, the calculation can be further simplified by finding $X_0(k)$ and its peak values at k_1 and k_2 and so forth. Then find the other three $X_i(k)$ at these k values. This way, one needs to use the FFT once instead of four times. However, there is a chance when the two signals are separated by 250 MHz that the amplitude in $X_0(k)$ may be very low and one will miss the signal. If all four $X_i(k)$ values are calculated, the four sets of outputs have different amplitudes and the signal can always be recognized.

4.15 Simplified Decimation Method [27]

In this section, the decimation method is further simplified by reducing the hardware. This basic approach is somewhat similar to the sub-Nyquist sampling scheme discussed in Section 4.10.

Let us continue to use the above example to demonstrate the approach. The basic idea is to use two sets of data delayed by t to find the peaks and the corresponding phase differences through $X_0(k)$ and $X_1(k)$. From this phase difference, the value of $X_2(k)$ and $X_3(k)$ can be estimated. When all the $X_i(k)$ values are obtained, the true frequency can be found through (4.68).

In Figure 4.21, only the first two pairs of samplers are kept. The sampling rate is maintained at 250 MHz and the delay time τ is still 1 ns. Under this arrangement, only half of the data shown in Figure 4.21 will be obtained. A total of 64 points of complex data will be

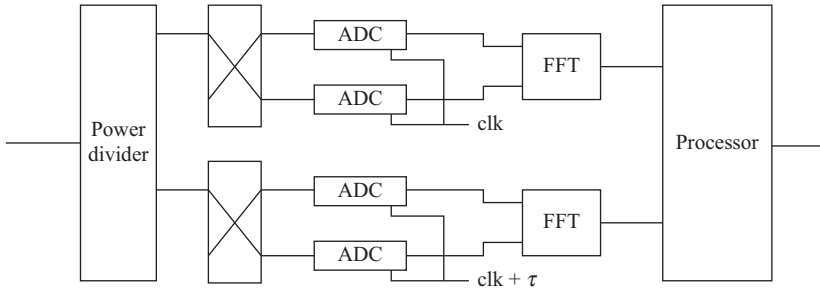


Figure 4.21 Simplified decimation method.

collected. In Figure 4.19, data points 0, 1, 4, 5, 8, 9... are available, but the other half consisting of 2, 3, 6, 7, 10, 11... will not be obtained. From the first group of 32 points one can calculate $X_0(k)$ and from the amplitude of $|X_0(k)|$ one can find two peaks, k_1 and k_2 . From the second group of 32 points one can find $X_1(k_1)$ and $X_1(k_2)$. From $X_0(k_m)$ and $X_1(k_m)$, where $m = 1, 2$ for this case, one can find the phase difference, which can be written as

$$\theta_m = \tan^{-1} \left\{ \frac{\text{Im}[X_1(k_m)]}{\text{Re}[X_1(k_m)]} \right\} - \tan^{-1} \left\{ \frac{\text{Im}[X_0(k_m)]}{\text{Re}[X_0(k_m)]} \right\}. \quad (4.73)$$

From θ_m one can find the frequency of the input signal as in the phase delay method. Now, let us use the θ_m values to generate $X_2(k_m)$ and $X_3(k_m)$ since the phase difference from one set of data to another set can be expressed as

$$X_{i+1}(k_m) = X_i(k_m)e^{-j\theta_m}. \quad (4.74)$$

From this equation one can find that

$$\begin{aligned} X_2(k_m) &= X_0(k_m)e^{-j2\theta_m} \\ X_3(k_m) &= X_0(k_m)e^{-j3\theta_m}. \end{aligned} \quad (4.75)$$

Once all the $X_i(k_m)$ values are found, the quantities of $X(k)$, $X(k + s)$, $X(k + 2s)$, and $X(k + 3s)$ can be calculated from (4.68).

The major advantages of this approach are that less hardware is used and the processing can be relatively simple since some of the $X_i(k_m)$ values are calculated by a simple phase shift.

One of the possible disadvantages is that this approach may lose a signal. When two signals are separated by a multiple of f_s , the two peaks in the power spectrum from the two input signals are folded into a single peak. If all the information is available, such as described in Section 4.13, the input signals can be uniquely determined. However, if there are only two pairs of ADCs, there is not enough information to determine the two frequencies.

The following question needs to be answered: If there are only two pairs of ADCs and there is only one peak detected from the power spectrum, how can one tell if there are two signals rather than one? This question can be answered from the amplitudes of $X_0(k_m)$ and $X_1(k_m)$, where k_m represents the frequency component in which the peak of the power

spectrum appears. When there is only one signal, the amplitudes of $X_0(k_m)$ and $X_1(k_m)$ are equal. When there are two signals, the amplitudes of $X_0(k_m)$ and $X_1(k_m)$ are different. The reason of this argument can be expressed as follows. For one signal

$$X_0(k_m) = c_1$$

and

$$X_1(k_m) = c_1 e^{-j\theta_1} = c_1 e^{-j2\pi f_1 \tau}, \quad (4.76)$$

where c_1 is a constant, θ_1 is the phase shift caused by the delay, f_1 is the input frequency, and τ is the delay time. It is obvious that the amplitudes of $X_0(k_m)$ and $X_1(k_m)$ are equal.

For two signals, the results can be written as

$$\begin{aligned} X_0(k_m) &= c_1 + c_2 \\ X_1(k_m) &= c_1 e^{-j\theta_1} + c_2 e^{-j\theta_2} = c_1 e^{-j2\pi f_1 \tau} + c_2 e^{-j2\pi f_2 \tau}, \end{aligned} \quad (4.77)$$

where c_2 is a constant and θ_2 and f_2 are the phase shift and frequency of the second signal, respectively. In general, the amplitude of $X_0(k_m)$ and $X_1(k_m)$ are not equal. From the relative amplitude of $X_i(k_m)$, one can find there is more than one signal. However, there is not enough information to determine their frequencies. Additional information is required to determine the frequency of more than one signal.

References

- [1] Brigham EO. *The Fast Fourier Transform*. Englewood Cliffs, NJ: Prentice Hall; 1973.
- [2] Rabiner LR. *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1975.
- [3] Oppenheim AV, Schaffer RW. *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1975.
- [4] Kay SM. *Modern Spectral Estimation Theory and Application*. Englewood Cliffs, NJ: Prentice Hall; 1988.
- [5] Marple SL, Jr. *Digital Spectral Analysis with Applications*. Englewood Cliffs, NJ: Prentice Hall; 1987.
- [6] Elliott DF (ed.). *Handbook of Digital Signal Processing: Engineering Applications*. San Diego: Academic Press; 1987.
- [7] Tsui JBY. *Digital Microwave Receivers: Theory and Concepts*. Norwood, MA: Artech House; 1989.
- [8] Shanmugan KS, Breipohl AM. *Random Signals Detection, Estimation and Data Analysis*. New York: John Wiley; 1988.
- [9] Papoulis A. *Probability, Random Variables, and Stochastic Processes*. New York: McGraw-Hill; 1965.
- [10] Kleinrock L. 'Detection of the peak of an arbitrary spectrum'. *IEEE Transactions on Information Theory* 1964;**10**(3):215–221.

- [11] Palmer L. 'Coarse frequency estimation using the discrete Fourier transform'. *IEEE Transactions on Information Theory* 1974;**20**(1):104–109.
- [12] Rife DC, Boorstyn RR. 'Single-tone parameter estimation from discrete-time observations'. *IEEE Transactions on Information Theory* 1974;**20**(5):591–598.
- [13] Rife DC, Vincent GA. 'Use of the discrete Fourier transform in the measurement of frequencies and levels of tones'. *Bell System Technical Journal* 1970;**49**(2):197–228.
- [14] Rife DC, Boorstyn RR. 'Multiple tone parameter estimation from discrete-time observations'. *Bell System Technical Journal* 1976;**55**(9):1389–1410.
- [15] Ng S., 'A technique for spectral component location within a FFT resolution cell.' *Proceedings ICASSP '84. IEEE International Conference on Acoustics, Speech, and Signal Processing*. New York: IEEE; 1984:147–149.
- [16] Pasala K. University of Dayton, private communication.
- [17] Papoulis A. *The Fourier Integral and Its Applications*. New York: McGraw-Hill; 1962.
- [18] Rader CM. 'Recovery of undersampled periodic waveforms'. *IEEE Transactions on Acoustics, Speech and Signal Processing* 1977;**25**(3):242–249.
- [19] Sanderson RB, Tsui JBY, Freese NA. 'Reduction of aliasing ambiguities through phase relations'. *IEEE Transactions on Aerospace and Electronic Systems* 1992;**28**(4):950–956.
- [20] Shapiro HS, Silverman RA. 'Alias-free sampling of random noise'. *Journal of the Society for Industrial and Applied Mathematics* 1960;**8**(2):225–248.
- [21] Beutler FJ. 'Error-free recovery of signals from irregularly spaced samples'. *SIAM Review* 1966; **8**(3):328–335.
- [22] Beutler FJ. 'Alias-free randomly timed sampling of stochastic processes'. *IEEE Transactions on Information Theory* 1970;**16**(2):147–152.
- [23] Jenq Y-C. 'Digital spectra of nonuniformly sampled signals: fundamentals and high-speed waveform digitizers'. *IEEE Transactions on Instrumentation and Measurement* 1988; **37**(2):245–251.
- [24] Cheng C-H, Liou L, Lin D, Tsui J, Tai H-M. 'Wideband in-phase/quadrature imbalance compensation using finite impulse response filter'. *IET Radar, Sonar & Navigation* 2014; **8**(7):797–804.
- [25] Vaidyanathan PP. *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice Hall; 1992.
- [26] Cooley JW, Lewis PAW, Welch PD. 'The finite Fourier transform'. *IEEE Transactions on Audio and Electroacoustics* 1969;**17**(2):77–85.
- [27] Choate DB, Tsui JBY. 'Note on Prony's method'. *IEE Proceedings F—Radar and Signal Processing* 1993;**140**(2):103–106.

Analog-to-Digital Converters, Amplifiers, and Their Interface

5.1 Introduction

The major difference between an analog receiver and a digital receiver is that instead of using a crystal detector, the digital receiver uses an analog-to-converter (ADC) to convert the microwave signal to digitized data and the receiver's bandwidth is determined by the ADC's sampling rate. Before the received signal is digitized by an ADC, the signal is amplified by an amplifier chain containing several amplifiers with different gains, noise figures, and third-order intermodulation points. One of the purposes of using amplifiers in front of the ADC is to match the input signal to the ADC. In general, adding amplifiers will improve the sensitivity of the receiver.

The main purpose of this chapter is to present an optimum way to match the radio frequency (RF) amplifier with the ADC. The word "optimum" means obtaining a certain sensitivity and dynamic range, desired by the designer, within the limits of the amplifier performance and the ADC.

This chapter will very briefly discuss analog receiver performance and point out the difference between it and digital receiver performance. The ADC and its impact on the receiver performance will then be discussed. The important parameters of the ADC related to receiver performance are the number of bits, number of effective bits, maximum sampling frequency, and input bandwidth. The most significant effect of an ADC is on the dynamic range of the receiver, which is closely related to the sensitivity of the receiver since the sensitivity is the lower limit of the dynamic range of the receiver. There are several ways to consider the dynamic range, and each approach leads to a slightly different result. All these approaches will be discussed. The performance of linear amplifiers, their gains, noise figures, and third-order intercept points and the interface between the amplifier and the ADC will also be presented. A detailed discussion of analog receivers and amplifiers can be found in Tsui [1]. Finally, a simple program will be provided to produce different combinations of sensitivity and dynamic range of a receiver. The designer can pick the desired performance.

5.2 Key Component Selection [2–16]

Analog microwave receivers have been built for many years. They include many different types of receivers (e.g., communication and electronic warfare [EW] receivers). In most of

these receivers, microwave components (i.e., amplifiers, attenuators, mixers, local oscillators) are used. After these components, a crystal video detector is used to convert the RF into a video signal, which is further processed.

After many years of development, many different components are available. For example, there is a large selection of microwave amplifiers with different operating frequency ranges, noise figures, gains, and so forth that one can choose from. Even so, it is probably very difficult to choose one amplifier that has the desired performance. However, in receiver design, many different RF amplifiers can be connected in series with proper attenuators added between them to obtain a characteristic close to the desired performance.

On the other hand, the technology of manufacturing ADCs with a high sampling frequency and a high number of bits (i.e., several gigahertz sampling frequency and more than 8 bits) has significantly progressed in recent years, yet there are still only a few ADCs with operating frequencies above 2 GHz that also have 10 bits. Because of the limited choice of available high-speed ADCs, when designing a digital receiver the ADC is the first component to be selected. The RF amplifier chain is then designed to match the selected ADC. The performance of the RF amplifier chain (i.e., the noise figure), the gain, and the third-order intercept point are chosen to optimize the receiver performance based on selection of the ADC. Once the performance is chosen, one can select different microwave components (i.e., amplifiers and attenuators) and connect them in the desired manner to match the calculated performance.

In a receiver, especially a wideband system, one would like both high sensitivity and dynamic range. Unfortunately, the higher the gain in a receiver, the lower the third-order intercept. As a result, high sensitivity means low dynamic range. Because of this property, one may prefer slightly higher sensitivity in some cases and higher dynamic range in other cases. The design procedure is to provide a list of receiver performances with different sensitivities and dynamic ranges versus different characteristics of amplifier chains. In this way a designer can pick up the desired combination.

5.3 Comparison of Analog and Digital Receiver Sensitivity [1]

In an analog receiver, after the RF chain there is a crystal detector to convert the microwave signal into a video signal. For an EW receiver, this video signal is often digitized and further processed through digital signal processing to generate the pulse descriptor word (PDW). Often the RF chain is designed to have enough gain to amplify the input noise floor to the tangential sensitivity of the detector [1]. Tangential sensitivity is defined for a pulsed signal on a scope display. It is the signal level where the minimum of the noise trace in the pulse region is roughly tangential to the top of the noise trace between pulses. With enough RF gain, the detector itself does not play any role in the noise figure, sensitivity, or dynamic range of the receiver. Only the video bandwidth following the detector and the RF chain (including the RF bandwidth and noise figure) determine the sensitivity of the receiver.

In a digital receiver, there is no crystal detector. The input signal is digitized and processed. Sometimes the signal is downconverted before digitization. The digital processing determines the RF bandwidth and the video bandwidth. If the fast Fourier transform (FFT) is used to process the input signal, the RF bandwidth equals the video bandwidth since the signal can be considered as filtered by the FFT operation. As a result, the sensitivity of the receiver is

dependent on the length of the FFT operation. Therefore the ADC should be considered as part of the RF chain. This phenomenon is further demonstrated in later sections.

Since the major difference between an analog and a digital receiver is the application of an ADC, the following 11 sections (Sections 5.4 through 5.14) are dedicated to the discussion of ADCs, and we start with a sample and hold circuit.

5.4 Basic Sample and Hold Circuit [17–19]

In order to quantize an input signal at a certain instant in time, the signal should be held constant at that instant. If the input signal changes rapidly and the digitizing process is slow, the accuracy of the output data will be poor. One approach is to place a sample and hold circuit in front of a quantizer. A sample and hold circuit can create a very narrow aperture window, which will take the input at the desired instant and keep the voltage constant for a relatively long period of time, such that the digitizing circuit can operate properly.

A simple sample and hold circuit is shown in Figure 5.1. It consists of a sample/hold switch and a holding capacitor. The two amplifiers are used for impedance matching.

When the switch is closed, the sample and hold is in the sample mode and the voltage on the capacitor follows the input voltage. When the switch is open, the voltage on the capacitor stays at a constant value and is called the hold mode.

In the sample mode, the operation can be divided into two time periods: acquisition and tracking. When the switch is closed, the voltage on the capacitor starts to change rapidly from the previously held value toward the input voltage until it finally reaches it. This period is called the acquisition time. The voltage on the capacitor follows the input voltage; this period is called the tracking time.

When the switch is opened, the voltage on the capacitor should stay constant; this is the hold mode. However, after the switch is open, the voltage on the capacitor usually oscillates slightly due to a transient effect; this is called the settling time. After the settling time, the voltage will drop slightly due to charge leakage through the finite input impedance of the amplifier. Figure 5.2 shows the time response of a sample and hold circuit. The aperture time is the elapsed time between initiation and completion of the mode transition from sample to hold.

The aperture uncertainty is the variance of the aperture time. This time can be very short, usually in the picoseconds range for high-frequency ADCs. The aperture jitter is the variation in the effective sample instant due to the aperture uncertainty. The sampling time offset is the time interval between the sample to hold mode transition command and the actual initiation of the transition. The sampling time uncertainty is the variance of the sampling time offset.

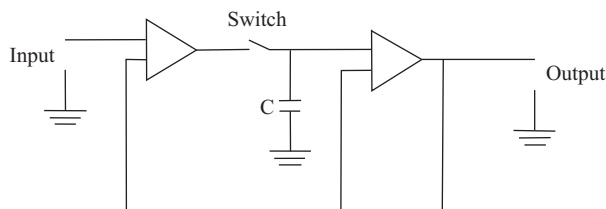


Figure 5.1 Basic sample and hold circuit.

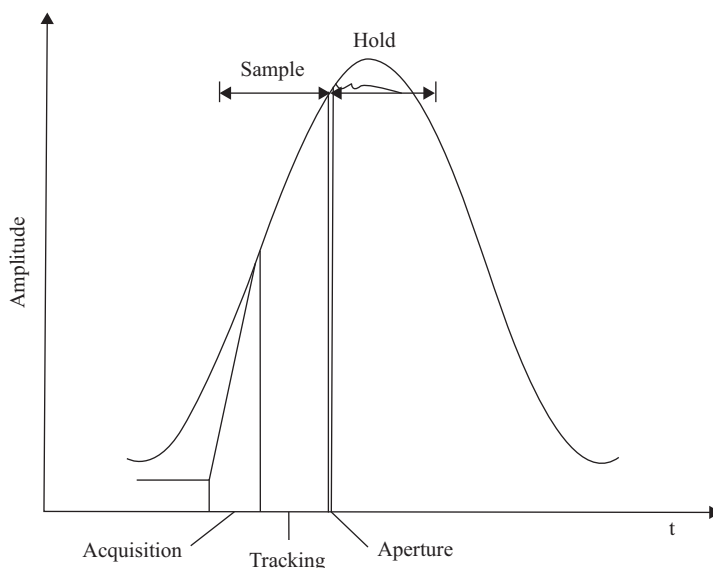


Figure 5.2 Time domain response of a sample and hold circuit.

5.5 Basic ADC Performance and Input Bandwidth

An ADC converts a continuous input voltage into discrete output levels, which can be represented by binary coded words. The smallest discrete step size is called the quantization level. The conversion usually occurs at uniformly spaced time intervals. This is often referred to as the sampling time. The transfer function representing the input versus output of an ADC is shown in Figure 5.3. Figures 5.3(a) and 5.3(b) show the 3-bit midread and midriser approaches, respectively. The x -axis is the analog input and the y -axis represents the digital output. In the midread configuration, there is a zero level, but the number of positive levels does not equal the negative levels since the total number of levels is usually a power of 2 number. In this figure there is one more negative level than the number of positive levels. It is obvious that the midread approach has an unsymmetrical output.

In the midriser approach, there is no zero level, but they have an equal number of positive and negative levels. Thus the output is symmetrical. In testing high-frequency ADCs, a sine wave is often used. Since the sine wave is symmetrical, the midriser model is often applied.

Figure 5.4 shows the transfer characteristic of an ideal ADC. If the input increases linearly with respect to time, the output and the quantization error are as shown in Figure 5.4. It is obvious that the quantization process is a nonlinear one, making it difficult to analyze mathematically. In a practical ADC it is also difficult to make the quantization level uniform, thus the quantization error is worse than in the ideal case.

In order to fulfill the Nyquist sampling criterion, it is often assumed that the input frequency to an ADC is half the sampling frequency, but this is not always true. In order to avoid aliasing ambiguity, it is required that the input bandwidth (not necessarily the input frequency) of the ADC be less than half of the sampling frequency. It is often desirable to have an input frequency higher than the maximum sampling frequency because the input bandwidth does not necessarily start from a direct current (dc) level. For example, if the maximum

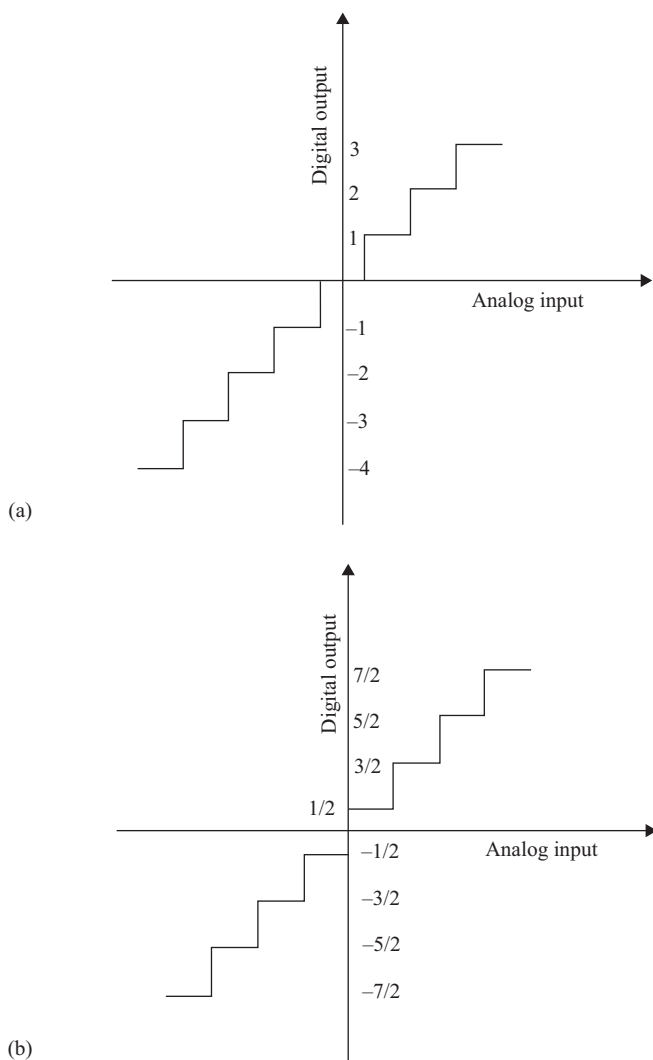


Figure 5.3 Transfer function of an ADC: (a) midtread and (b) midriser.

sampling frequency of the ADC is 200 MHz, the unambiguous bandwidth is 100 MHz. The input frequency spectrum does not necessarily extend from dc to 100 MHz. It can be from 120 to 220 MHz, and this selection keeps the input bandwidth below an octave. An input bandwidth of less than one octave will eliminate the second harmonic, which can be generated by an analog front end or by the ADC nonlinear transfer characteristic. Octave bandwidth means the highest frequency is double the lowest frequency, such as from 1 to 2 GHz.

Sometimes ADCs are used in parallel to increase sampling speed, as discussed in the previous chapter. If the input frequency of the ADC is high, these ADCs can be used to increase the bandwidth of the system. If the input frequency of the ADC is limited to half the sampling frequency, a parallel approach cannot be adopted because the ADC cannot receive high-frequency input signals.

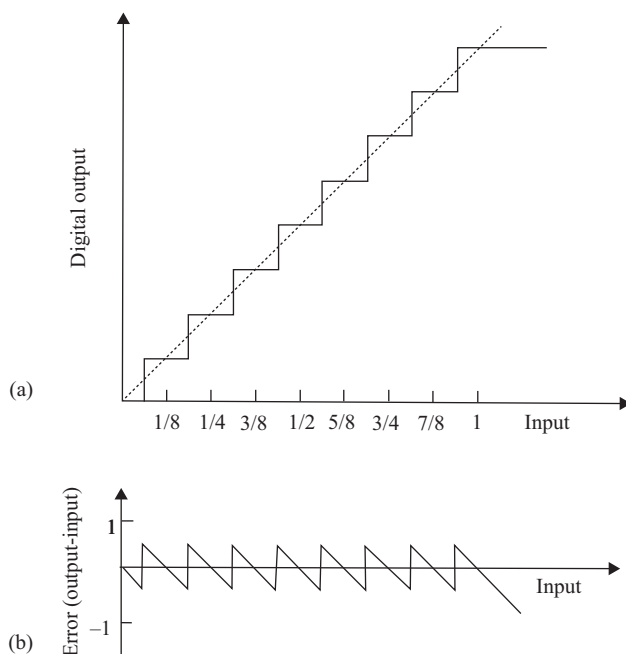


Figure 5.4 Performance of an ideal ADC: (a) input versus output and (b) quantization error.

5.6 Apparent Maximum and Minimum Signals to an ADC

The maximum signal to an ADC is often defined as a sine wave with an amplitude that matches the highest level of the ADC. If a signal is stronger than this maximum level, the output waveform will be clipped. If an input signal is lower than this signal, the total number of bits may not be used. The maximum level is often used to determine the upper limit of the dynamic range. If there is no noise and the input voltage matches the maximum range of the ADC, the voltage V_s is

$$2V_s = 2^b Q$$

or

$$V_s = 2^{(b-1)} Q, \quad (5.1)$$

where b is the number of bits and Q is the voltage per quantization level. This equation shows that the sine wave reaches the top of the highest as well as the bottom of the lowest quantization levels.

A sine wave with an amplitude matching this maximum voltage has a power of

$$P_s = \frac{V_s^2}{2} = \frac{2^{2(b-1)} Q^2}{2} = \frac{2^{2b} Q^2}{8}. \quad (5.2)$$

In the above equation, the input impedance is assumed to be unity. In later discussions, the power ratio will be of interest and hence the impedance is canceled when the ratio of

powers is taken. However, in some calculations it is necessary to include the impedance in order to have the actual power.

If there is no noise, the minimum signal is considered as the voltage that can cause change in the least significant bit (LSB), otherwise the ADC cannot detect the signal. Under this condition, the minimum voltage $V_{\min}$ is equal to one quantization level, or

$$2V_{\min} = Q. \quad (5.3)$$

The corresponding power level is

$$P_{\min} = \frac{V_{\min}^2}{2} = \frac{Q^2}{8}. \quad (5.4)$$

The dynamic range can be defined as the ratio of P_s to $P_{\min}$, which can be written as

$$DR = \frac{P_s}{P_{\min}} = 2^{2b}, \quad (5.5)$$

which is often expressed in logarithmic form as

$$DR = 10 \log\left(\frac{P_s}{P_{\min}}\right) = 20b \log(2) \approx 6b \text{ dB}. \quad (5.6)$$

This is why the dynamic range of an ADC is often referred to as 6-dB per bit.

However, the dynamic range of a receiver also depends on the amplifier performance in front of the receiver and the ADC. This problem is discussed in later sections of this chapter.

5.7 Quantization Noise of an Ideal ADC [17–25]

An ADC converts the input signal from analog to digital form, and this process is nonlinear. For example, a 1-bit ADC is equivalent to a hard limiter, which is a nonlinear device. As shown in Figure 5.5, a sine wave is converted point by point into two different output levels. There is a difference (or error) between the true value of the sine wave and the quantized one. Because the error can be any value within the quantization level, it is reasonable to assume that the probability of the error is uniformly distributed over the quantization level Q . Thus the probability density function of the amplitude is $1/Q$. The quantization noise power can be calculated from the error as

$$N_b = \frac{1}{Q} \int_{-Q/2}^{Q/2} x^2 dx = \frac{Q^2}{12}. \quad (5.7)$$

This quantity is sometimes used as the sensitivity level of the receiver. Under this condition, the maximum signal-to-noise (S/N) ratio, by combining (5.2), can be expressed as

$$\left(\frac{S}{N}\right)_{\max} = \frac{P_s}{N_b} = \frac{3}{2} 2^{2b}. \quad (5.8)$$

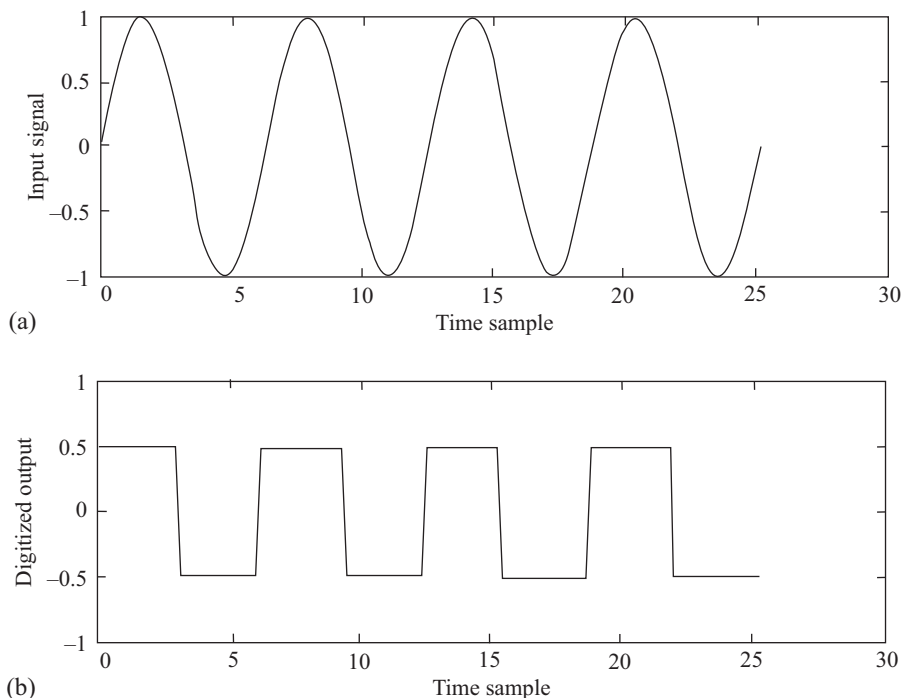


Figure 5.5 Sine wave and one bit quantizer: (a) input signal and (b) digitized output.

This quantity can be expressed in logarithmic form as

$$DR = 10 \log \left(\frac{P_s}{N_b} \right) = 10 \log(1.5) + 20b \log(2) = 1.76 + 6.02b \text{ dB}. \quad (5.9)$$

The difference between (5.6) and (5.9) is the factor 1.76 since the two lower limits are different.

5.8 Noise Level Determined by Processing Bandwidth and Dithering Effect [26, 27]

The lower limit (noise level) of the dynamic range depends on the processing bandwidth. The processing bandwidth mentioned here is usually data length dependent. If one determines to perform an N -points DFT on the output of the ADC, the processing bandwidth B_v is related to the DFT length as

$$B_v = \frac{f_s}{N} = \frac{1}{Nt_s}, \quad (5.10)$$

where f_s and t_s are the ADC sampling frequency and sampling time, respectively. Thus a longer DFT operation creates a narrower processing bandwidth. The narrower the bandwidth, the lower the noise level.

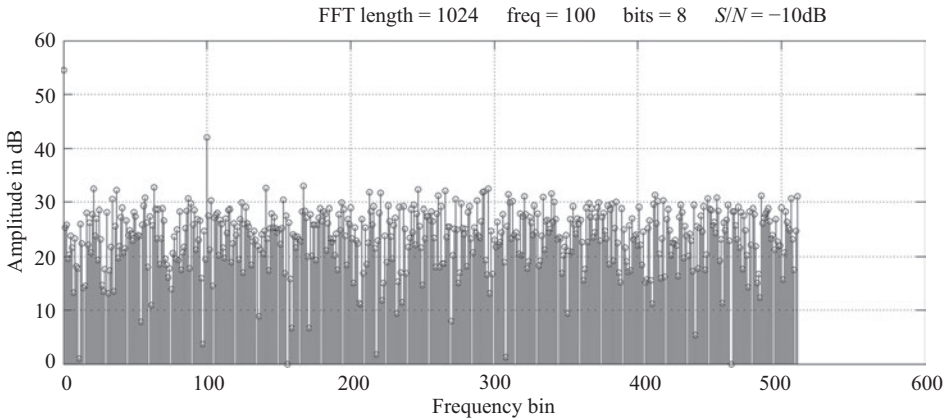


Figure 5.6 Signal detected with noise dithering.

Since the signal is coherent and the noise is incoherent, the signal level from the DFT output is proportional to N and the noise level is proportional to $N^{1/2}$. Therefore, when the length of the DFT increases, the signal amplitude increases faster than the noise level. As a result, a weak signal can be detected by the ADC. However, as mentioned before, the signal must be strong enough to cross the first level of the ADC in order to be detected. Therefore, one can add noise to the signal to be detected by the ADC and at the same time increase the FFT length to improve the S/N ratio. This approach can be used to detect a very weak signal. This processing is often referred to as noise dithering. The purpose of a dithering is to make a weak signal cross the quantization level in an ADC.

Figure 5.6 shows an example of noise dithering. In this example, the input signal alone is too weak to cross the LSB and the output is at a constant level of 0.5. If noise is added, the noise can be sensed by the ADC. In this case the S/N ratio is set at -10 dB. The signal can be identified at the 100th frequency bin. The DC component shown in the spectrum occurs because the duration of signals to be digitized is not a multiple of signal cycles.

If there are two input signals of different amplitude and frequency, an ideal EW receiver will receive both the strong and weak signals. The difference between the strong and weak signals is referred to as the instantaneous dynamic range of the receiver.

If there is only one weak signal that is not strong enough to cross the first quantization level, the ADC cannot sense the signal. If there is a strong signal, the weak signal may not need to cross the first quantization level to be detected. The strong signal will cross different quantization levels and since the weak signal is superimposed on the strong one, both signals may be detected. In this case, the strong signal can be considered as the dithering signal.

In addition to the noise floor, the spurious responses (sometimes referred to as spurs) often limit the lower limit of the dynamic range. The spurious responses in an ADC are discussed in the next section.

5.9 Spurious Response

If an input signal with arbitrary frequency is applied to the input of the ADC, the error between the true signal and the digitized value cannot be predicted, thus it is reasonable to

assume the error is uniformly distributed. However, if the input signal frequency is commensurate with respect to the sampling frequency f_s , the error function is highly correlated. Under this condition, a uniform distribution is no longer a good assumption.

For example, if the input frequency f_i is related to the sampling frequency f_s by

$$f_s = nf_i, \quad (5.11)$$

where n is an integer, the error will exhibit a repetitive pattern from one cycle to the next, as shown in Figure 5.7. From this consideration, the assumption of uniform distribution of error will no longer hold. Figure 5.7(a) shows a sine wave with two cycles that is sampled at 32 points. Figure 5.7(b) shows the quantized version with 3 bits. This case can be considered as $f_s = 32$ and $f_i = 2$. The error in the time domain is shown in Figure 5.7(c). One should note that the errors from points 0 to 15 are the same as those at 16 to 32. If one increases the number of points, the errors will just repeat themselves. Figure 5.7(d) shows the FFT results of the sine wave. Since the input frequency matches one of the frequency bins ($n = 2$) of the FFT output, there are no sidelobes. Figure 5.7(e) shows the FFT result of the quantized version, which contains spurs.

Since the error data repeat every input cycle, increasing the FFT length will not change the levels of the spurs. Figure 5.7(f) shows the results of 64 points of FFT outputs with a signal of four cycles. The spur levels are identical to the results in Figure 5.7(e), thus the spur levels do not decrease with increasing the FFT length. It should be noted that in producing Figure 5.7, the input signal starts from $t = 1 \times 10^{-8}$ to $t = 4\pi + 1 \times 10^{-8}$. If the input signal passes exactly at $t = 0$, the computer error will disturb the quantized data.

These spurs are caused by the quantization error. The DFT of the quantized data can be written as

$$X_d(k) = \sum_{n=0}^{N-1} x_d(n) e^{-j2\pi nk/N} = \sum_{n=0}^{N-1} [x(n) - x_e(n)] e^{-j2\pi nk/N}, \quad (5.12)$$

where $x(n)$ is the input in the time domain, $x_d(n)$ is the corresponding digitized data, and $x_e(n)$ represents the error function. The error in the frequency domain is caused by the DFT of $x_e(n)$.

It has been demonstrated that errors in the digitized signal can be periodic in nature. In (5.11), even if n is not an integer, the output can still be periodic. For example, if $f_s = 32$ Hz and $f_i = 0.5$ or 1.5 Hz, the output will repeat itself every 64 points instead of every 32 points. This section illustrates that some of the spurs cannot be reduced by increasing the FFT length.

5.10 Analysis on Spur Amplitudes [28–30]

In the previous section it was demonstrated that digitized signals generate spurious responses. Some of spurs cannot be reduced, even if a longer FFT is used. It is desirable to find the maximum of the spur level so that the dynamic range of the receiver can be determined. Spurs can be generated through other processes (i.e., through the nonideal characteristics of ADCs). Sometimes sidelobes can be considered as spurs since they too will limit the instantaneous dynamic range of a receiver.

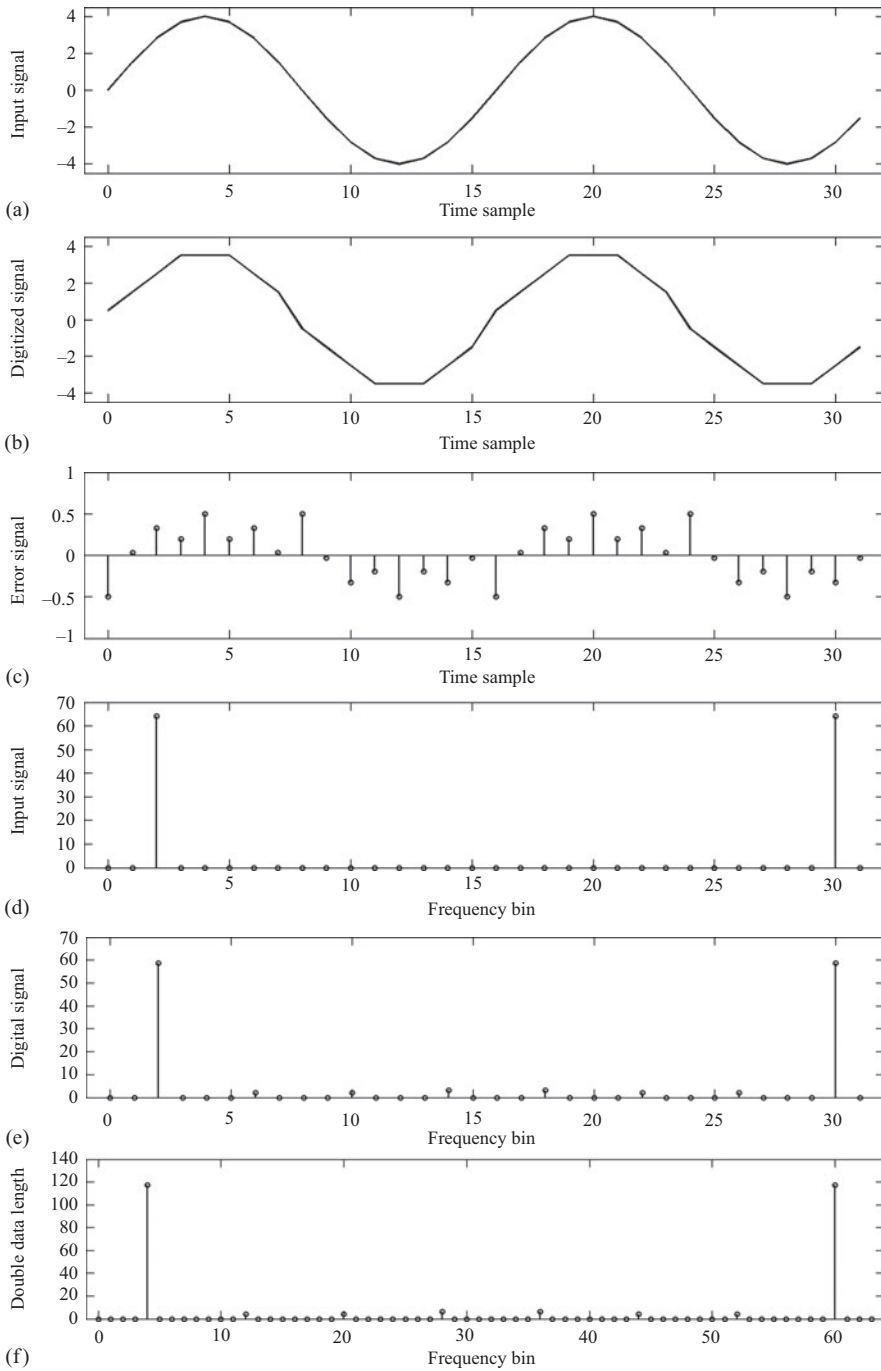


Figure 5.7 Coherent digitizing error: (a) input signal, (b) digitized output, (c) error signal, (d) FFT of the input signal, (e) FFT of the digitized output, and (f) FFT of the digitized output with 64 points.

This section is a study of the amplitude of the spurs based on West [29]. Since digitization is a nonlinear process, a general analysis is not achievable. The following discussion can be considered as a special case, but it does provide some interesting results. If x is the input signal to an ADC, x_d is the output signal, and g is the transfer characteristic function of the ADC, then

$$x_d(t) = g(x(t)). \quad (5.13)$$

If the input is a cosine wave, then

$$x(t) = \cos(2\pi ft + \theta) \equiv \cos \phi(t)$$

and

$$x_d(t) = g[\cos \phi(t)]. \quad (5.14)$$

Since the digitized signals are inherently periodic, they can be represented by the Fourier series as

$$x_d(t) = A_0 + \sum_{n=1}^{\infty} A_n \cos(n\phi), \quad (5.15)$$

where A_1 is the amplitude of the fundamental frequency and the higher orders of A_n represent the spurious responses. The values of A can be found as

$$A_0 = \frac{1}{\pi} \int_0^{\pi} g(\phi) d\phi$$

$$A_n = \frac{2}{\pi} \int_0^{\pi} g(\phi) \cos(n\phi) d\phi. \quad (5.16)$$

Let us use an example to demonstrate this approach. Figure 5.8 shows that a cosine wave is digitized by a 3-bit ADC into eight levels and the amplitude of the signal matches the full level of the ADC. The function of $g(\phi)$ can be written as

$$g(\phi) = \begin{bmatrix} 7/8 & \cos^{-1}(1) \geq \phi \geq \cos^{-1}(3/4) \\ 5/8 & \cos^{-1}(3/4) \geq \phi \geq \cos^{-1}(1/2) \\ 3/8 & \cos^{-1}(1/2) \geq \phi \geq \cos^{-1}(1/4) \\ 1/8 & \cos^{-1}(1/4) \geq \phi \geq \cos^{-1}(0) \\ -1/8 & \cos^{-1}(0) \geq \phi \geq \cos^{-1}(-1/4) \\ -3/8 & \cos^{-1}(-1/4) \geq \phi \geq \cos^{-1}(-1/2) \\ -5/8 & \cos^{-1}(-1/2) \geq \phi \geq \cos^{-1}(-3/4) \\ -7/8 & \cos^{-1}(-3/4) \geq \phi \geq \cos^{-1}(-1) \end{bmatrix}. \quad (5.17)$$

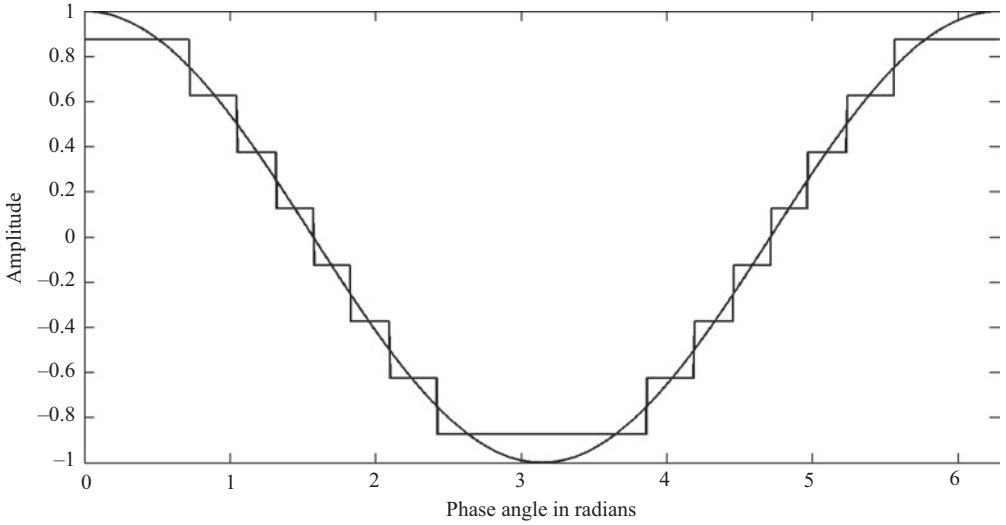


Figure 5.8 A cosine wave digitized by a 3-bit ADC.

In this equation, the ADC is a midrise type and the output is normalized to unity. To find the constant A_n , one can substitute $g(f)$ into (5.16) and the result is

$$\begin{aligned}
 A_n &= \frac{2}{\pi} \int_0^{\pi} g(\phi) \cos(n\phi) d\phi \\
 &= \frac{2}{\pi} \left[\frac{7}{8n} \sin(n\phi) \right]_{\cos^{-1}(1)}^{\cos^{-1}(\frac{3}{4})} + \frac{2}{\pi} \left[\frac{5}{8n} \sin(n\phi) \right]_{\cos^{-1}(\frac{3}{4})}^{\cos^{-1}(\frac{1}{2})} + \dots \\
 &\quad + \frac{2}{\pi} \left[\frac{-7}{8} \sin(n\phi) \right]_{\cos^{-1}(-1)}^{\cos^{-1}(-\frac{3}{4})} \\
 &= \frac{1}{2n\pi} \left\{ 1 + 2 \sin \left[n \cos^{-1} \left(\frac{1}{4} \right) \right] + 2 \sin \left[n \cos^{-1} \left(\frac{1}{2} \right) \right] \right. \\
 &\quad \left. + 2 \sin \left[n \cos^{-1} \left(\frac{3}{4} \right) \right] \right\}. \tag{5.18}
 \end{aligned}$$

In general, this approach can be extended to b bits. If an ADC has b bits, the amplitude of the n th harmonics can be found in a similar way as

$$A_n = \frac{(-4)^{n+1}}{2^b n \pi} \left[1 + 2 \sum_{k=1}^{\frac{2^b}{2}-1} \sin \left[\left[n \cos^{-1} \left(\frac{2k}{2^b} \right) \right] \right] \right]. \tag{5.19}$$

Table 5.1 Highest spur level versus number of bits.

Number of bits	Largest spur (dBc)
1	−9.5
2	−18.3
3	−27.0
4	−35.9
5	−44.8
6	−53.8
7	−62.8
8	−71.8

In this equation, A_1 is the amplitude of the fundamental component and A_3 is the third harmonic, which is usually the strongest harmonic because it is the lowest-ordered odd harmonic. Thus the ratio of the fundamental to the highest harmonic in decibels can be written as

$$\left(\frac{A_3}{A_1}\right)_{\text{dB}} = 20 \log \left(\frac{A_3}{A_1}\right). \quad (5.20)$$

The results of (5.19) and (5.20) are shown in Table 5.1. From these values, it is found that the highest spurious response is approximately $9b$ dB below the fundamental component. If $b = 8$, the strongest spur is about 72 dB below the fundamental component. From this discussion, it appears that the maximum dynamic range is 9 dB per bit rather than 6 dB as predicted in (5.6).

It should be noted that the above discussion is based on the assumption that the time domain data are continuous. Under this situation, if the input signal is continuous, all the levels of the ADC are exercised and at each level there are many data points. This situation does not occur in the sampled data, especially for high-frequency signals.

Some simulated results are used to demonstrate the spur analysis. Figure 5.9 shows the results of the time and frequency domain of ADC with 3 to 5 bits. In these simulations, the input signal is a cosine wave with one complete cycle and 1024 points of data are obtained in the time domain. Since the data are real, only 512 frequency components are independent. However, in order to show the first few individual frequency components clearly, only the first 32 components are shown. In these figures, the fundamental frequency is at location 1, which is the highest. The third harmonic is 3, which is next to the fundamental since the second harmonic is zero. It appears that the third harmonic is close to the highest one and is approximately 9 dB per bit below the input signal. However, these special cases may not represent the worst situation in practice.

5.11 Further Discussion of Spur Amplitudes

In a digital receiver, the length of the FFT might be limited by the minimum pulse width. If the input frequency is relatively high compared with the sampling frequency, all the ADC output levels may not be exercised. Under this condition, the third frequency component might not be the highest one and the amplitude of the highest spur will be difficult to predict. In other words, the analysis discussed above may no longer be applicable. Figures 5.10 and 5.11 show the time

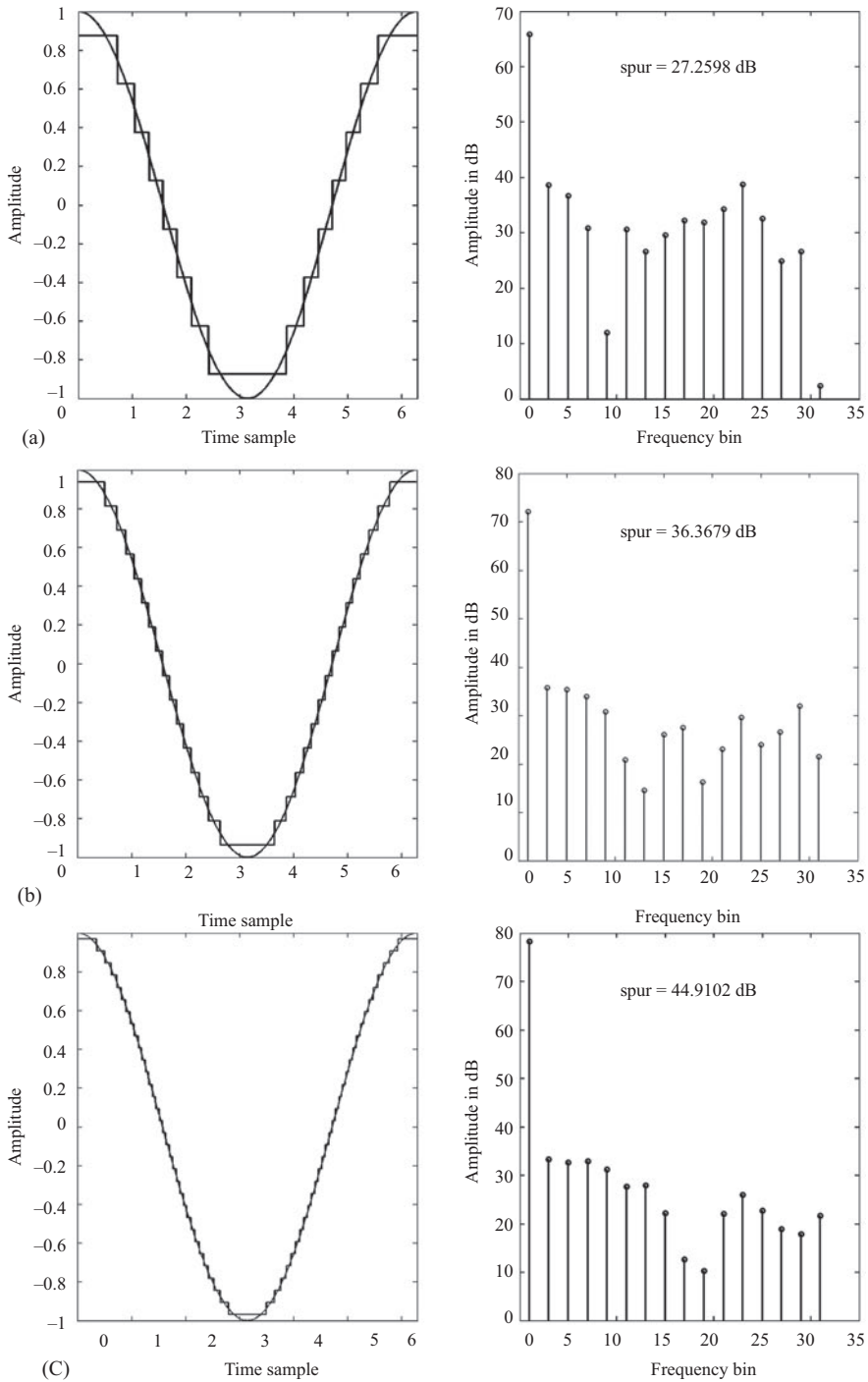


Figure 5.9 Time and frequency domains of ADC outputs: (a) 3 bits, (b) 4 bits, and (c) 5 bits.

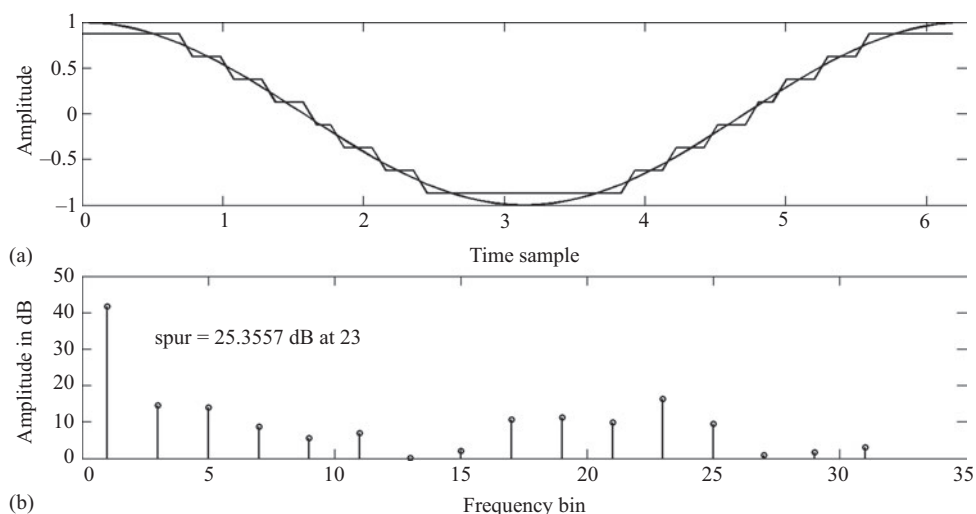


Figure 5.10 Outputs of a 3-bit ADC with no levels missing: (a) time domain and (b) frequency.

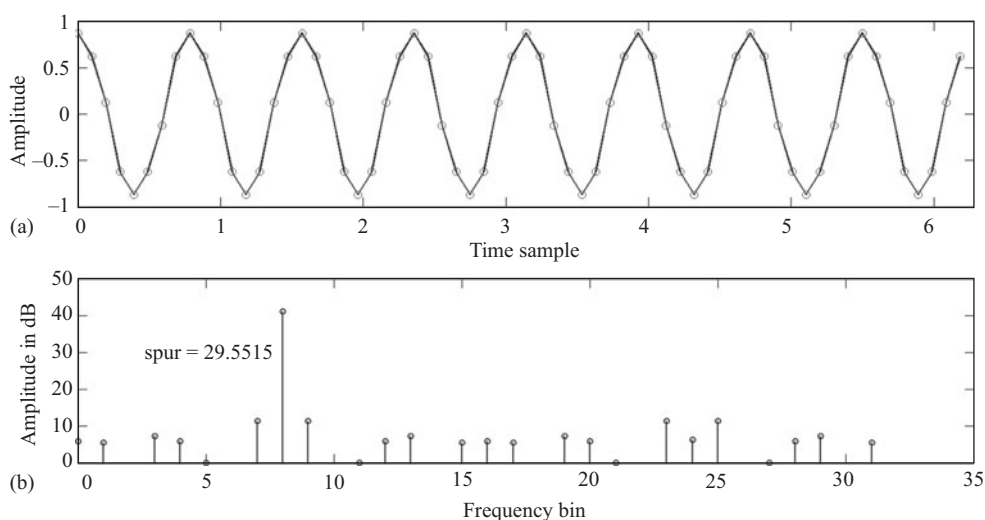


Figure 5.11 Outputs of a 3-bit ADC with many levels missing: (a) time domain and (b) frequency.

and frequency responses of a 3-bit ADC with 64 points of the FFT. In Figure 5.10(a), all the digitization levels are exercised, the frequency output has (Figure 5.10(b)) approximately 25 dB of dynamic range, but the highest spur is not the third harmonic of the fundamental. In Figure 5.11(a), many quantization levels are missing and the highest output in the frequency domain (Figure 5.11(b)) is about 30 dB, which is higher than the expected value of 27 dB.

In all the above simulated results, the input frequency equals one of the frequency bins of the FFT output. Under this condition, no sidelobes are generated. If the input frequency does not match one of the FFT output frequency bins, the output has high sidelobes and it is difficult

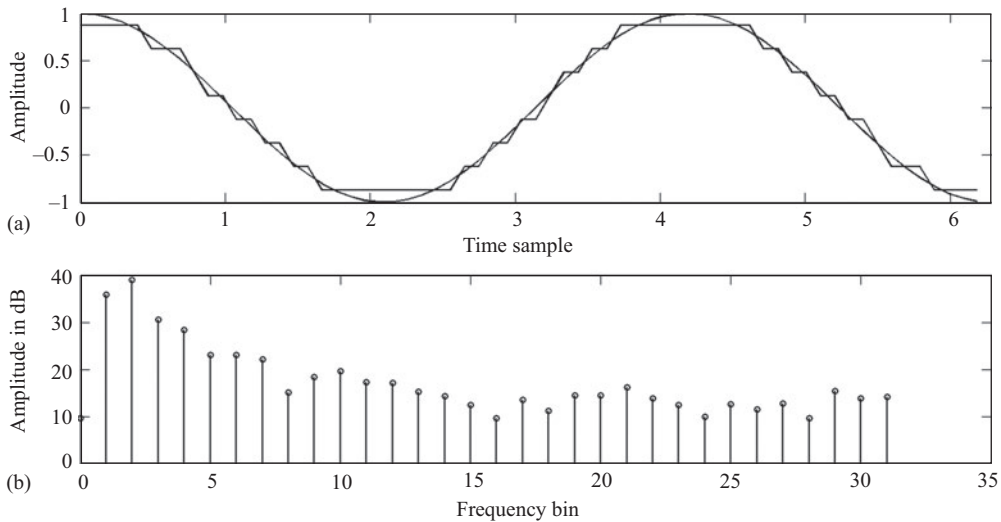


Figure 5.12 Output of 3-bit ADC with the input frequency between bins: (a) time domain and (b) frequency.

to distinguish the sidelobes from the spurs. Figure 5.12 shows these results for a 1.5-cycle sine wave. The frequency plot is shown in Figure 5.12(b), and it is difficult to separate the spurs from the sidelobes.

Finally, the spur levels are evaluated through simulated data. The input conditions are as follows. A Blackman window, which has a maximum sidelobe of -58 dB, is used to suppress the sidelobes. The length of the FFT is arbitrarily chosen to be 128 points and the ADCs have from 3 to 12 bits. For each ADC, 1000 sine waves with random frequencies are used as input. There are 64 frequency bins. The input frequency is limited from the second frequency bin to the 62nd frequency bin because if the input frequency is too close to the ends, it is difficult to find the proper dynamic range. The input signal passing through the Blackman filter is digitized and 128 data points are collected. First, take the absolute values of the FFT outputs, then take the logarithm of them. A peak is defined as a frequency component that is higher than both its adjacent neighbors and at least 1.5-dB higher than one of them to avoid a false peak. The dynamic range is defined as the distance between the highest and the second highest peaks. There are 1000 dynamic ranges corresponding to the 1000 input frequencies and the lowest value is considered the desired dynamic range. These results are shown in Figure 5.13 as the “*” marked curve. In this figure, it appears that the dynamic range increases by approximately 6 dB per bit from 3 to 7 bits. It starts to saturate at about 8 bits and approaches 58 dB, which is the dynamic range of the Blackman window.

A similar simulation is carried out for a 512-point FFT. The result is also shown in Figure 5.13 and is represented by the “o” curve. The result is quite similar to the 128-point FFT. However, from 3 to 8 bits the dynamic ranges are slightly higher. This is probably due to the finer frequency resolution to obtain the main peak. The maximum dynamic range is approaching 58 dB. This is one way to obtain an approximate result for the maximum dynamic range expected from a certain ADC.

The above discussion still represents the best results because the amplitude of the input signal matches the maximum level of the ADC. If the input signal does not match the full

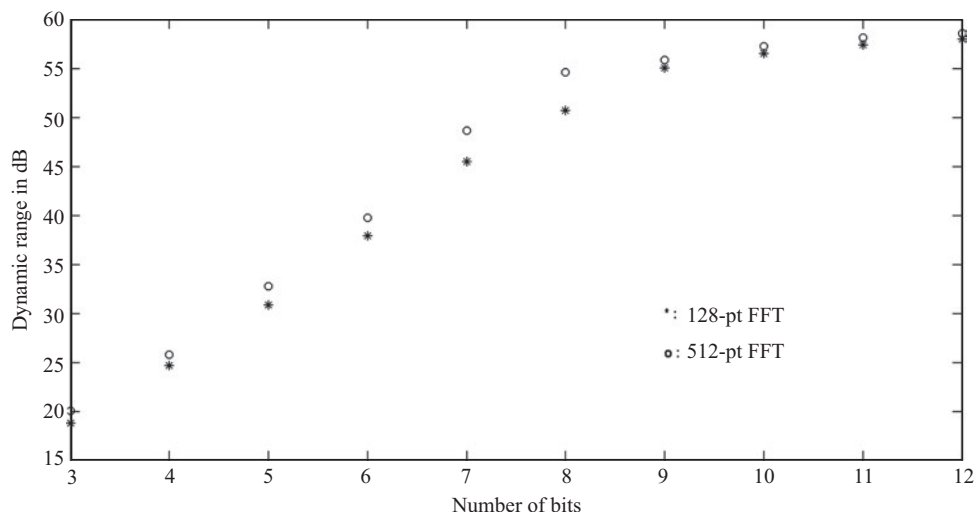


Figure 5.13 Dynamic range versus bits of an ADC.

scale of the ADC, the spurs will be higher than in the ideal case. Assume that the input signal is less than the optimum input, therefore not all the possible ADC output values are used. Under this condition, the number of bits actually used is less than the number of available bits and the spurious responses will be higher than the ideal case. If the signal is stronger than the maximum input level, the digitized output will show a saturation effect. As a result, the maximum spur level will increase as well.

5.12 Noise Effects in ADCs [26, 27]

The ADCs discussed in the previous sections are assumed to be ideal. However, the performance of almost all the ADCs is not perfect. For example, the quantization steps may not be uniform. Some steps are wider and some are narrower. In the extreme case, a certain quantization level is so narrow that it may never generate an output, and this is referred to as missing bits. The sampling window is not always stable and the window will jitter, and this will have an adverse effect. There is also noise in the ADC circuit. For some ADCs, even without an input signal, the LSB may toggle in a random manner. The effect of noise is discussed in this section.

Common sense tells us that noise will affect the sensitivity of a receiver. In many narrowband receivers (e.g., those used for communications), the noise level is kept as low as possible. In an EW receiver, the noise level is not the only concern, the dynamic range is also of concern. High sensitivity (low noise) usually means low dynamic range. This is true in a digital receiver as well.

In an ADC, the noise sometimes has a positive effect. For instance, noise may reduce the spurs generated from the coherent digitization error. Some of the spurs are generated from coherent error, but the noise is incoherent. When noise is added into the input signal, the digitization coherence is reduced, thus the spurs are usually reduced or may even disappear. Figure 5.14 shows this effect through simulated data. Figure 5.14(a) shows the FFT output of

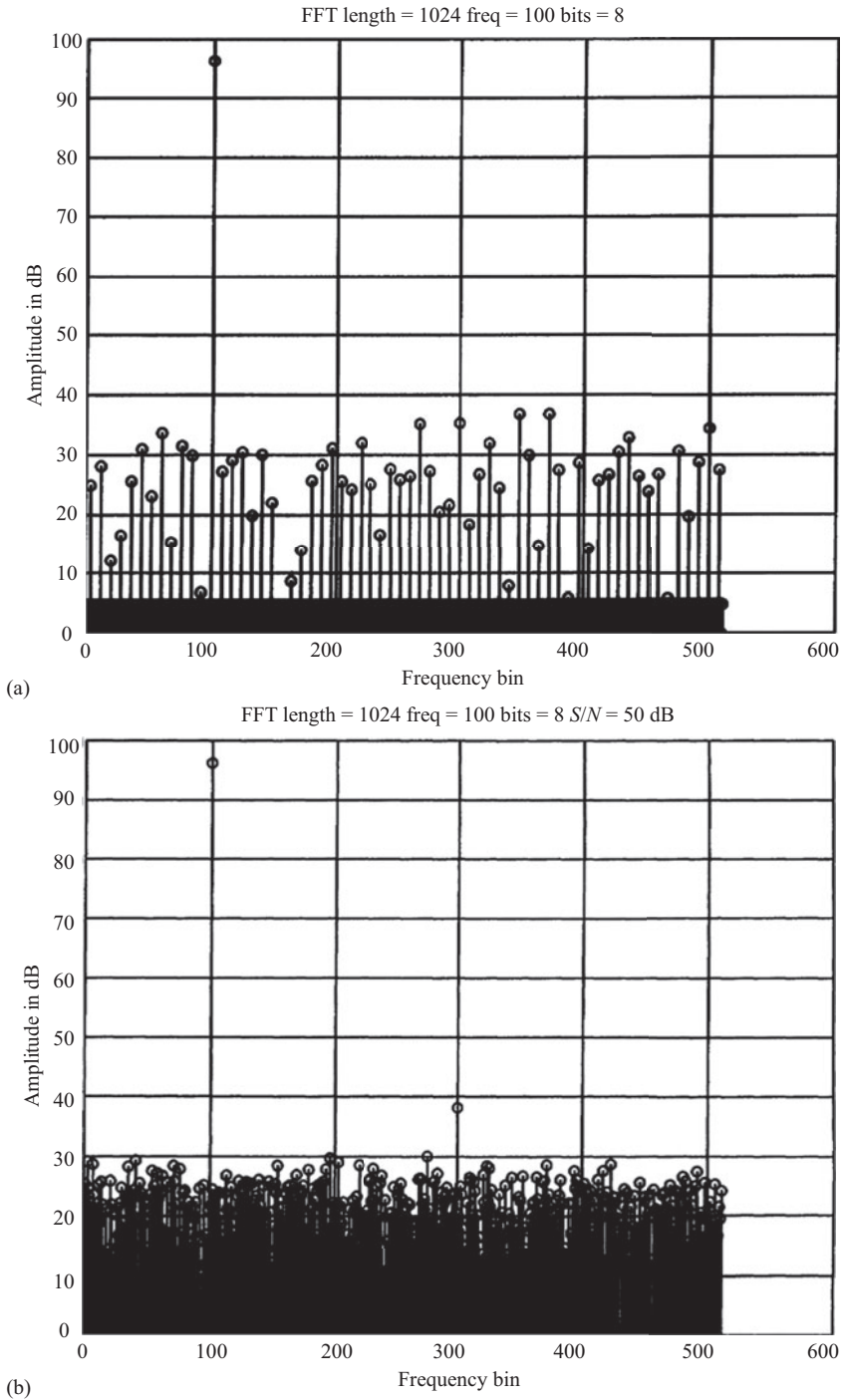


Figure 5.14 Noise effect on spurs and signals: (a) no noise and (b) noise.

the digitized sine wave without noise. The data contains two sine waves and the ADC has 8 bits. In order to keep this discussion simple, the frequencies of the two signals are selected to match two frequency bins. The first signal is at the 100th frequency bin and the second is at the 300th frequency bin with an amplitude 59 dB below the first one. From this figure it is difficult to determine the position of the second signal because of the high spur level. If a receiver is designed to process simultaneous signals, these spurs will limit the instantaneous dynamic range because the weak signal must be higher than the spurs to be detected.

Figure 5.14(b) uses the same input signals, but with added noise. In this case the S/N ratio is 50 dB with respect to the strong signal. The amplitude of the second signal is clearly shown in the figure.

In general, the noise will reduce most of the spurs. Since the noise is random, there is a chance under certain conditions that a spur may increase in amplitude. If a spur is generated entirely from the digitizing effect and there is no signal at the spur frequency, noise can disperse them easier than a true signal because the noise has a zero mean and its effect on the signal will be averaged out for a longer period of time. It can be easily shown that when the noise power increases, it will mask all the spurs, but it also reduces the sensitivity of the receiver. Thus a little noise will reduce the spur levels. It might improve the dynamic range slightly (at least there is no adverse effect), but more noise will reduce the sensitivity. A general analysis of noise effect might be difficult. The noise floor is discussed further later in this chapter.

5.13 Sampling Window Jittering Effect [30, 31]

As mentioned in Section 5.4, in a sample and hold circuit the sampling window (or aperture) has an unavoidable uncertainty period, which is called sampling window jittering. This effect can be illustrated as follows.

Consider the following case. If the input is a constant voltage, the jittering effect will not affect the output because the input does not change with time. If the input voltage changes rapidly, a small jittering of the sampling window will have a prominent effect on the output. If the input is sinusoidal, the ADC output also depends on the amplitude of the signal, because the larger the amplitude, the larger the change with respect to time. This effect is shown in Figure 5.15. Figure 5.15(a) shows a signal with low amplitude and low frequency and Figure 5.15(b) shows a signal with high amplitude and high frequency. If the ADC has the same amount of jitter, the signal with high amplitude and high frequency has a higher output change.

The jitter effect can be measured in terms of the S/N ratio. This effect can be studied as follows. If the input signal is

$$v(t) = A \sin(2\pi ft), \quad (5.21)$$

where A and f are the amplitude and frequency of the signal, respectively, then the derivative is

$$\frac{dv}{dt} = 2\pi f A \cos(2\pi ft). \quad (5.22)$$

The root mean square (rms) value of the derivative is

$$\left. \frac{dv}{dt} \right|_{\text{rms}} = \frac{2\pi f A}{\sqrt{2}}. \quad (5.23)$$

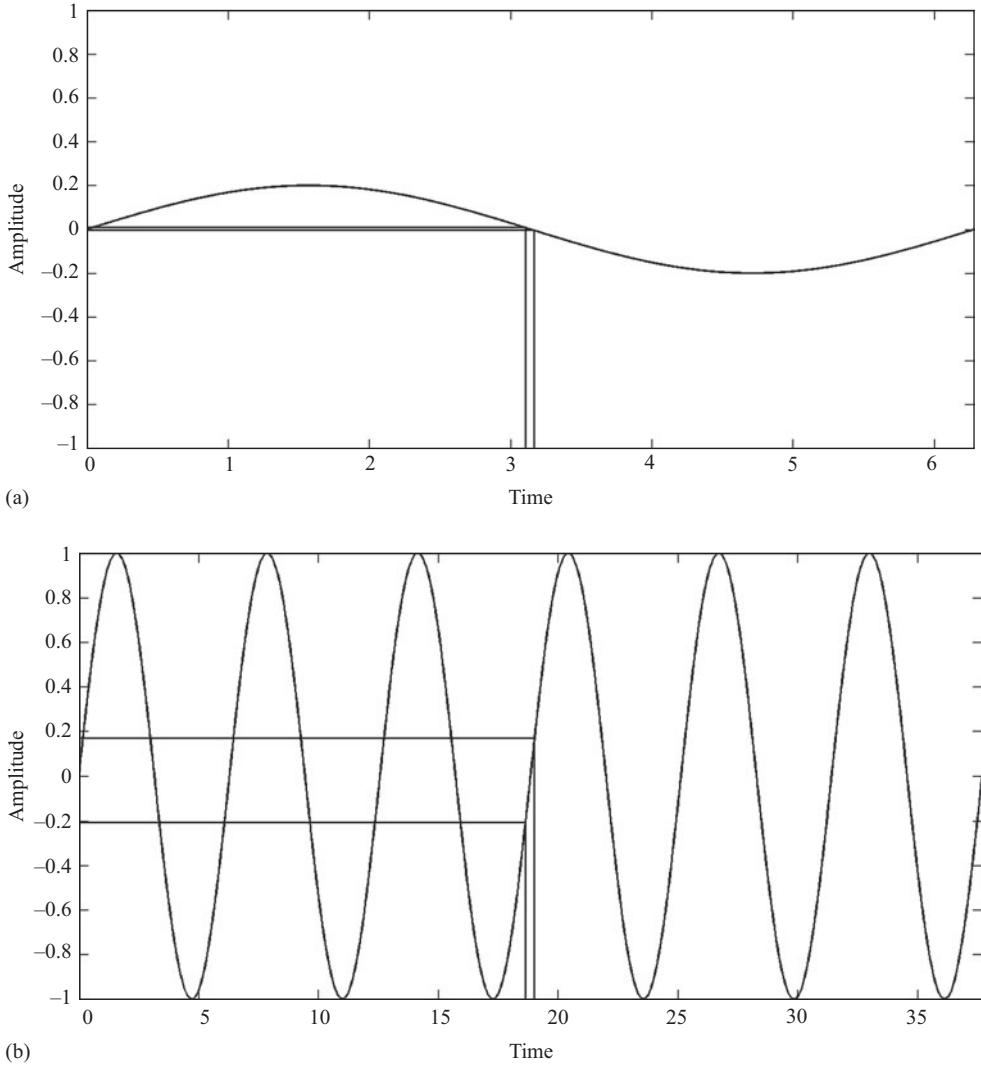


Figure 5.15 Sampling window jitter effect on signal output: (a) signal with low amplitude and low frequency and (b) signal with high amplitude and high frequency.

This relation can be used to relate the rms error voltage and rms aperture. This relation can be written as

$$\frac{\Delta V_{\text{rms}}}{t_a} = \frac{2\pi f A}{\sqrt{2}}, \quad (5.24)$$

where t_a represents the rms value jitter time. This equation can be written in terms of S/N as

$$\left(\frac{S}{N}\right)_{\text{dB}} = 20 \log \left[\frac{\frac{A}{\sqrt{2}}}{\Delta V_{\text{rms}}} \right] = 20 \log \left[\frac{1}{2\pi f t_a} \right], \quad (5.25)$$

where $A/\sqrt{2}$ is the input amplitude and ΔV_{rms} is considered as noise. It is assumed that jitter has a normal distribution with zero mean and a variance of σ^2 or ($\sigma = t_a$). Under this condition, (5.24) can be written as

$$\Delta V_{\text{rms}} = \frac{2A\pi f \sigma}{\sqrt{2}}. \quad (5.26)$$

If one considers only the jittering and the quantization noise, the signal-to-noise (S/N) ratio can be found from the sum of the two noise powers. The total noise N_{qj} is

$$N_{qj} = \frac{Q^2}{12} + \frac{(A2\pi f \sigma)^2}{2}. \quad (5.27)$$

The S/N ratio can be written as

$$\frac{S}{N} = \frac{A^2}{2N_{qj}} = \frac{A^2}{2 \left[\frac{Q^2}{12} + \frac{(2\pi f A \sigma)^2}{2} \right]}. \quad (5.28)$$

From (5.1), this equation can be written in decibels as

$$\frac{S}{N} = 10 \log \left[\frac{3 \cdot 2^{2b}}{2 + 3(2^b 2\pi f \sigma)^2} \right] \text{ dB}. \quad (5.29)$$

This result is plotted in Figure 5.16. In this figure, $b = 8$ and $t_a = \sigma = 2.5$ ps. This S/N ratio drops from about 50 dB at low frequency to about 36 dB at 1 GHz. If the S/N response of an ADC can be measured, this type of curve can be used to determine the sample window jittering time.

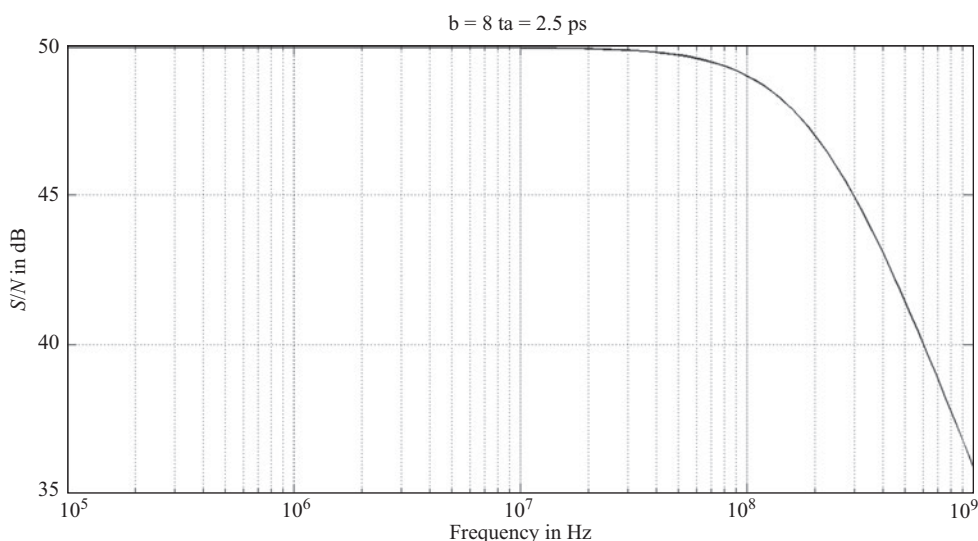


Figure 5.16 S/N ratio versus frequency of an ADC.

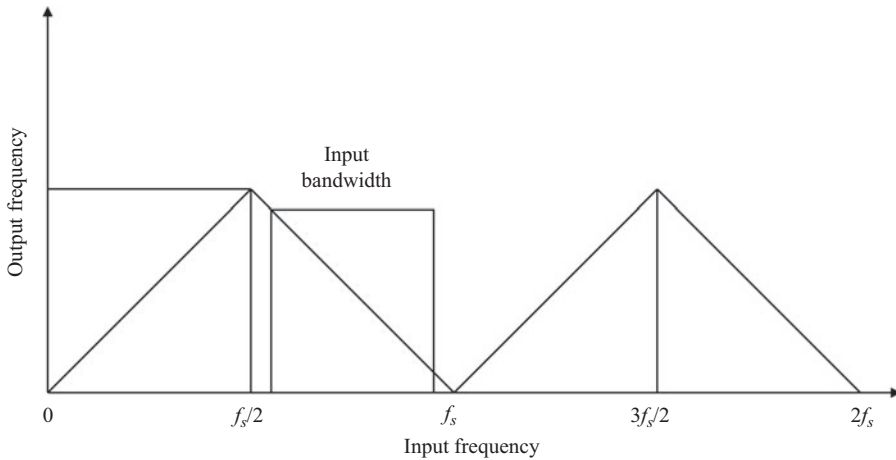


Figure 5.17 Input versus output frequency of band aliasing.

5.14 Requirements of ADCs

The performance of a receiver depends on the receiver design and the performance of the ADC. We briefly introduced some characteristics of ADCs important for digital receiver design. Readers interested in learning more about ADC technologies will find Kester [32] a useful information source. The receiver performance cannot surpass the performance of the ADC used in the receiver. The spur-free dynamic range of an ADC is defined as the ratio of the signal to the strongest spur in the ADC output. If the single-signal spur-free dynamic range is about 57 dB, one cannot expect to build a receiver using this ADC to obtain a single-signal dynamic range of more than 57 dB.

The RF input bandwidth of a receiver should be less than an octave to avoid an in-band second harmonic. For a wide-bandwidth receiver, it is usually impossible to achieve this design goal in the baseband from 0 to $f_s/2$, as shown in Figure 5.17. In an intermediate frequency band, such as from $f_s/2$ to f_s , this goal can be achieved. It is common practice to build the RF channel of a wideband receiver in the second frequency zone. Under this condition, the ADC sampling the RF must be able to accommodate the input frequency. Therefore, if the maximum sampling frequency of the ADC is f_s , the input frequency should also be able to reach f_s . If an ADC can operate at a sampling frequency of 3 GHz, the input frequency can also reach 3 GHz, thus it is suitable for wideband receiver operation. In order to achieve the performance goals listed in Table 2.2 in Chapter 2, it appears that an ADC with 10 bits and a sampling frequency above 2.5 GHz is needed.

5.15 Notation

After the brief discussion on the performance of ADCs, we are now ready to discuss the interface between ADCs and amplifiers and how to design an RF chain based on a given ADC specification. It should be noted that the equations used in this chapter are mixed with

- p_3 Third-order intermodulation spur generated at the output of the amplifier.
- P_i Input signal power to the amplifier.
- P_I Input signal power level when p_3 is at a desired noise level.
- P_o Output power level of the amplifier.
- P_s Power level at the input of the ADC to generate a full-scale output.
- P_{sn} Power level at the input of the ADC to generate a full-scale output under noisy conditions.
- Q One quantization level (in volts).
- Q_3 Third-order intercept point of the amplifier chain.
- V_n Voltage reduction caused by noise.
- V_s Maximum input voltage to the ADC without causing saturation.

There are a total of 25 quantities represented by different notations. However, many of the notations are commonly used by engineers in the receiver area.

5.16 Noise Figure and Third-Order Intercept Point [33–44]

After covering ADCs, we are now ready to describe how to design an amplifier chain to optimize the receiver performance based on the selection of the ADC. We start with a discussion on the definitions of a noise figure and a third-order intercept point of an amplifier. In the next section, determination of the noise figure and third-order intercept point of an amplifier chain will be presented.

The gain of an amplifier is defined as the ratio of output power to input power, which can be written as

$$G = \frac{P_o}{P_i}. \quad (5.30)$$

This equation can be expressed in decibel form by taking the logarithm on both sides of the equation and multiplying the result by 10. The result is

$$10 \log G = 10 \log \left(\frac{P_o}{P_i} \right)$$

or

$$G = P_o - P_i \text{ dB}. \quad (5.31)$$

Whenever equations are expressed in decibels, this operation applies.

The noise figure of a receiver is defined as

$$F = \frac{\text{actual receiver output noise}}{\text{ideal receiver output noise}} = \frac{\text{actual receiver output noise}}{\text{receiver input noise} \times G}, \quad (5.32)$$

where G is the gain of the amplifier. This definition will be used to find the overall receiver performance including the ADC.

The third-order intermodulation is a quantity related to the dynamic range of a device (e.g., amplifier and mixer) as well as the receiver. Assume that two signals are of the same amplitude with frequencies of f_1 and f_2 . If the two signals are increased to the saturation level

of the amplifier, two additional frequency components, $2f_1 - f_2$ and $2f_2 - f_1$, will appear; this is referred to as the third-order intermodulation. Figure 5.19 shows this concept.

The common approach to calculate the third-order intermodulation products is to use the third-order intercept point. The third-order intercept point can be obtained from the input versus output plot and the third-order intermodulation. This result is shown in Figure 5.20. The output versus input of the fundamental frequency is a straight line of unit slope. The third-order intermodulation versus input has a 3:1 slope. The third-order intercept point is the intersection of these two lines.

However, it is difficult to obtain the third-order intercept point accurately. Theoretically, when the third-order intermodulation products are at a very low level, the input signals increase by 1 dB and the intermodulation products increase by 3 dB. However, when data are taken experimentally, the 3:1 ratio is seldom realizable. Because of this difficulty, the third-order intercept point sometimes is obtained from one data point. In this approach the third-order intermodulation product is measured at a level near the noise floor. A straight line is drawn through this point with a slope of 3:1 to reach the third-order intercept point.

The third-order intermodulation product P_3 , as shown in Figure 5.20, is related to the input signal P_i through the linear relation as

$$\frac{y - Q_3}{x - (Q_3 - G)} = 3. \quad (5.33)$$

Substituting $x = P_i$ and $y = P_3$ into the above equation, the result is

$$P_3 = 3P_i - 2Q_3 + 3G \text{ dB}. \quad (5.34)$$

However, the input power P_i and the output power P_o are related by the G of the amplifier, which can be written as

$$P_o = P_i + G \text{ dB}. \quad (5.35)$$

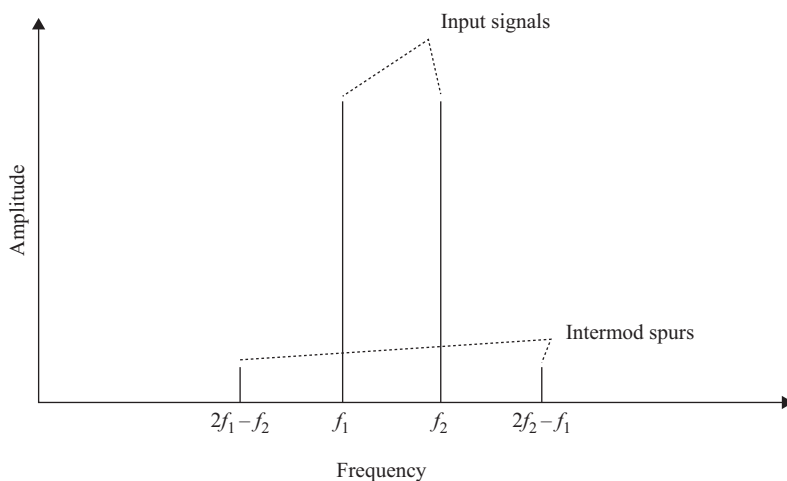


Figure 5.19 Third-order intermodulation products.

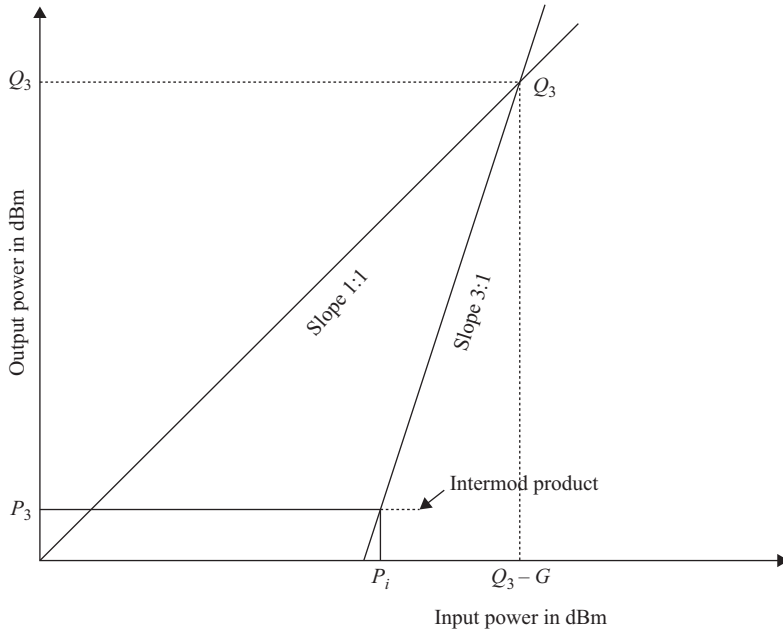


Figure 5.20 Third-order intercept point.

Using this relation in (5.34), the amplitude of the third-order intermodulation product is

$$P_3 = 3 \left(P_o - \frac{2}{3} Q_3 \right) \text{ dB.} \quad (5.36)$$

This relation is often used to determine the two-signal spur-free dynamic range. For example, in many receiver designs the maximum input level is defined as the level when the intermodulation products generated equal the noise level. Under this condition, the input power level is designated as P_I . The corresponding dynamic range is often called the two-signal spur-free dynamic range. This relationship is used in digital receiver designs to select the performance of the amplifier chain to match a given ADC and is presented in Section 5.20.

5.17 Characteristics of the Amplifiers in Cascade [1, 33–44]

In this section, the parameters of an amplifier chain are discussed. An amplifier chain is defined as several amplifiers that are connected in cascade. There are three important parameters in an amplifier chain: the noise figure, the gain, and the third-order intercept point. These parameters affect the performance of the receiver and are discussed briefly in this section. In the amplifier chain design, if one of the parameters is changed, the other two usually change also. There are many different ways to design an amplifier chain. The general rule is presented at the end of this section and some examples are provided.

In general, it is desirable to design an amplifier chain to have the lowest possible noise figure and highest possible third-order intercept point. The gain of the amplifier chain is

determined by the system being designed. This gain value is dependent on the characteristics of the ADC (or the crystal detector in an analog receiver) used at the end of the amplifier chain.

Derivation of the gain, the noise figure, and the third-order intercept point are given in Tsui [1]; only the results are presented here.

If several amplifiers are cascaded in an amplifier chain, the overall gain G can be written as

$$G = G_1 G_2 \dots G_n$$

or

$$G = G_1 + G_2 + \dots + G_n \text{ dB}, \quad (5.37)$$

where $G_1, G_2, \dots$ are the gains of individual components of the RF chain.

The overall noise figure F of the amplifiers connected in cascade can be written as

$$F = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \dots + \frac{F_n - 1}{G_1 G_2 \dots G_{n-1}}, \quad (5.38)$$

where $F_1, F_2, \dots$ are the noise figures of individual components of the RF chain. From this equation, one can see that when G_1 is very large, the overall noise figure is determined approximately by F_1 . In other words, all the microwave components (i.e., filters and mixers) with insertion losses used before the first amplifier have an adverse effect on the noise figure. All the components used after a high-gain amplifier have a minor effect on the overall noise figure.

The overall third-order intercept point can be calculated as

$$Q_3 = \frac{G}{\frac{G_1}{Q_{3,1}} + \frac{G_1 G_2}{Q_{3,2}} + \dots + \frac{G_1 G_2 \dots G_n}{Q_{3,n}}}, \quad (5.39)$$

where $Q_{3,1}, Q_{3,2}, \dots$ are the third-order intercept points of each individual component. The third-order intercept points of the amplifier and the mixer are often provided by the manufacturer. The value is in reference to the output of the device. From this equation, the effect of each component on the overall third-order intercept point is difficult to see. Some examples will be used to demonstrate the effect.

In the above equations, if the component is an amplifier, then G_i, F_i , and $Q_{3,i}$ are all given and the above equations can be used directly. If the component is passive (i.e., an attenuator or a filter), the three quantities are not given, but the insertion loss is. Under this condition, the gain and noise figure can be obtained from the insertion loss; that is, the gain equals the negative value of the insertion loss and the noise figure is equal to the insertion loss. Since a passive component usually does not have a nonlinear region, the third-order intercept point is very high. Thus a very large number (i.e., 100 dBm) can be assigned to a passive component. In general, such a high value will have little if any effect on the overall third-order intercept point.

A computer program (df7eq1.m) is provided in Appendix 5.D. This program calculates the overall gain, noise figure, and third-order intercept point. The calculations are based on (5.37) through (5.39).

Let us use an example to conclude this section. There are two amplifiers and a 3-dB attenuator, with their characteristics listed in Table 5.2. These three components are connected in different ways as shown in Figure 5.21. Figures 5.21(a) and 5.21(b) show two

Table 5.2 Amplifier and attenuator characteristics.

	Amplifier 1	Amplifier 2	Attenuator
Gain (dB)	15	15	−3
Noise figure (dB)	3	5	3
Third-order intercept point (dBm)	15	20	100

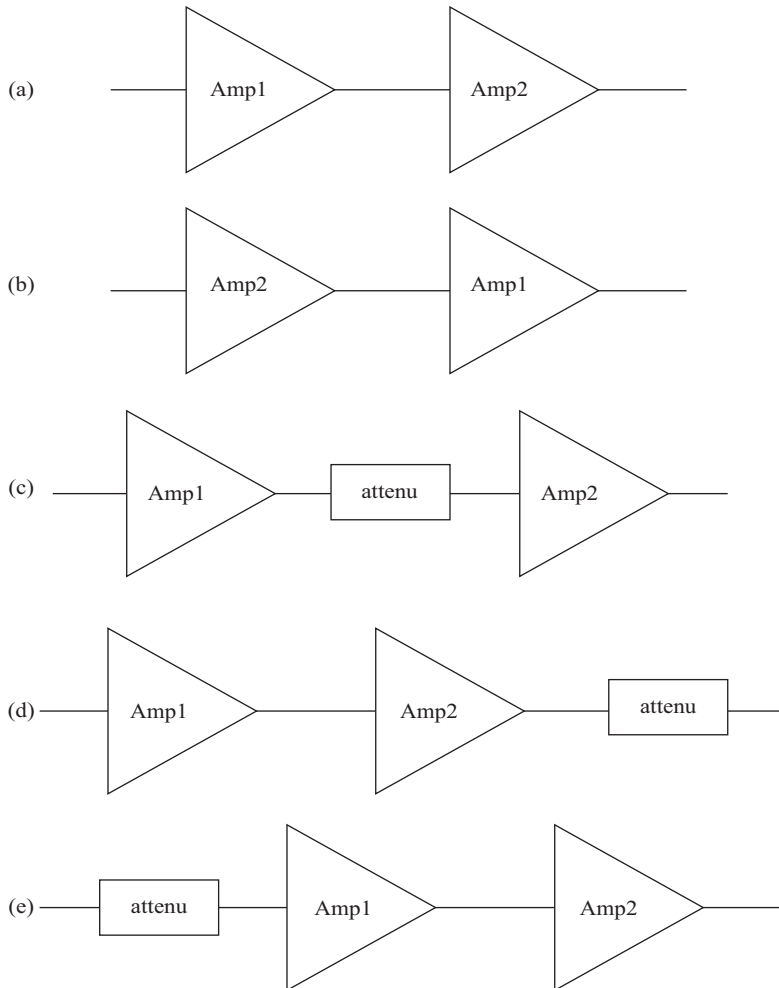


Figure 5.21 Different ways to cascade amplifiers (amp) and an attenuator (attenu):
 (a) amp1-amp2, (b) amp2-amp1, (c) amp1-attenu-amp2, (d) amp1-amp2-attenu,
 and (e) attenu-amp1-amp2.

amplifiers connected in cascade without the attenuator. In these two arrangements, the order of the two amplifiers is reversed. In Figure 5.21(c), the attenuator is placed between the two amplifiers. In Figure 5.21(d), the attenuator is placed at the output of the second amplifier. In Figure 5.21(e), the attenuator is placed at the input of the two amplifiers.

Table 5.3 Results of different amplifier–attenuator connections.

	Total gain (dB)	Overall noise figure (dB)	Third-order intercept point (dBm)
Figure 5.21(a)	30	3.146	19.586
Figure 5.21(b)	30	5.043	14.957
Figure 5.21(c)	27	3.351	19.219
Figure 5.21(d)	27	3.148	16.586
Figure 5.21(e)	27	6.146	19.586

Usually all the characteristics of amplifiers are given in decibels or decibels referred to 1 mW, but in (5.37) through (5.39), all the values represent actual power. Before the equations are used, the given values in decibels and decibels referred to 1 mW must be converted to a ratio or power through the following relation:

$$G_{\text{dB}} = 10 \log (G)$$

or

$$G = 10^{G_{\text{dB}}/10}, \quad (5.40)$$

where G_{dB} represents the gain in logarithmic scale. This equation can be applied to quantities other than gain, such as the noise figure. The calculated results of the three arrangements are in power ratio or watts. The results are converted back into decibels or decibels referred to 1 mW. All these conversion operations are included in the computer program in Appendix 5D (df7eq1.m).

The calculated results are listed in Table 5.3. This example is helpful in designing a receiver with given amplifiers, and more discussion is included in Section 5.23. In order to show the effect of different connections, the results are kept to three decimal points, although this kind of accuracy does not have much meaning in a receiver design because the specifications of components do not have such high accuracy. The following important factors should be noted.

1. Comparing the first two cases, one can see that both cases have the same gain. However, the arrangement in Figure 5.21(a) has a lower noise figure and a higher third-order intercept point, which is the desirable result. In general, RF amplifiers can be briefly divided into two groups: one with low noise and the other with high power. A low-noise amplifier usually has a low third-order intercept point and a high-power one usually has a high noise figure. In a cascade system, the low-noise amplifier should be placed at the beginning of the chain and the high-power one at the end of the chain. From this comparison, one can see that the noise is dominated by the first amplifier and the third-order intercept point is dominated by the last amplifier.
2. When two amplifiers are connected in cascade as shown in Figure 5.21(a), the noise figure is higher than the first amplifier and the intercept point is lower than the second amplifier.
3. When an attenuator is inserted in the amplifier chain, no matter where it is placed, the overall gain is decreased by the insertion loss of the attenuator.

4. When the attenuator is placed between the two amplifiers as shown in Figure 5.21(c), both the noise figure and the third-order intercept point degrade slightly in comparison with Figure 5.21(a).
5. When the attenuator is placed at the end of the amplifier chain as shown in Figure 5.21(d), the noise figure degrades very little, but the third-order intercept point degrades 3 dB in comparison with Figure 5.21(a).
6. If the attenuator is placed in front of the amplifier chain as shown in Figure 5.21(e), the noise figure degrades 3 dB, but the intercept point does not change.

From this simple example, one can generally say that moving the attenuator to the front of the amplifier chain will degrade the noise figure more, and if the attenuator is moved toward the end, the third-order intercept point will suffer. In general, the attenuator is seldom placed at the input of the amplifier chain because the sensitivity will be reduced by the insertion loss.

This simple example also reveals an important trade-off in receiver design. Assume that the amplifiers and attenuators are connected in the proper order. Under this condition, when the noise figure is low, the third-order intercept is also low. That is why a receiver with high sensitivity usually has low dynamic range. When the dynamic range is high, the sensitivity is low. In actual receiver design, there are usually more than two amplifiers and the attenuator is divided into several separate ones and placed at different locations to obtain the desired results.

One can see that if Figure 5.21(a) is used, it provides the highest gain, the lowest noise figure, and the highest third-order intercept point. However, this arrangement might not be used since the gain of the amplifier chain must be a specific value. Too high a gain value can produce an adverse effect on receiver performance. This subject is discussed in Section 5.20.

5.18 ADC [25]

As mentioned in Section 5.2, an ADC is selected first in a digital receiver. ADC performance was discussed earlier in this chapter. In this section the results are presented again, but these results will be used to determine the specification of the amplifier chain in front of it. The two important parameters of an ADC are the quantization noise and the maximum input power without saturating the device.

The quantization power (N_b) is obtained in (5.7) where the impedance of the system is considered unity. The result is rewritten here as

$$N_b = \frac{Q^2}{12R}, \quad (5.41)$$

where Q is the size of the quantization level and R is the input impedance of the ADC. In this equation, the impedance is assumed to be R rather than unity.

The maximum voltage V_s of a sinusoidal wave that can be applied to the input of the ADC without causing saturation is

$$V_s = 2^{(b-1)}Q, \quad (5.42)$$

where b is the number of bits. The maximum power P_s is related to the maximum voltage V_s by

$$P_s = \frac{V_s^2}{2R} = \frac{2^{2(b-1)}}{2R}Q^2. \quad (5.43)$$

The S/N ratio can be obtained from (5.41) and (5.43) as

$$\frac{S}{N} = \frac{P_s}{N_b} = \frac{3}{2} 2^{2b}$$

or

$$\frac{S}{N} = P_s - N_b = 6b + 1.76 \text{ dB}. \quad (5.44)$$

In this case, the only noise considered is the quantization noise.

This ADC is used to collect digitized data. If the sampling frequency is f_s and an N -point FFT is performed on the digitized data, the processing bandwidth B_v can be found as follows. If the input is complex data, the maximum input bandwidth is f_s and there are N output channels. If the input is real data, the maximum input bandwidth is $f_s/2$ and there are $N/2$ independent output channels. Therefore in either case the processing bandwidth B_v is

$$B_v = \frac{f_s}{N}. \quad (5.45)$$

In a digital receiver, this processing bandwidth is also the RF resolution bandwidth.

5.19 Noise Figure of an Amplifier and ADC Combination [45]

In this section, the overall noise figure of an amplifier and ADC combination is determined. To find the noise figure of an amplifier–ADC combination, the ADC can be considered as an additional noise source. As shown in Figure 5.18, the noise at the input of the amplifier is N_i . This noise can be found as

$$N_i = N_1 + B_R \text{ dBm}, \quad (5.46)$$

where N_1 ($= -174$ dBm) is the noise level at room temperature with unity bandwidth and B_R is the RF bandwidth. The output noise of the amplifier is N_o , which can be written as

$$N_o = N_i + F + G \text{ dBm}, \quad (5.47)$$

where F and G are the noise figure and the gain of the amplifier, respectively.

The noise N_s at the output of the ADC is the sum of the amplifier output noise N_o and quantization noise N_b , assuming the noise is band limited and no noise is folded into the baseband through the ADC. From the definition of the noise figure given in (5.32), the overall noise figure F_s can be written as

$$F_s = \frac{N_s}{GB_R N_1} = \frac{N_s}{GN_i} = \frac{N_o + N_b}{GN_i} = F + \frac{N_b}{GN_i}$$

or

$$F_s = N_s - G - B_R - N_1 \text{ dB}, \quad (5.48)$$

where F is the noise figure of the amplifier. This equation is obtained because N_s is the actual noise output of the amplifier–ADC combination and GN_i is the noise output of an ideal system. The noise generated by an ideal system is the input noise multiplied by the gain

of the amplifier, and there is no noise contributed by the system. The overall noise figure degrades by the quantity N_b/GN_i .

In order to simplify the operation, the amplifier output noise is measured in terms of quantization noise. Let us define two quantities M and M' as

$$M \equiv \frac{N_o}{N_b} \quad \text{and} \quad M' \equiv M + 1. \quad (5.49)$$

Thus the noise figure in (5.48) can be written as

$$F_s = \frac{N_o(1 + \frac{1}{M})}{GN_i} = \frac{N_o(1 + M)}{GMN_i} = \frac{FM'}{M}$$

or

$$F_s = F + M' - M \text{ dB}. \quad (5.50)$$

It should be noted that in the decibel scale, $M' = 10 \log(M + 1)$.

5.20 Amplifier and ADC Interface [45, 46]

So far, the amplifier and the ADC have been discussed separately. Only the noise figure of the combined system has been calculated. In this section, the output of the amplifier will be made to match the input of the ADC. The meaning of match is twofold. First, at a certain input level the third-order intermodulation equals the noise level. Second, the amplifier amplifies this input signal to the maximum allowable signal level of the ADC. From these relations, the required gain and third-order intercept point of the amplifier can be obtained.

First, let us choose an amplitude of the third-order intermodulation output p_3 from the amplifier to match the noise level. The noise level is defined in the processing band as $N_s B_v / B_R$. Expressed in decibels and applying the relations in (5.48) and (5.50), the result can be written as

$$P_3 = N_s - B_R + B_v = N_1 + G + F + B_v + M' - M \text{ dBm}. \quad (5.51)$$

Under this condition, the input power P_i is designed as P_I , which is a special input level. Now, let us find the third-order intercept point Q_3 in terms of this input level. From (5.34), the input P_I can be written as

$$P_I = \frac{P_3 + 2Q_3 - 3G}{3}. \quad (5.52)$$

Substituting P_3 from (5.51), this level can be written as

$$P_I = \frac{2Q_3 + N_1 - 2G + F + B_v + M' - M}{3} \text{ dBm}$$

or

$$Q_3 = \frac{3P_I - N_1 + 2G - F - B_v - M' + M}{2} \text{ dBm}. \quad (5.53)$$

This Q_3 is the required overall third-order intercept point of the amplifier with input P_I to produce the third-order intermodulation level to match the noise level in bandwidth B_v .

This input power P_I after the amplifier should equal the maximum power level allowed to the input of the ADC without reaching saturation. The voltage is V_s , shown in (5.42). However, there is noise at the output of the amplifier. This noise added to the signal can cause the ADC to reach saturation. Thus, in considering the maximum allowable voltage, this noise power should be taken into account. The noise power at the output of the amplifier N_o will reduce the maximum input signal allowable to the ADC. To allow this adjustment, the maximum input power to the ADC is arbitrarily reduced by three standard deviations of N_o . The corresponding voltage is

$$V_n = \sqrt{3N_o R}. \quad (5.54)$$

The maximum allowable ADC power considering the noise reduction is

$$P_{sn} = \frac{(V_s - V_n)^2}{2R} = \frac{[2^{(b-1)}Q - V_n]^2}{2R}. \quad (5.55)$$

This result represents the maximum power of a single sinusoidal input.

In an evaluation of the third-order intermodulation products, two signals are required. The amplitude of each of the signals must be one-half of the amplitude of the voltage $(V_s - V_n)$. Expressed in terms of power, there is a factor 4, which corresponds to 6 in the logarithmic expression. Therefore the input power P_I is related to the maximum allowable power as

$$(P_I + 6) + G = P_{sn}. \quad (5.56)$$

Combining this equation with the conditions in (5.54) and (5.55) provides the desired gain G . The dynamic range can be found as

$$DR = P_I + G - P_3 \text{ dB}, \quad (5.57)$$

where the third-order intermodulation product P_3 equals the noise floor as described in (5.51).

It should be emphasized here that the third-order intercept point is the minimum required value. If the amplifier chain has a Q_3 greater than the calculated value, it does not cause any adverse effect. The gain calculated is the optimum value. If the gain of the amplifier chain is greater than the calculated value, the amplifier will drive the ADC into saturation and cause the generation of spurs.

5.21 The Meaning of M and M'

It should be noted that M and M' are used to represent both the values and their logarithmic forms. Whenever they are used in logarithmic form, the unit dB is included at the end of the equation.

The result of (5.50) can be rewritten here as

$$F_s = \frac{FM'}{M} = F \frac{M + 1}{M}. \quad (5.58)$$

If $M = 1$, it means that the quantization noise N_b equals the amplifier output noise. Under this condition, the system noise figure is two times the amplifier noise figure, or the noise figure is degraded by 3 dB. The larger the M value, the less the degradation of the noise figure.

If $M < 1$, from the above equation one can see that the system noise figure will be high, which means the quantization noise dominates the noise figure. This is undesirable because the sensitivity of the receiver will suffer. In order to increase the value of M , the gain of the amplifier must be high.

If $M = 9$, then $M' = 10$ and $M' \text{ (dB)} - M \text{ (dB)} = 10 \log(10) - 10 \log(9) = 10 - 9.54 = 0.46 \text{ dB}$, which implies the system noise figure will be degraded by 0.46 dB. Under this condition, the amplifier output noise power is nine times the quantization noise. As M increases to a very large value, the difference $M' \text{ (dB)} - M \text{ (dB)}$ will become very small. Under this condition, the noise figure of the receiver approaches the noise figure of the amplifier and the contribution of the quantization noise is negligible.

5.22 Computer Program and Results

In previous sections many equations are used to describe the RF chain design, but it might be difficult to use them in practice. In order to show how these equations are used, a design example is provided. A computer program (df7eq2.m) is listed in Appendix 5.E. In this computer program, the parameters in the following example are used as input. However, the main purpose of this program is to illustrate the design of other amplifier chains if the input data are available.

In this example, two parameters are needed from the amplifier: the noise figure F and the RF bandwidth B_R . The noise figure of the first amplifier can be used as the initial value in the program. If an amplifier chain must be used to obtain the proper gain and third-order intercept point, the noise figure of the amplifier chain is usually higher than the noise figure of the first amplifier. If this situation occurs, the new noise figure of the amplifier chain should be used in the program to make the necessary adjustment.

In an analog receiver, the resolution bandwidth (i.e., the bandwidth of the fine frequency filter) is used to find the noise floor and the sensitivity in the receiver. The overall bandwidth plays no important role in determining the sensitivity. In a digital receiver, the bandwidth is determined by the FFT length and the frequency resolution bandwidth equals the video bandwidth B_v . Under this condition, one may deduce that the RF bandwidth should not play any role in determining the receiver sensitivity. However, this is not true because the digitization noise of the ADC is added to the noise in the bandwidth B_R instead of being added to the noise in processing bandwidth B_v .

Let us use an example to illustrate the application of the computer program listed in Appendix 5.E. Assume the amplifier has the following specifications:

$$B_R = 30 \text{ MHz (RF bandwidth)}$$

$$F = 3.3 \text{ dB (amplifier noise figure)}.$$

The ADC has the following parameters:

$$b = 8 \text{ bits (number of bits)}$$

$$V_s = 270 \text{ mV (maximum allowable voltage to ADC)}$$

$$f_s = 250 \text{ MHz (sampling frequency)}$$

$$R = 50 \, \Omega \text{ (ADC input impedance)}.$$

The FFT is specified as follows:

$$N = 1,024 \text{ points}$$

$$\text{Maximum unambiguous input bandwidth} = f_s/2 = 125 \text{ MHz.}$$

In this example, the bandwidth is limited by the input amplifier: $B_v = f_s/N = 244 \text{ kHz}$ or $B_v \text{ (dB)} = 53.87$. This result is obtained from (5.45).

The maximum power P_s can be found from (5.43), where the value of R is used. In using the computer program in the appendix, one must pay attention to the units. The voltage is given in millivolts and the power is in milliwatts.

The input to the program is a range of M (defined in (5.49)) values. The outputs are gain, third-order intercept point, noise figure, dynamic range, and the input noise-to-quantization noise voltage ratio (N_v/Q) with each given an M value. The dynamic range discussed here is the two-tone spur-free dynamic range as discussed in Section 2.16. The N_v/Q can provide a general idea of how the quantization levels are filled with noise.

The results are shown in Table 5.4. It should be noted that the P_3 value is also the noise floor after FFT processing. The performance of the receiver is determined by $F_s \text{ (dB)}$ and $DR \text{ (dB)}$. In this table the value of M is arbitrarily chosen as 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, and 516. As discussed in Section 5.20, a small value of M will cause a high noise figure. For $M = 1$, the noise figure will degrade about 3 dB. In order to show a trend, some small values of M are chosen in Table 5.4.

A designer should pick the desired receiver performance (i.e., the noise figure and dynamic range). The gain in column 2 of Table 5.4 is the desired value. This means that in order to obtain the desired performance, the gain must be adjusted to the listed value. On the other hand, the third-order intercept point represents the minimum required values. If Q_3 is less than the listed value, the dynamic range of the receiver will be less than the listed value and the third-order intermodulation products will be at the lower limit of the dynamic range. If Q_3 is larger than the listed value, the dynamic range will equal the listed value. Under this condition, the noise floor is the lower limit of the dynamic range. When Q_3 equals the listed value, the intermodulation products will equal the noise floors and both become the lower limit of the dynamic range.

Table 5.4 Calculated performance of an rf chain.

M	$G \text{ (dB)}$	$Q_3 \text{ (dBm)}$	$P_3 \text{ (dBm)}$	$F_s \text{ (dB)}$	$DR \text{ (dB)}$	N_3/Q
0.25	38.78	24.41	-71.06	10.29	63.64	0.21
0.5	41.79	24.02	-70.27	8.07	62.86	0.29
1	44.80	23.38	-69.02	6.31	61.60	0.42
2	47.81	22.46	-67.26	5.06	59.81	0.59
4	50.82	21.31	-65.04	4.27	57.57	0.83
8	53.83	19.97	-62.49	3.81	54.97	1.18
16	56.84	18.51	-59.72	3.56	52.15	1.66
32	59.85	16.94	-56.84	3.43	49.19	2.35
64	62.86	15.29	-53.90	3.37	46.13	3.33
128	65.87	13.54	-50.92	3.33	42.98	4.71
256	68.88	11.68	-47.93	3.32	39.74	6.66
512	71.89	9.63	-44.93	3.31	36.37	9.42

5.23 Design Example

Let us use actual hardware to design a receiver with the help of Table 5.4. The ADC used in this design is the Tektronix TKAD20C, which has the performance listed in the last section. Thus the results listed in Table 5.4 can be used directly. There are two available amplifiers. Their characteristics are listed in Table 5.5.

If these amplifiers are connected in cascade, the overall gain is 71 dB. From the results shown in Table 5.4, when the gain is at about 71 dB and the noise figure of the system is 3.3 dB, the minimum required Q_3 is 9.6 dBm and the dynamic range is 36 dB. This dynamic range appears low.

The noise figure and dynamic range versus M is plotted in Figure 5.22. It is interesting to note that the noise figure does not change much when $M > 16$, but the dynamic range degrades rapidly with increasing M . For this experiment, the dynamic range of about 52 dB is selected (for $M = 16$). Under this condition, the noise figure will degrade about 0.26 dB ($3.56 - 3.3$). The gain of the receiver should be about 57 dB, the minimum required Q_3 is about 18.5 dBm, and the dynamic range is 52.1 dB.

In order to achieve this gain, 14 dB of attenuation must be inserted in the amplifier chain to reduce the available gain. In order to keep this design procedure simple, let us use only

Table 5.5 Amplifier characteristics.

Amplifier	G (dB)	NF (dB)	Q_3 (dBm)
1	42	3.3	12
2	29	4.0	33

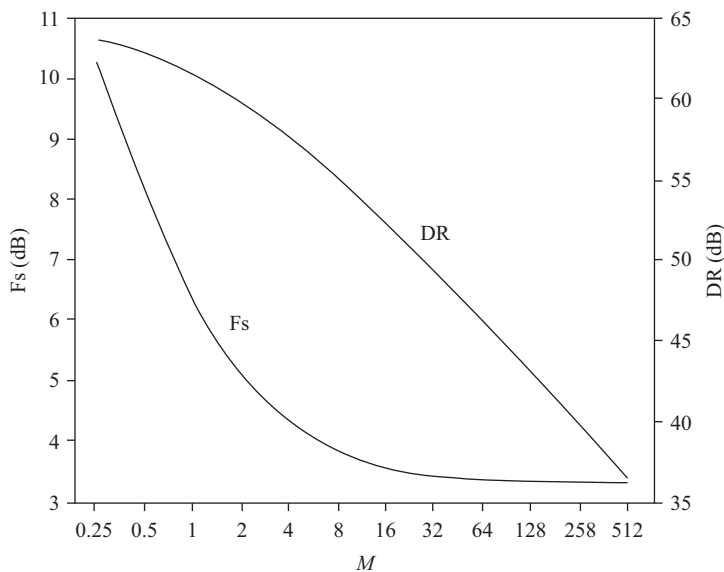


Figure 5.22 Noise figure and dynamic range versus M .

Table 5.6 Results of connecting amplifiers in two different ways.

Connections	G	F	Q_3
Case 1	57	3.3	26.0
Case 2	57	3.3	18.4

one 14-dB attenuator rather than splitting it into two attenuators. There are two possible ways to connect the two amplifiers. Case 1 is to place the attenuator between the two amplifiers and case 2 is to place the attenuator at the end of the second amplifier. The results can be obtained through the program in Appendix 5.D (df7eq1.m) and are listed in Table 5.6.

From these results, it appears that both cases fulfill the amplifier requirement. Interestingly, both cases have practically the same noise figure. Since the amplifier noise figure does not change from the first amplifier, the calculated system noise figure need not be modified. Otherwise, the new noise figure should be used to repeat the calculation again to generate new performance data.

In general, case 1 should be used because it provides a large margin in the Q_3 requirement, while in case 2 the Q_3 barely meets the minimum requirement. However, in this example case 2 is used because one can demonstrate that the third-order intermodulation products are very close to the noise floor. This example demonstrates that with a minimum required Q_3 value, both the noise floor and the third-order intermodulation products can be the lower limit of the dynamic range. If case 1 is selected, the intermodulation products will be much lower than the noise floor and one will not see the third-order intermodulation products.

5.24 Experimental Results [46]

The experimental setup is shown in Figure 5.23. A 30-MHz (20–50 MHz) bandpass filter was placed in front of the amplifier chain to eliminate possible stray signals in the environment and limit the noise bandwidth. The noise figure was measured after the filter. The two amplifiers mentioned in the last section were connected in cascade and a variable attenuator was added at the output of the amplifier. The variable attenuator can provide adjustment in the amplifier chain. After the attenuator, the Tektronix TKAD20C ADC was used to collect data. A 1024-point FFT was used to analyze the data.

5.24.1 Noise Figure Measurement

First, the noise figure of the system was measured. This test was performed using two different techniques. In the first technique, the input of the amplifier chain was terminated with a 50 Ω resistor. The noise power was computed in the frequency domain from 20 to 50 MHz by averaging five realizations of a 1024-point magnitude spectrum.

In the second technique, a full-scale sine wave at 36 MHz was injected into the amplifier and the noise power was again measured in the frequency domain. Once the value of N_s is obtained, the noise figure can be obtained from (5.48). The results are shown in Figure 5.24. This figure contains three sets of data: the theoretically calculated one, the measured one with input terminated, and the measured one with a signal. The noise figure measured with an input signal is very close to the theoretical one.

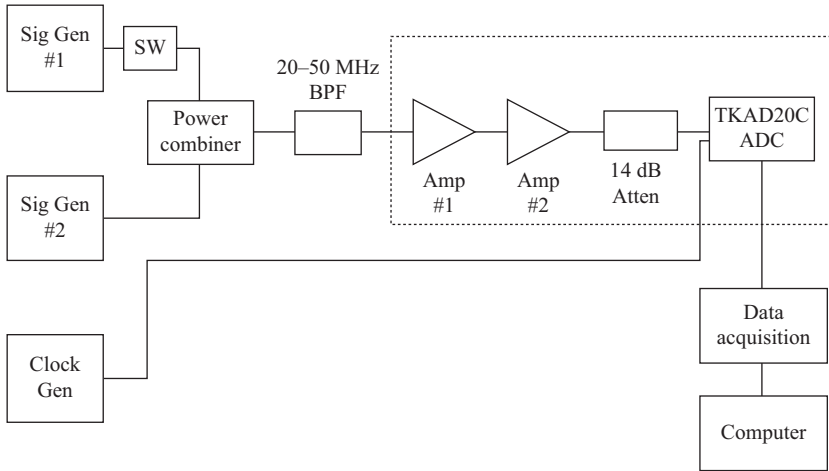


Figure 5.23 Amplifier chain experimental setup.

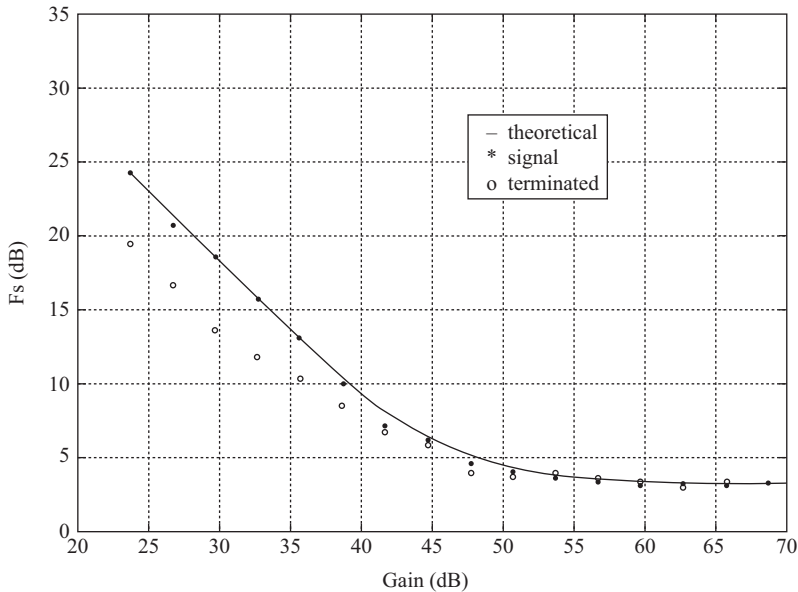


Figure 5.24 Noise figure versus gain.

The noise figure measured with the terminated input matches the theoretical one very well at high gain, but it falls below the theoretical one at low gain values. This discrepancy can be explained as follows. At low gain values, the noise from the amplifier fills only a small part of the first quantization level. Since the quantization noise model is based on the premise that the input signal is uniformly distributed across a given quantization level, the uniform distribution is no longer valid. When a full-scale signal is applied, the noise model is valid and the result matches the calculated one.

From this experiment, it appears that the noise figure of the receiver is signal dependent when the gain of the receiver is low. This phenomenon is caused by the nonlinear effect of the ADC. However, in most receiver designs, usually the gain is high enough and the receiver noise figure matches the theoretical value and is signal independent.

5.24.2 Dynamic Range Test

The second test is to find the dynamic range of the receiver. In this test, the goal is to detect the third-order intermodulation products. Two input frequencies at 36 and 41 MHz are kept at the same amplitude. The power of each sine wave was set at -7.3 dBm, which was 6-dB below the full-scale value of the ADC (-1.3 dBm). If the noise effect is considered, the maximum input power should be reduced from -7.3 to -7.5 dBm. Since this difference (0.2 dBm) is within the measurement error, the noise effect is neglected in this experiment. A time domain plot is shown in Figure 5.25. It shows clearly that the signals do not saturate the ADC.

In order to demonstrate the noise floor and the third-order intermodulation products, three attenuation values are selected and are listed in Table 5.7.

The spectrum plot is shown in Figure 5.26. These results are of five realizations of the magnitude spectrum that are averaged. In Figure 5.26(a), the noise floor was measured to

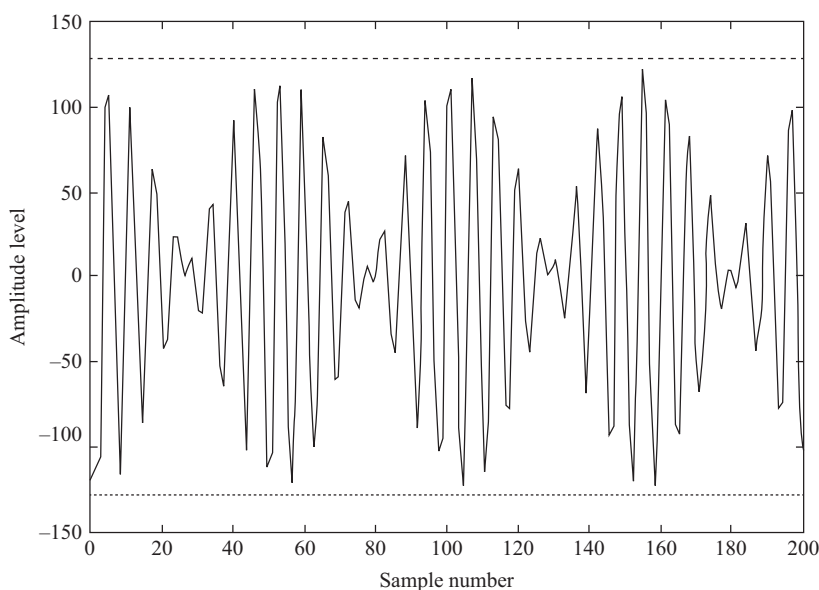


Figure 5.25 Time domain plot of two signals of the same amplitude.

Table 5.7 Gain and third-order intercept point with different attenuation values.

Figure	Attenuation (dB)	Gain (dB)	Q_3 (dBm)
5.26(a)	17	54	15
5.26(b)	14	57	18
5.26(c)	11	60	21

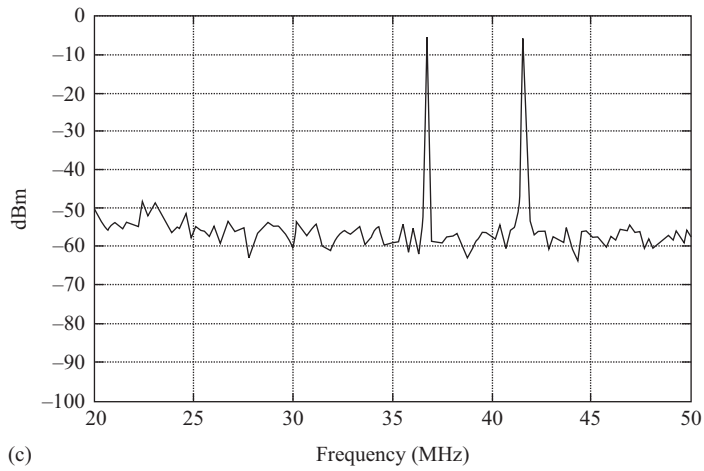
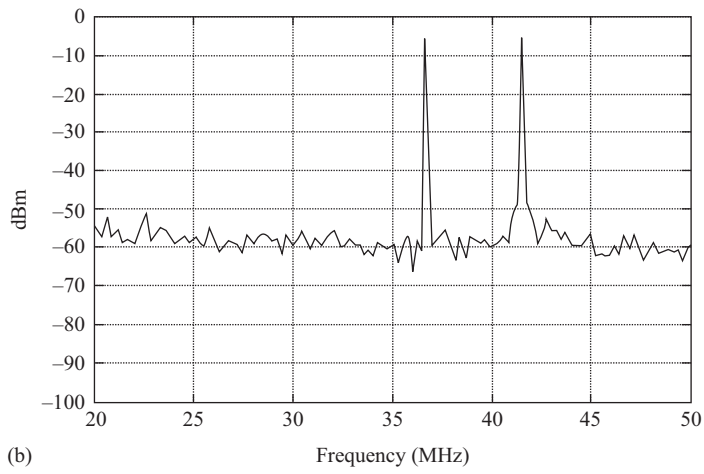
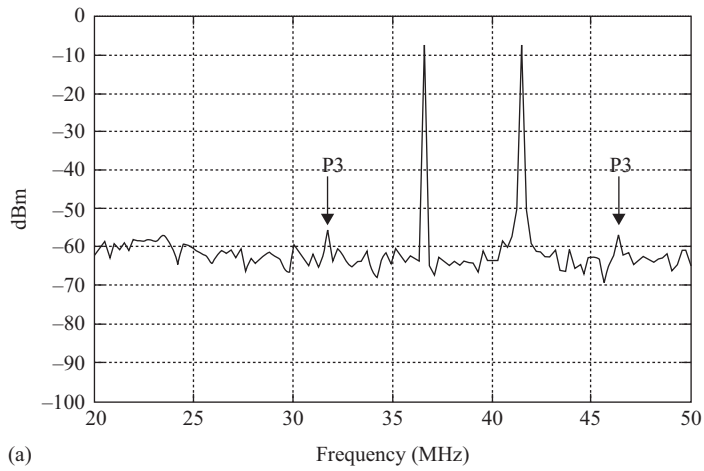


Figure 5.26 Output spectrum: (a) gain = 54 dB, (b) gain = 57 dB, and (c) gain = 60 dB.

be -62.2 dBm and the third-order intermodulation products at frequencies are clearly shown. In this case, the third-order intermodulation products are the lower limits of the dynamic range. The third-order intermodulation product at 31 MHz is at -56.4 dBm. Thus the dynamic range is 49.1 dB ($-7.3 + 56.4$). This is referred to as the two-tone spur-free dynamic range.

In Figure 5.26(b), the noise floor is measured at -58.6 dBm. The third-order intermodulation products, which should appear at 31 and 46 MHz, cannot be observed. The noise floor is the lower limit of the dynamic range. The dynamic range in this case is 51.3 dB ($-7.3 + 58.6$).

In Figure 5.26(c), the noise floor is at -55.2 dBm. The third-order intermodulation products cannot be observed. The noise floor is the lower limit of the dynamic range. The dynamic range is 47.9 dB ($-7.3 + 55.2$).

These three cases show that Figure 5.26(b) provides the largest dynamic range. The measured value is very close to the designed one. The reason the two other cases provide a lower dynamic range is that the gain is not at an optimum value. From these experimental data, one can see that the computer program can very accurately predict the performance of the receiver.

In order to see how close the third-order intermodulation products are to the noise floor (Figure 5.26(b)), the input signals are increased by 0.5 dB. The resulting spectrum is shown in Figure 5.27, where the third-order intermodulation products are clearly visible. Thus one can claim that the third-order intermodulation products in Figure 5.26(b) are very close the noise floor, which matches the design goal.

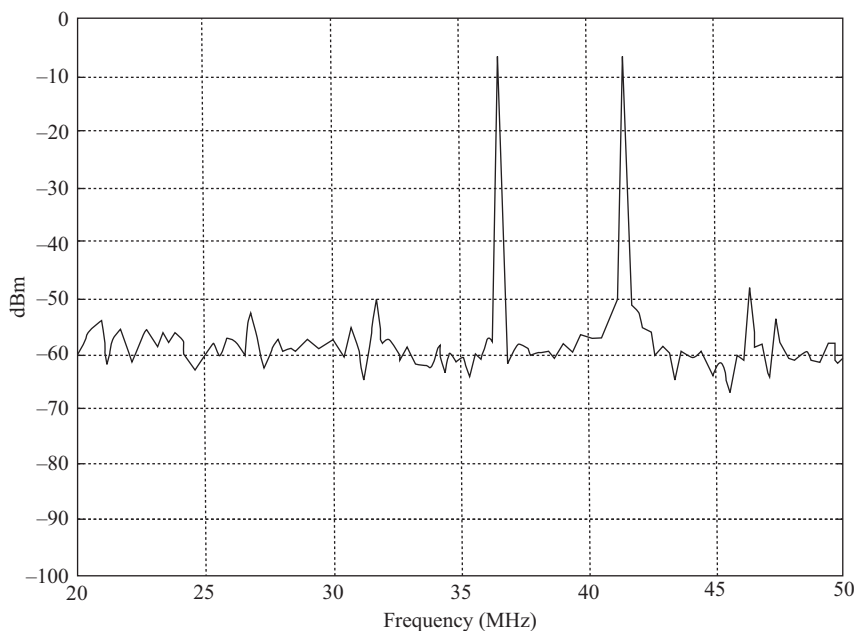


Figure 5.27 Spectrum output with gain = 57 dB and input slightly above the maximum level.

References

- [1] Tsui JBY. *Microwave Receivers with Electronic Warfare Applications*. New York: John Wiley & Sons; 1986.
- [2] Tserng HQ. 'Design and performance of microwave power GaAs FET amplifiers'. *Microwave Journal* 1979;**22**(6):94–95, 98–100.
- [3] Ohta K, Jodai S, Fukuden N, Hirano Y, Itoh M. 'A five watt 4–8 GHz GaAs FET amplifier'. *Microwave Journal* 1979;**22**(11):66–67.
- [4] Dilorenzo JV, Wisseman WR. 'GaAs power MESFET's: design, fabrication, and performance'. *IEEE Transactions on Microwave Theory and Techniques* 1979;**27**(5):367–378.
- [5] Peter G. 'Low noise GaAs FET dual channel front end'. *Microwave Journal* 1982;**25**(5):153–154, 156, 158.
- [6] Whelehan J. 'Low noise amplifiers for satellite communications'. *Microwave Journal* 1984;**27**(2):125, 128–130, 132–136.
- [7] Ayasli Y. 'Decade bandwidth amplification at microwave frequencies'. *Microwave Journal* 1984;**27**(4):71, 72, 74.
- [8] Bharj JS. '17 GHz low noise GaAs FET amplifier'. *Microwave Journal* 1984;**27**(10):121–122, 124, 126–127.
- [9] Sholley M, Maas S, Allen B, Sawires R, Nichols A. 'HEMT mm-wave amplifiers, mixers and oscillators'. *Microwave Journal* 1985;**28**(8):121–124, 126–131.
- [10] '6–18 GHz fully integrated MMIC amplifier'. *Microwave Journal* 1986;**29**(8):121–122.
- [11] Browne J. 'MMIC chip amplifier boosts 0.5 to 5 GHz'. *Microwave & RF* 1986;**25**(9):157–159.
- [12] Morgan W. 'Minimize IM distortion in GaAs FET amplifiers'. *Microwave & RF* 1986;**25**(10):107–108, 110.
- [13] Franke E, Deleon J. 'Broadband noise improvement in RF power amplifiers'. *RF Design* 1986;**November**:104.
- [14] Smith MA, Anderson KJ, Pavio AM. 'Decade-band mixer covers 3.5 to 35 GHz'. *Microwave Journal* 1986;**29**(2):163–164, 166–171.
- [15] Browne J. 'Microwave mixer family converts 1 to 18 GHz'. *Microwave & RF* 1986;**25**(10):209.
- [16] 'Mixer-amplifier combination is virtually load-insensitive'. *Microwave Journal* 1987;**30**(12):131–132, 134–135.
- [17] Jaeger RC. 'Tutorial: analog data acquisition technology part I—digital-to-analog conversion'. *IEEE Micro* 1982;**2**(2):20–37. 'Tutorial: analog data acquisition technology part II—analog-to-digital conversion'. *IEEE Micro* 1982;**2**(3):46–57. 'Tutorial: analog data acquisition technology part III—sample-and-holds, instrumentation amplifiers, and analog multiplexers'. *IEEE Micro* 1982;**2**(4):20–35. 'Tutorial: analog data acquisition technology part IV—system design, analysis, and performance'. *IEEE Micro* 1983;**3**(1):52–61.
- [18] Stafford KR, Blanchard RA, Gray PR. 'A complete monolithic sample/hold amplifier'. *IEEE Journal of Solid-State Circuits* 1974;**9**(6):381–387.

- [19] Tewksbury SK, Meyer F, Rollenhagen D, Schoenwetter H, Souders T. 'Terminology related to the performance of S/H, A/D, and D/A circuits'. *IEEE Transactions on Circuits and Systems* 1978;**25**(7):419–426.
- [20] Naylor J. 'Testing digital/analog and analog/digital converters'. *IEEE Transactions on Circuits and Systems* 1978;**25**(7):526–538.
- [21] Hewlett Packard. Dynamic Performance Testing of A to D Converters. Product note 5180A-2.
- [22] Carrier P. 'A microprocessor based method for testing transition noise in analog to digital converters'. *Proceedings of 1983 IEEE International Test Conference*. New York: IEEE; 1983: 610–620.
- [23] Doernberg J, Lee H-S, Hodges D. 'Full-speed testing of A/D converters'. *IEEE Journal of Solid-State Circuits* 1984;**19**(6):820–827.
- [24] IEEE Trial-Use Standard for Digitizing Waveform Recorders. IEEE Standard 1057. New York: IEEE; 1989.
- [25] Waveform Measurement and Analysis Committee of the IEEE Instrumentation and Measurement Society. *A Guide to Waveform Recorder Testing*. New York: IEEE; April 1990. This guide contains four papers: 1) Linnenbrink TE. 'Introduction to waveform recorder testing'; 2) Green PJ. 'Effective waveform recorder evaluation procedures'; 3) Souders TM, Flach DR. 'Step and frequency response testing of waveform recorders'; 4) Grosby PS. 'Waveform recorder sine wave testing-selecting a generator'.
- [26] Sklar B. *Digital Communications: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice Hall; 1988.
- [27] Wong PW. 'Quantization noise, fixed-point multiplicative roundoff noise, and dithering'. *IEEE Transactions on Acoustics, Speech and Signal Processing* 1990;**38**(2):286–300.
- [28] Davenport WB, Root WL. *An Introduction to the Theory of Random Signals and Noise*. New York: McGraw-Hill; 1958. Reprinted by IEEE Press, 1987.
- [29] West PD. Georgia Tech Research Institute. Private communication.
- [30] Larson LE. 'High-speed analog-to-digital conversion with GaAs technology: prospects, trends and obstacles'. *Proceedings of IEEE International Symposium on Circuits and Systems*, vol. 3. New York: IEEE; 1988:2871–2878.
- [31] Walter K. 'Test video A/D converters under dynamic condition'. *EDN* 1982;**August**:103–112.
- [32] Kester WA (ed.). *Data Conversion Handbook*. Oxford: Newnes; 2005.
- [33] Solid state microwave amplifiers. Catalog no. 5978. Sunnyvale, CA: Aertech Industries.
- [34] Avantek high frequency transistor primer. Santa Clara, CA: Avantek; 1971.
- [35] Designing with GPD amplifiers. Santa Clara, CA: Avantek; 1972.
- [36] Cheadle DL. Cascadable amplifiers. Tech-note, vol. 6, no. 1. Palo Alto, CA: Watkins-Johnson; January/February 1979.
- [37] Solid state amplifiers. Palo Alto, CA: Watkins-Johnson; June 1979.
- [38] Blackham D, Hoberg P. 'Minimize harmonics in scalar tests of amplifiers'. *Microwave & RF* 1987;**26**(8):143, 145, 147, 149.

- [39] Sorger GU. 'The 1 dB gain compression point for cascaded two-port networks'. *Microwave Journal* 1988;**July**:136.
- [40] '17 most asked questions about mixers'. Brooklyn, NY: Mini-Circuits.
- [41] Cheadle D. 'Selecting mixers for best intermod performance'. *Microwaves* 1973;**12**(11):48–52 (part 1), 1973;**12**(12):58–62 (part 2).
- [42] Neuf D, Brown D. 'What to look for in mixer Specs'. *Microwaves* November 1974.
- [43] Reynolds JF, Rosenzweig MR. 'Learn the language of mixer specification'. *Microwaves* 1978; **17**(5):72–80.
- [44] Jacobi JH. 'IMD: still unclear after 20 years'. *Microwave & RF* 1986;**25**(12):119–126.
- [45] Steinbrecher DH. 'Broadband high dynamic range A/D conversion limitations'. *Proceedings of International Conference on Analogue to Digital and Digital to Analogue Conversion*. New York: IEEE; 1991:95-99.
- [46] Sharpin DL, Tsui JBY. 'Analysis of the linear amplifier/analog-digital converter interface in a digital microwave receiver'. *IEEE Transactions on Aerospace and Electronic Systems* 1995; **31**(1):248–256.

Appendix 5.A

```
% df6_18a.m simulates an ideal ADC and find spur levels with automatic change
% of input frequencies.
% JT 26 March 92
% ***** USER INPUT *****
clear
input(' # of bits = ');
bits = ans;
kk = input('Do you want window? y/n: ', 's');
if kk == 'y',
    ws = 'Blackman window';
else
    ws = 'square window';
end
snr = 20*log10((2^(bits-1))/sqrt(2));
f_samp = 250;
ts = 1/f_samp;
% *****
n = 516;
points = n;
time = [0:n-1];
amp_no = 1;
% ***** CALCULATION OF CONSTANTS *****
amp1 = sqrt(2*amp_no)*(10^(snr/20));
% ***** START LOOP CHANGE INPUT FREQUENCY *****
To = n*ts;
for k= 1:1000;
kn = randn(1,1);
```

```

fil = (kn*240+8)/To;
x = amp1*cos(2*pi.*fil*ts*time);
% ***** QUANTIZATION *****
x_q1 = quantiz(x, bits);
% ***** WINDOW *****
win = blackman(n);
if kk == 'y',
    x_q = x_q1 .* win';
else
    x_q = x_q1;
end
% ***** FFT *****
x_qf = fft(x_q);
y = abs(x_qf);
y_log = 20*log10(y);
% ***** FIND MAX SPURS *****
[p1 i1 p2 i2 dr(1,k)] = peak(y_log(1:n/2));
end
% ***** END OF LOOP *****
m_min = min(dr)

```

Appendix 5.B

```

% QUANTIZ simulates an ideal mid-rise quantizer
% JT 6 April 92
% x : input data to exercise all bits xmax=2^(bits-1)
% bits : number of bits
% x_q: output
% *****
function x_q = quantiz(x,bits)
% *****
q_levels = 2^bits;
q_max = 2^(bits-1);
q_min = -2^(bits-1)+1;
n = length(x);
adj = 0.5 * ones(size(n));
% adj = 0.5 * ones(1:n);
x_adj = x + adj;
x_qt = round(x_adj);
bigger = find(x_qt >= q_max);
if length(bigger) > 0,
    x_qt(bigger) = (q_max)*(ones(size(bigger)));
    % x_qt(bigger) = (q_max)*(ones (1:length(bigger)));
end

```

```

smaller = find(x_qt <= q_min);
if length(smaller) > 0,
    x_qt(smaller) = (q_min)*(ones(size(smaller)));
    % x_qt(smaller) = (q_min)*(ones(1:length(smaller)));
end
x_q = x_qt-.5;

```

Appendix 5.C

```

% PEAK detects the highest two peaks
% JT Modified 12 May 1992
function [peak1,ind1,peak2,ind2,dr] = peak(r)
% r = input(' input matrix = ');
rif = r(1);
ril = r(length(r));
th = 1.5; % threshold
[max_r ind_r] = max(r);
if ind_r==1,
    m1 = [0 r ril-1];
    m2 = [0 0 r];
    m3 = [r ril-1 ril-1];
elseif ind_r==length(r),
    m1 = [rif-1 r 0];
    m2 = [rif-1 rif-1 r];
    m3 = [r 0 0];
else
    m1 = [rif-1 r ril-1];
    m2 = [rif-1 rif-1 r]; % shift right
    m3 = [r ril-1 ril-1]; % shift left
end
m4 = m1-m2 > 0; % compare amp
m5 = m1-m3 > 0; % " "
m4_th = m1-m2 > th; % threshold right shift
m5_th = m1-m3 > th; % " left "
m6_th = m4_th + m5_th > 0; % Combine threshold ones
m6_zo = m4.*m5; % combine m4 and m5
m6 = m6_th .* m6_zo;
ind = find(m6); % find the peaks
m7 = m1(ind); % form a new matrix of peaks only
[peak1 indm7_1]= max(m7);
Ind1 = ind(indm7_1)-1;
m7(indm7_1) = -200;

```



```

if length(m7) == 1,
    peak2 = 0;
    ind2 = 0;
    dr= 100;
else
    [peak2 indm7_2] = max(m7);
    ind2 = ind(indm7_2)-1;
    dr = peak1-peak2;
end
[peak1 ind1 peak2 ind2 dr];

```

Appendix 5.D

```

% df7eq1.m :This prog calculates total gain, noise figure and 3rd
order intercept pt
% ***** input data *****
gc_db = input('gain of all components in dB i.e. [15 15 -3] = ');
fc_db = input('noise figure of all components in dB [ ] = ');
qc_db = input('3rd order intercept pt of all components in dB [ ] = ');
len = length(gc_db);
% ***** convert dB into power/ratio *****
gc = 10 .^(gc_db./10);
fc= 10 .^(fc_db./10);
qc= 10 .^(qc_db./10);
% ***** calculate gain *****
g1 = cumprod(gc);
g1m = [1 g1(1:len-1)];
gt = g1(len);
gt_db = 10*log10(gt);
% ***** calculate noise figure *****
f1 = [0 ones(1,len-1)];
f2 = fc-f1;
f_div = f2./g1m;
ft = sum(f_div);
ft_db= 10*log10(ft);
% ***** calculate 3rd order intercept pt *****
q1 =g1./qc;
q_den = sum(q1);
qt = gt/q_den;
qt_db = 10*log10(qt);
outp2 = [gt_db ft_db qt_db];
disp(' ')
disp(' Gain? NF Q3')
disp(outp2)

```

Appendix 5.E

```
% df7eq2.m provides the design between an amplifier and ADC.
% JT 24 June 1992
clear
%***** INPUT *****
% ** AMP **
n1_db = -174;      % noise at input of amplifier per unit bandwidth
br = 30e6; % rf bandwidth
br_db = 10*log10(br);
f_db = 3.3; % noise figure
% ** ADC **
b = 8; % # of bits
vs = 270; % saturation voltage in mv
q = vs/(2^(b-1)); % voltage per quantization level
fs = 250e6; % sampling frequency in Hz
R = 50; % input impedance
n = 1024; % FFT length
m = input('enter the value of m = ');
m_db = 10*log10(m);
m1 = m+1; %Eq 20
m1_db = 10*log10(m1);
md_db = m1_db - m_db;
% ***** GENERATE CONSTANT *****
ps = (vs*vs)*1e-3/(2*R); %Eq 14 1e-3 changes to mw
ps_db = 10*log10(ps);
nb_db = ps_db - 1.76 - 6*b; %Eq 15
bv = fs/n; %Eq 16
bv_db = 10*log10(bv);
no_db = nb_db + m_db; %Eq 20
no = 10.^(no_db/10);
von = sqrt(no*1e3*2*R); %noise voltage
similar as Eq 14
vn = sqrt(3*no*1e5); %Eq 25
ns_db = no_db + md_db; %Eq 17 18 19 21
vsn = vs-vn; %Eq 26
psn = (vsn.*vsn)/1e5; %Eq 26
psn_db = 10*log10(psn);
g_db = no_db - n1_db - f_db - br_db; %Eq 17 18
pi_db = psn_db - 6 - g_db; %Eq 27
p3_db = ns_db - br_db + bv_db; %Eq 22
% ***** CALCULATION *****
q3_db=(3*pi_db-n1_db+2*g_db-f_db-bv_db-md_db)/2; %Eq 24
fs_db = f_db + md_db; %Eq 21
dr_db = pi_db + g_db - p3_db; %Eq 28
```

```
nqvr = von/q;  
disp(' m Gain Q3 P3 NF DR N/Q')  
en = length(m);  
dp = [m' g_db' q3_db' p3_db' fs_db' dr_db' nqvr'];  
disp(dp)
```

Frequency Downconverters

6.1 Introduction

The frequency range of interest in electronic warfare (EW) applications extends from 100 MHz to 18 GHz. Some long-range searching radars operate under 2 GHz, and one can consider EW operations nominally from 2 to 18 GHz. An EW receiver usually has an instantaneous bandwidth of 0.5 to 4 GHz, which is limited mainly by hardware constraints. This receiver is often referred to as an intermediate frequency (IF) receiver, which is time-shared to cover the 2- to 18-GHz frequency range. Generally it is difficult to build an IF receiver with wide instantaneous bandwidth. Even if one could build such an IF receiver, the digital processor following the receiver would be unable to process the data in near real time. As a result, some of the data will be ignored by the processor. In this case, the digital processor limits the bandwidth capability of the IF receiver. In order to increase the bandwidth of IF receivers, both receiver and processor technologies must be advanced.

To simplify the discussion in this chapter, let us assume that an IF receiver has approximately 1 GHz instantaneous bandwidth, which means any signal in the bandwidth will be intercepted. In order to cover the frequency range of interest, the input frequency range is broken into many parallel channels (referred as channelization) and each channel is frequency converted to match the input frequency range of the IF receiver.

In conventional analog receivers, the input frequency of 2 to 18 GHz is divided into 1-GHz bandwidths and each band is converted to some common IF. In a digital receiver, two types of conversion from analog to digital are often considered. One kind is single channel (or real-data conversion), where there is only one output data channel. The other approach is to generate two output channels that are 90-deg out of phase; this is referred to as in-phase or quadrature-phase downconversion, or simply the *I* and *Q* channels.

This chapter discusses both schemes of frequency conversion—one-channel and two-channel conversions. Both analog and digital frequency conversions are discussed. Several digital approaches to create the *I* and *Q* channels are also discussed. The impact of imbalance between the *I* and *Q* channels on receiver performance are presented. Finally, a correction scheme to rectify the *I* and *Q* channels imbalance is discussed. Plenty of imbalance compensation methods have been proposed for communication systems in which the signal frequency is known to the receiver and the working bandwidth of the communication receiver is usually narrower than that of the wideband receiver. For wideband receiver applications, signal frequency is unknown to the receiver. Both imbalance compensation techniques for narrowband and wideband systems are discussed in this chapter.

6.2 Baseband Receiver Frequency Selection [1, 2]

In general, the input of an IF receiver should be kept under an octave of bandwidth. An octave of bandwidth means that the high end of the bandwidth is twice the frequency as at the low end of the bandwidth. For example, if the input bandwidth is 1 GHz, the frequency range from 1 to 2 GHz is an octave. If the input frequency range is lower than these values (i.e., 0.5 to 1.5 GHz), the bandwidth is over an octave. If the frequency range is above these values (i.e., 2 to 3 GHz), the bandwidth is under an octave.

If the bandwidth is over an octave, the second harmonic of a low-frequency signal may be in the bandwidth. The second harmonic can limit the dynamic range. For the 0.5- to 1.5-GHz band, if an input frequency is at 600 MHz, then the second harmonic is at 1200 MHz, which is still in the input bandwidth.

The only known analog receivers that are designed to cover more than an octave of bandwidth are the crystal video and instantaneous frequency measurement (IFM) receivers. However, neither of these receivers can recognize or process simultaneous signals. The crystal video receiver only reports the pulse amplitude, pulse width, and time of arrival (TOA). In this case, the second harmonic does not have any adverse effect. The IFM receiver can encode the frequency of one signal, as discussed in Chapter 2. The second harmonic amplitude is usually lower than the input signal amplitude and will not affect the frequency encoding circuits. Almost all other types of analog receivers have an input bandwidth less than one octave.

An analog-to-digital converter (ADC) has limited input frequency range. In many cases, due to limitations of the ADC, it is impractical to have a high-input frequency. As mentioned in Chapter 5, the dynamic range in many ADCs degrades when the input frequency is high. Thus there are two choices. One approach is to narrow the input bandwidth and restrict the input frequency range to under one octave. Through technology improvements in ADCs, a wider band should be achievable with this approach. A second approach is to choose a bandwidth over an octave, or even start from a frequency close to direct current (dc). In the second approach, many spurs may be contained in the input bandwidth of the receiver, which is discussed in the next section.

6.3 Frequency Conversion [3–7]

The purpose of frequency conversion is to translate the input frequency from one frequency range to a different one at the output of the converter. The common way to achieve frequency conversion is through a mixer, as shown in Figure 6.1. A mixer is a nonlinear device. The output current can be related to the input voltage V as

$$I = a_0 + a_1V + a_2V^2 + \cdots, \quad (6.1)$$

where a_i s are constants. Assume that the input voltage V contains two sine waves as

$$V = V_i \sin(2\pi f_i t) + V_o \sin(2\pi f_o t), \quad (6.2)$$

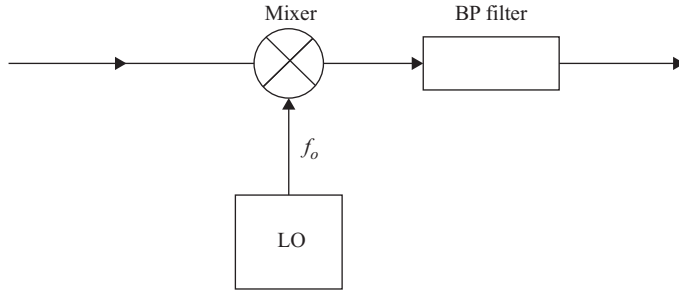


Figure 6.1 A mixer circuit.

where V_i and f_i are the amplitude and frequency of the signal, respectively, and V_o and f_o are the amplitude and frequency of the local oscillator, respectively. Substituting this relationship into (6.2) and considering the nonlinear term a_2V^2 in particular,

$$\begin{aligned} a_2V^2 &= a_2V_i^2 \sin^2(2\pi f_i) + a_2V_o^2 \sin^2(2\pi f_o) \\ &+ a_2V_iV_o\{\cos[2\pi(f_o - f_i)t] - \cos[2\pi(f_o + f_i)t]\}. \end{aligned} \quad (6.3)$$

The last term in this equation corresponds to the desired output frequency ($f_o - f_i$) [or ($f_i - f_o$)] and ($f_o + f_i$). If the output frequency is lower than the input frequency, the process is called frequency downconversion, otherwise it is called frequency upconversion.

Although a mixer is a nonlinear device, the mixing process is often considered as a linear process in the sense of input/output superposition. The information contained in the input signal does not change, only the frequency is shifted. In an amplifier chain, a mixer can be treated as an amplifier. A mixer has a gain, a noise figure, and a third-order intercept point, as discussed in the previous chapter. Its gain is usually less than unity (negative in decibels, or a loss), although in some mixers the gain might be positive. The noise figure may often be taken as equal to the loss of the mixer unless another value is given by the manufacturer. The third-order intercept point is given by the manufacturer.

As (6.3) indicates, there are many frequencies at the output of a mixer besides the desired one. These intermediate output frequencies f_{if} can be written as

$$f_{if} = mf_l + nf_h, \quad (6.4)$$

where m and n are positive or negative integers. In this equation, f_l represents the low-input frequency and f_h the high-input frequency. Either one can be used to represent the signal or oscillator frequency, thus f_i and f_o are not used. If both m and n are 1, the output is the sum of the two frequencies. If one of them is +1 and the other is -1, the output is the difference frequency. Other than these two frequencies, all the other frequencies are considered spurious frequencies (or spurs) and should be kept at minimum.

A convenient way to show the spurs is by using a spur chart, which is shown in Figure 6.2. In this mixer chart, the difference frequency is the desired result. To simplify the

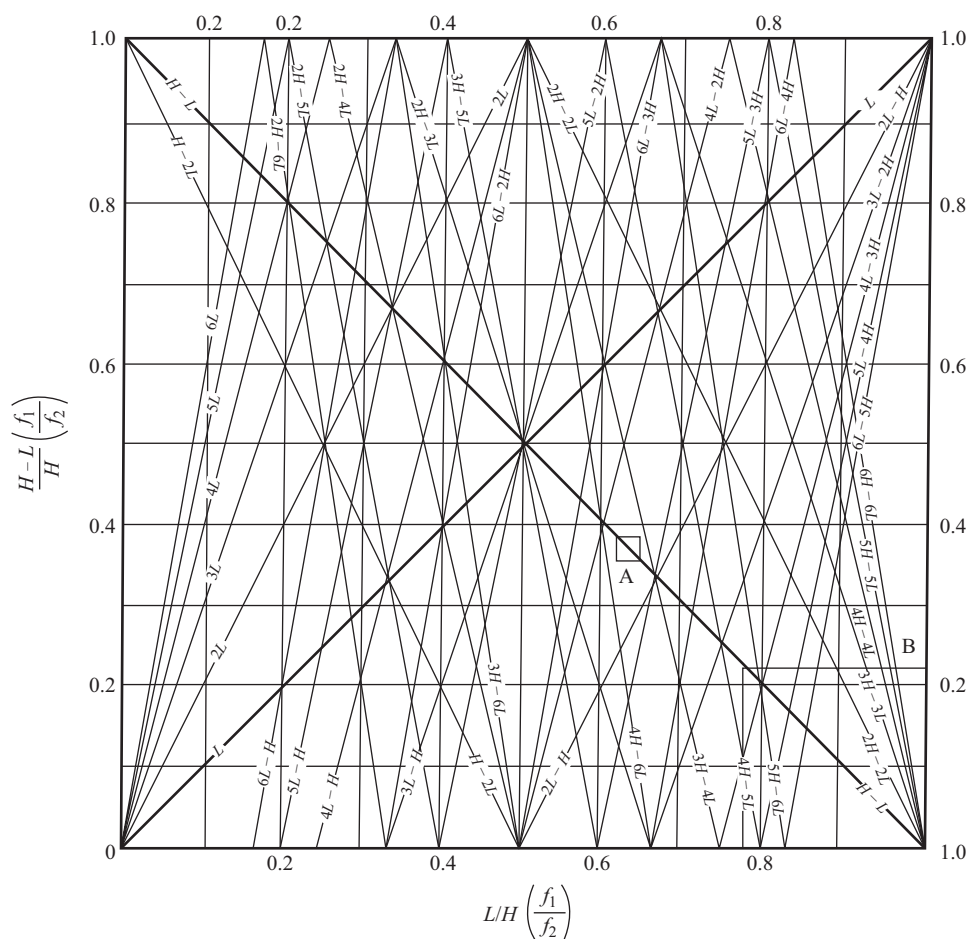


Figure 6.2 Mixer spur chart.

notation, H is used to represent the higher frequency, which can be either the signal or the local oscillator, and L represents the lower frequency. The line drawn diagonally and labeled $H - L$ represents the desired output. All the other lines represent spurious outputs. The highest-order spur expressed in this figure is 6, which is represented by $6H$ or $6L$. As a general rule, a high-order spur has a smaller amplitude.

In Figure 6.2, the square marked “A” represents an area in which there is no spur. In a narrowband downconverter, this is a desired choice. An IF output starting from zero frequency is represented by a square “B” on the lower right corner. In the square, there are many spurs (i.e., $2H - 2L$, $3H - 3L$, etc.), therefore the choice of an IF bandwidth starting from zero frequency is not a good one. This is a compromise needed to cover a wide bandwidth. If the technology in ADCs and digital processing can be improved to gigahertz operation speed, a wide IF bandwidth of less than an octave might be achievable.

6.4 In-Phase (*I*) and Quadrature-Phase (*Q*) Channel Conversion

The input of a receiver is always a single channel and the data can be considered as real-valued. In order to keep this discussion simple, the following input signal $S(t)$ will be considered:

$$S(t) = A \sin(2\pi f_i t), \quad (6.5)$$

where A and f_i are the amplitude and frequency of the input signal, respectively. To convert a real-valued signal to a complex-valued signal that has two channels, *I* and *Q*, two outputs are generated and they are 90-deg out of phase with one another. If the frequency is down-converted, the two outputs can be written as

$$\begin{aligned} I(t) &= A \cos[2\pi(f_i - f_o)t] \\ Q(t) &= A \sin[2\pi(f_i - f_o)t], \end{aligned} \quad (6.6)$$

where f_o is the local oscillator frequency. In this equation, the *I* and *Q* channels are arbitrarily designated. As long as the two outputs are 90-deg out of phase, one can be called the *I* channel and the other one the *Q* channel. These two outputs can be written into a complex form as

$$I(t) + jQ(t) = A\{\cos[2\pi(f_i - f_o)t] + j\sin[2\pi(f_i - f_o)t]\} = Ae^{j2\pi(f_i - f_o)t}. \quad (6.7)$$

To achieve real-to-complex signal conversion, more hardware is needed, but there are compensating advantages.

1. The bandwidth of the input signal can be doubled if both outputs are digitized. This point can be explained in either the time or the frequency domain. In the time domain, if the sampling frequency is f_s , one must obtain two samples per cycle at the highest input frequency to fulfill the Nyquist sampling rate, thus the highest frequency is $f_s/2$. If there is a *Q* channel, one more sample will be collected, thus the highest frequency can be extended to f_s . In the frequency domain, if the input is real there are positive and negative frequency components, as discussed in Chapter 3, and the highest frequency without ambiguity is $f_s/2$. For complex data, there are no negative frequency components and the unambiguous range extends to f_s .
2. Amplitude information is maintained in an *I-Q* channel conversion. If real-valued data are processed by an analog receiver, the amplitude information can be recovered from the video detector, which has a lowpass filter to smooth out the radio frequency (RF) ripples. Therefore the amplitude detection has never been considered as a problem in analog microwave receivers. Of course, a similar idea can be extended to digital receivers by taking the absolute value of the real input data. This can be accomplished by passing the square of the real input through a lowpass filter to obtain an approximation of the amplitude information.

With *I* and *Q* channels, an obvious approach is through the relationship

$$A = \sqrt{A^2 \sin^2[2\pi(f_i - f_o)t] + A^2 \cos^2[2\pi(f_i - f_o)t]}. \quad (6.8)$$

If there is only one signal, the amplitude A calculated from this equation will be a constant for each sample. If the amplitude changes, one can say that the signal is amplitude modulated, or in the case of an EW receiver, one can conclude that there are simultaneous signals. Thus, from the output of the I - Q channels, either the amplitude of a signal can be found or a simultaneous signal condition can be detected.

3. The I - Q channels can be used to determine instantaneous frequency if only one signal is present. This method can be extended to determine two frequencies, which will be discussed in Chapter 8. Considering that the signals in (6.6) are digitized at time t_i and t_{i+1} , the instantaneous angle at these time instances can be found as

$$\begin{aligned}\theta_i &= \tan^{-1} \left[\frac{A \sin[2\pi(f_i - f_o)t_i]}{A \cos[2\pi(f_i - f_o)t_i]} \right] = 2\pi(f_i - f_o)t_i \\ \theta_{i+1} &= \tan^{-1} \left[\frac{A \sin[2\pi(f_i - f_o)t_{i+1}]}{A \cos[2\pi(f_i - f_o)t_{i+1}]} \right] = 2\pi(f_i - f_o)t_{i+1}.\end{aligned}\tag{6.9}$$

The difference in angle θ can be used to find the instantaneous frequency as

$$f_i - f_o = \frac{\theta_{i+1} - \theta_i}{2\pi\Delta t},\tag{6.10}$$

where $\Delta t = t_{i+1} - t_i$.

If the input signal contains a single frequency, this approach can improve the frequency data resolution. In contrast, if a signal is sampled over a 1- μ s interval and an FFT is performed on the input, the frequency resolution is 1 MHz. If the above method is used and the delay time is 1 μ s, the maximum unambiguous bandwidth is 1 MHz. If one can measure the phase angle with 6 bits of resolution, the resolution will be 5.625 deg (360 deg/64). Under this condition, the frequency resolution can be measured as 15.625 kHz (1000 kHz/64), which means the resolution improved 64 times from the FFT approach. If there are simultaneous signals, this simple phase approach does not work. However, a slightly different approach can solve two frequencies, which will be discussed in Chapter 8.

6.5 Imbalance in I and Q Channels [8, 9]

When an I and Q downconverter is simulated, it is often assumed that the two outputs are perfectly balanced (i.e., the two output components have equal amplitude and are 90 deg out of phase). However, in the actual fabrication of an I and Q downconverter, this is usually not the case, especially if the converter covers a wide bandwidth. In other words, the two channel outputs may not have the same amplitude and their respective phases are not exactly 90 deg apart. The effect of this imbalance can generate an image signal, which can limit the instantaneous dynamic range of a receiver.

Let us use a Fourier transform to explain the generation of an image frequency. Instead of finding the Fourier transform of $\exp(j2\pi f_i t)$ directly, let us use the relationship

$$e^{j2\pi f_i t} = \cos(2\pi f_i t) + j \sin(2\pi f_i t).\tag{6.11}$$

The Fourier transforms of $\cos(2\pi f_i t)$ and $\sin(2\pi f_i t)$ can be written as

$$\begin{aligned}\mathcal{F}[\cos(2\pi f_i t)] &= \frac{\delta(f - f_i) + \delta(f + f_i)}{2} \\ \mathcal{F}[\sin(2\pi f_i t)] &= \frac{-j\delta(f - f_i) + j\delta(f + f_i)}{2}.\end{aligned}\quad (6.12)$$

The output can be combined as

$$\mathcal{F}(e^{j2\pi f_i t}) = \frac{1}{2} \{ \delta(f - f_i) + \delta(f + f_i) + j[-j\delta(f - f_i) + j\delta(f + f_i)] \} = \delta(f - f_i). \quad (6.13)$$

The negative frequency components $\delta(f + f_i)$ cancel each other. Only the positive frequency component $\delta(f - f_i)$ is left. These results are shown in Figure 6.3. Figure 6.3(a) represents the Fourier transform of the cosine signal, Figure 6.3(b) represents the Fourier transform of the sine signal, and Figure 6.3(c) represents the combined results of (6.13).

The two outputs of I and Q may not have the same amplitude, for example, $\cos(2\pi f_i t)$ versus $0.8 \sin(2\pi f_i t)$. The negative frequency components of the combined result will not exactly cancel each other. There will be an output in the negative frequency component position, as shown in Figure 6.4, and this component is often referred to as the image of the true signal. In receiver design, the image of a strong signal can limit the instantaneous dynamic range of the receiver because the weak signal must be higher than the image to be detected.

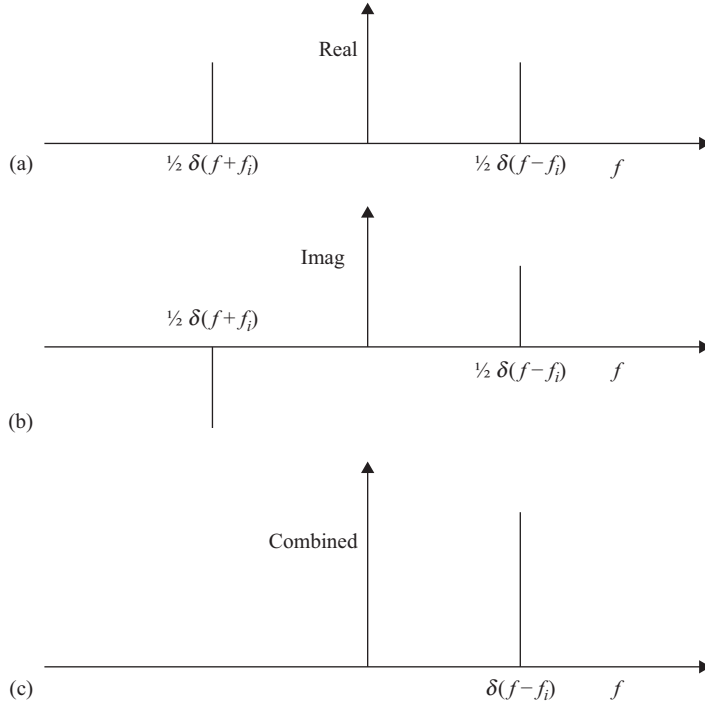


Figure 6.3 Output of balanced I and Q channels of signal $e^{j2\pi f_i t}$: (a) frequency spectrum of $\cos(2\pi f_o t)$, (b) frequency spectrum of $\sin(2\pi f_o t)$, and (c) combined result of (a) and (b).

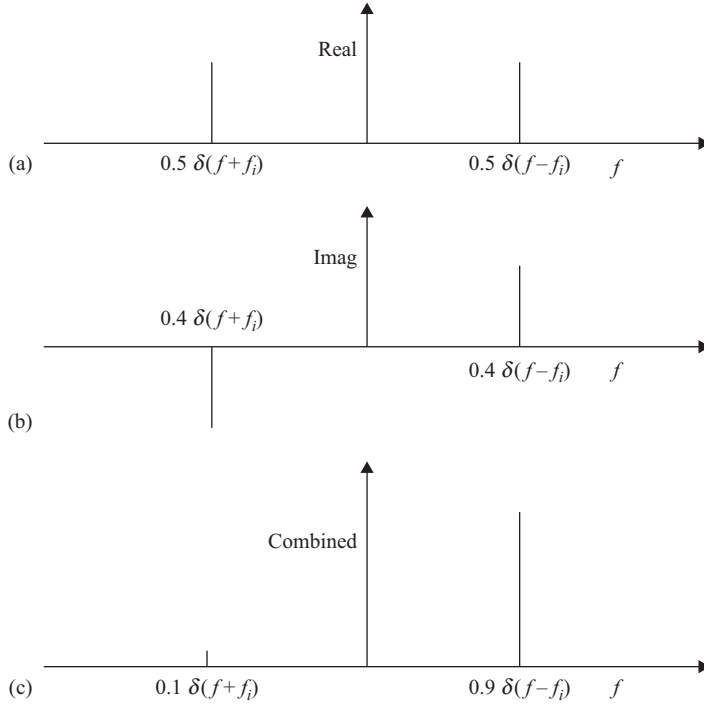


Figure 6.4 Output of imbalanced I and Q channels: (a) frequency spectrum of $\cos(2\pi f_o t)$, (b) frequency spectrum of $0.8 \sin(2\pi f_o t)$, and (c) combined result of (a) and (b).

Now let us find the amplitude of the image as a function of imbalance between the two channels and relate that to the instantaneous dynamic range of the receiver. This discussion is based on Engler [9]. First, the two outputs are expressed as

$$\begin{aligned}
 s(t) &= \cos(2\pi f_{if} t) + ja \sin(2\pi f_{if} t + \epsilon) \\
 &= \frac{1}{2} [e^{j2\pi f_{if} t} + e^{-j2\pi f_{if} t}] + \frac{\alpha}{2} [e^{j(2\pi f_{if} t + \epsilon)} - e^{-j(2\pi f_{if} t + \epsilon)}] \\
 &= \frac{1}{2} [e^{j2\pi f_{if} t} (1 + \alpha e^{j\epsilon})] + \frac{1}{2} [e^{-j2\pi f_{if} t} (1 - \alpha e^{j\epsilon})], \tag{6.14}
 \end{aligned}$$

where f_{if} is the IF output angular frequency, α is the amplitude imbalance, and ϵ is the phase imbalance. In this example, the cosine channel is assumed to be perfect and the sine channel is imbalanced. Since the amplitude and phase errors are relative values between the channels, the error can be included in one channel without loss of generality. From this equation, the desired signal can be considered as $e^{j2\pi f_{if} t}$ and the image as $e^{-j2\pi f_{if} t}$. If $\alpha = 1$ and $\epsilon = 0$, the desired term becomes $e^{j2\pi f_{if} t}$ and the image becomes zero as expected.

In general, the corresponding amplitudes of the signal and image are $1 + \alpha e^{j\epsilon}$ and $1 - \alpha e^{-j\epsilon}$, respectively, which can be represented by phasors, as shown in Figure 6.5. In this figure, A_d is the amplitude of the desired signal and A_i is the amplitude of the image signal.

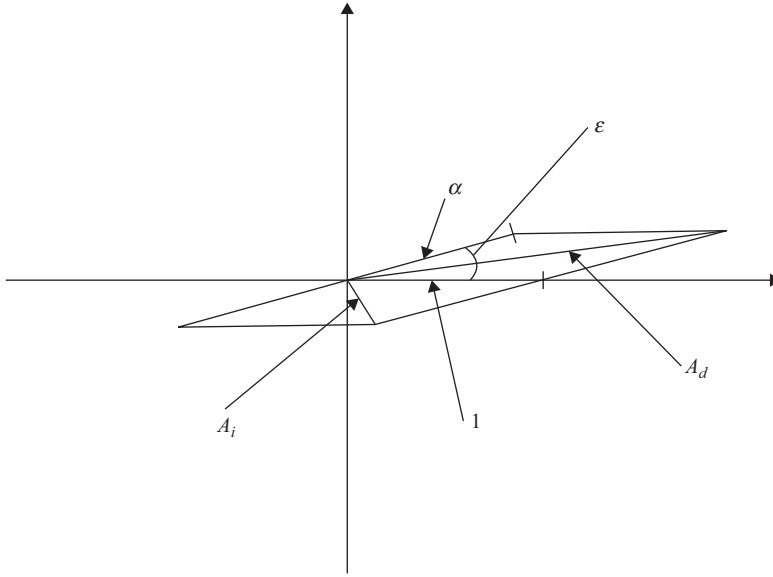


Figure 6.5 Graphical representation of *I* and *Q* outputs.

Through the law of cosine with reference to Figure 6.5, one can see that the desired output A_d and the image A_i can be expressed as

$$\begin{aligned} A_d^2 &= 1 + \alpha^2 + 2\alpha \cos(\epsilon) \\ A_i^2 &= 1 + \alpha^2 - 2\alpha \cos(\epsilon). \end{aligned} \quad (6.15)$$

The image amplitude relative to the desired amplitude can be written in decibels as

$$10 \log \left(\frac{A_i}{A_d} \right)^2 = 10 \log \frac{1 + \alpha^2 - 2\alpha \cos(\epsilon)}{1 + \alpha^2 + 2\alpha \cos(\epsilon)}. \quad (6.16)$$

This value is the dynamic range of the *I-Q* converter as limited by the channel imbalance.

The result of (6.16) is displayed in Figure 6.6. In this figure, the curves are in either a horizontal or a vertical direction. The effect can be explained as follows. Assume that there is a 2-deg imbalance in phase. From the figure, the image will be 35 dB from the desired signal as long as the amplitude balance is less than approximately 0.15 dB. If the amplitude balance is worse than 0.15 dB, the image amplitude will be dominated by the amplitude imbalance. For example, if there is a 1.5-dB amplitude imbalance, the image will be 15-dB down as long as the phase imbalance is below 20 deg. In other words, the worst performance of the imbalance factor will be the dominant factor in determining the image amplitude. In general, the phase match is more difficult to achieve, especially, in wideband downconverter systems.

6.6 Analog *I* and *Q* Downconverters

Analog *I* and *Q* downconverters are the most commonly used approach in EW receiver design. The main advantage in such a design is the wide bandwidth, and achieving a few gigahertz bandwidth is not unreasonable. The major deficiency of this approach is that it is difficult to achieve good balance between the channels.

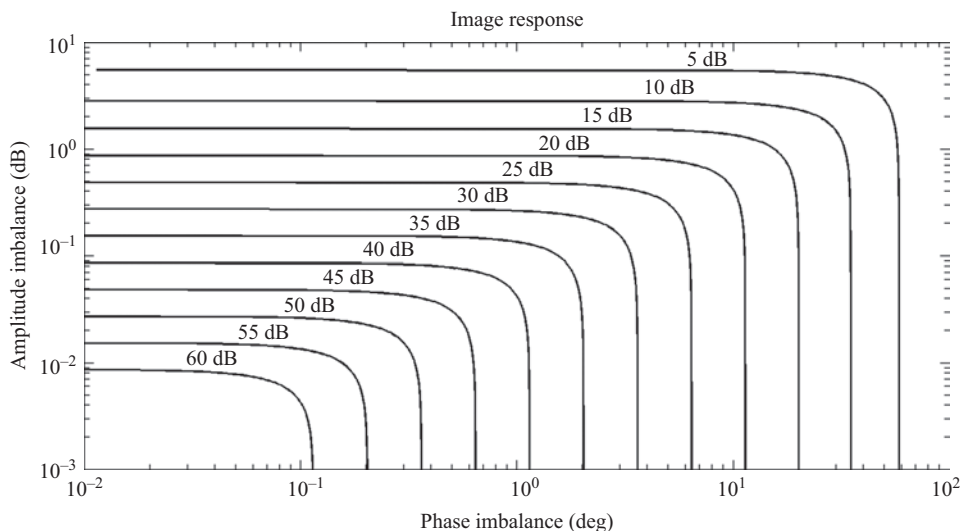


Figure 6.6 Image amplitude as a function of amplitude and phase imbalance.

The main component is a 90-deg phase shifter. There are two different approaches to build analog *I-Q* downconverters and they are shown in Figure 6.7.

In both cases, there are two mixers, but only one local oscillator that feeds both mixers. In Figure 6.7(a), a 90-deg phase shift is introduced in the input of the converter. In Figure 6.7(b), the 90-deg phase shift is introduced in the local oscillator.

These two approaches provide very similar results. The choice of the approach usually depends on the availability of components. Narrowband components are used in the network between the local oscillator and the mixers. The operating frequency of these components is equal to the local frequency f_o . The components used in the network between the input and the mixers are wideband, which compromises the bandwidth of the input signal.

There are, in general, three ways to select the local oscillator frequency: (1) f_o is below the input frequency f_i , (2) f_o is above the input frequency f_i , and (3) f_o is in the middle of the input bandwidth. The first two approaches are obvious, thus only the last approach will be discussed.

Let f_1 and f_2 be the lower and upper bounds of the input frequency band. In the third approach, the local oscillator frequency f_o is often put at the center of f_1 and f_2 or $(f_1 + f_2)/2$. In this approach, the IF bandwidth is half the input bandwidth, that is, $(f_2 - f_1)/2$, because both the upper and lower bands fold in the same frequency range. However, whether the input frequency is above or below f_o can be determined from the phase relation between the *I* and *Q* channels. Therefore, although the IF bandwidth is half the input bandwidth, the overall bandwidth is equal to the input bandwidth.

Let us use a simple example to demonstrate this idea. Assume the local oscillator signal feeding the two mixers is 90-deg out of phase. Thus the inputs to the two mixers from the local oscillators can be written as

$$\begin{aligned} v_{o1} &= \sin(2\pi f_o t) \\ v_{o2} &= \cos(2\pi f_o t) \end{aligned} \quad (6.17)$$

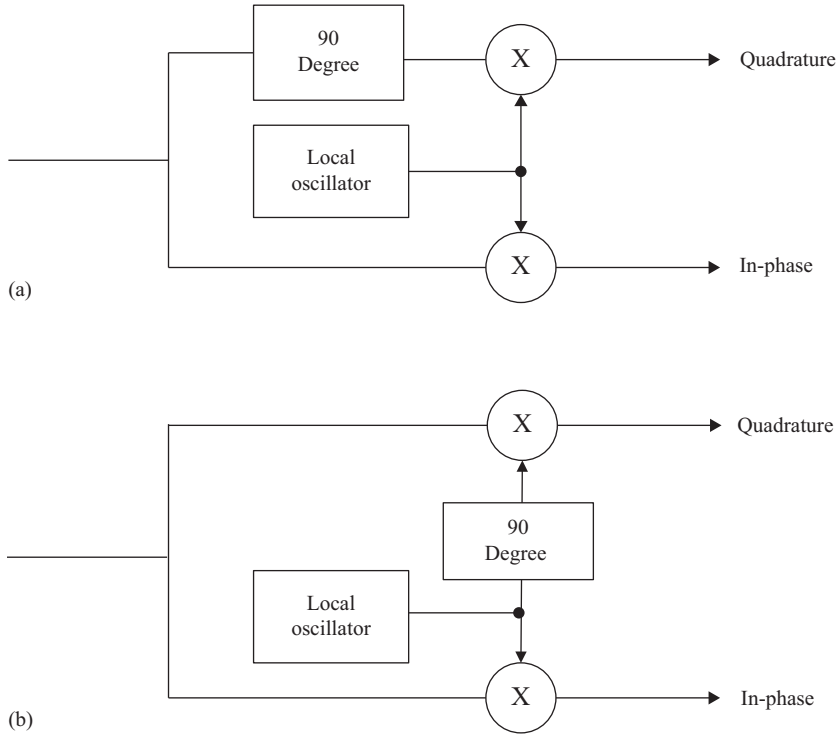


Figure 6.7 Analog *I* and *Q* downconverters: (a) a 90-deg phase shift introduced in the input path and (b) a 90-deg phase shift introduced in the local oscillator.

and the input signal is

$$v_i = \sin(2\pi f_i t), \quad (6.18)$$

where all the signals are assumed to have unity amplitudes. The output of the mixers can be written as

$$\begin{aligned} v_{if1} &= v_{o1} v_i = \sin(2\pi f_o t) \sin(2\pi f_i t) \\ &= \frac{1}{2} \{ \cos[2\pi(f_i - f_o)t] - \cos[2\pi(f_i + f_o)t] \} \\ v_{if2} &= v_{o2} v_i = \cos(2\pi f_o t) \sin(2\pi f_i t) \\ &= \frac{1}{2} \{ \sin[2\pi(f_i - f_o)t] + \sin[2\pi(f_i + f_o)t] \}. \end{aligned} \quad (6.19)$$

The high-frequency terms (i.e., the last term in the above two equations) will be neglected in the discussion because in actual design these terms will be filtered out through a lowpass filter.

If the input signal is higher than the local oscillator frequency ($f_i > f_o$) in the above equation, then

$$\begin{aligned} v_{if1} &= \cos[2\pi(f_i - f_o)t] \\ v_{if2} &= \sin[2\pi(f_i - f_o)t], \end{aligned} \quad (6.20)$$

where v_{if1} is leading v_{if2} by 90 deg. If the input signal is lower than the local oscillator frequency ($f_i < f_o$), the results from (6.19) will be

$$\begin{aligned} v_{if1} &= \cos[2\pi(f_i - f_o)t] \\ v_{if2} &= -\sin[2\pi(f_i - f_o)t], \end{aligned} \quad (6.21)$$

where v_{if1} is lagging v_{if2} by 90 deg. Therefore, by measuring the relative phase between the I and Q channels, the input frequency can be determined, even the IF bandwidth is only half of the input bandwidth.

If the IF bandwidth is wide and conventional operational amplifiers cannot accommodate the bandwidth, an RF amplifier can be used. As discussed before, an RF amplifier usually does not cover dc and very low frequency ranges. As a result, this approach will create a hole in the center of the band, which is highly undesirable.

Two input signals, one above and one below the f_o by the same frequency difference, will occupy the same frequency bin after downconversion. Under this condition, the phase relation will be disturbed and their frequencies may not be identified correctly.

Since two similar analog components are difficult to balance over a wide frequency range, it is very hard to build an analog I and Q downconverter with balanced outputs. However, highly balanced I and Q channels can be generated digitally through signal processing. These subjects are discussed in the following sections.

6.7 Digital Approach to Generate I and Q Channels

In the previous section, the I and Q channels are built through analog means. The advantage is wide bandwidth and the disadvantage is poor balance between the two channels. There are digital approaches that can be used to generate the desired I - Q channels. One approach is to use the Hilbert transform, the other is to use a special sampling scheme. In the digital approaches, the data of one channel (say the I channel) are obtained from a single-channel downconverter. The data of the Q channel have to be generated by processing the I channel data. Since the Q channel data are generated digitally, the unbalance between the outputs of the I and Q channels can be kept at a minimum.

One of the major disadvantages of the digital approach is the available operating speed. The bandwidth is limited to tens or a few hundreds of megahertz because of the slow processing speed. For EW applications, such a bandwidth would usually be considered to be too low. However, with an increase in digital operation speed, it might be possible to obtain useful bandwidth for EW applications.

The following sections will discuss the Hilbert transform in the continuous sense. Although the concept of Hilbert transform might be difficult, the mathematic definition appears straightforward. The actual computation is usually difficult to carry out. A popular way to obtain the Hilbert transform is through the Fourier transform (as in MATLAB[®]; MathWorks, Natick, MA).

6.8 Hilbert Transform [8, 10–13]

The Hilbert transform of a function $x(t)$ is defined as the convolution of $x(t)$ and a function $h(t)$. This relation can be shown mathematically in the following equation:

$$H[x(t)] \equiv x^h(t) = x(t) \otimes h(t) = x(t) \otimes \frac{1}{\pi t} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau, \quad (6.22)$$

where $\otimes$ represents convolution and $H[x(t)]$ and $x^h(t)$ represent a Hilbert transform in the time domain. The function $h(t)$ is defined as

$$h(t) = \frac{1}{\pi t}. \quad (6.23)$$

It is interesting to note that the Hilbert transform of a time domain function stays in the time domain. In the frequency domain, the Hilbert transform $X^h(f)$ can be written as

$$X^h(f) = X(f)H(f). \quad (6.24)$$

The Fourier transform of $h(t)$ can be obtained from (3.71) and (3.73) as

$$\mathcal{F}[h(t)] \equiv H(f) = -j \operatorname{sgn}(f) = -j \begin{cases} 1 & \text{if } f > 0 \\ -1 & \text{if } f < 0, \end{cases} \quad (6.25)$$

where sgn represents the sign function. Thus to obtain the Hilbert transform in the frequency domain, the negative frequency of $X(f)$ is multiplied by j and the positive frequency is multiplied by $-j$.

The $X(f)$ can be obtained from (6.24) as

$$X(f) = \frac{X^h(f)}{H(f)}. \quad (6.26)$$

However, one can see that $1/H(f) = -H(f)$ from (6.25). Thus

$$X(f) = -X^h(f)H(f)$$

or

$$x(t) = -x^h(t) \otimes h(t) = \frac{-1}{\pi} \int_{-\infty}^{\infty} \frac{x^h(\tau)}{t - \tau} d\tau. \quad (6.27)$$

Both the Hilbert transform and the inverse transforms are improper integrals because they have infinite discontinuity at $\tau = t$. In order to avoid this discontinuity, the integration should be carried out symmetrically about $\tau = t$. The integral can be written as

$$\int_{-\infty}^{\infty} \frac{g(\tau)}{t - \tau} d\tau = \lim_{\epsilon \rightarrow 0} \left[\int_{-\infty}^{t-\epsilon} \frac{g(\tau)}{t - \tau} d\tau + \int_{t+\epsilon}^{\infty} \frac{g(\tau)}{t - \tau} d\tau \right], \quad (6.28)$$

where ϵ is an infinitesimally small quantity. The Cauchy's principal value of this integral has to be used to calculate this integral. The result of the Hilbert transform can also be obtained from the inverse Fourier transform as

$$x^h(t) = x(t) \otimes h(t) = \mathcal{F}^{-1}[X(f)H(f)]. \quad (6.29)$$

Let us use an example to conclude this section. If the input signal is

$$x(t) = \sin(2\pi f_i t), \quad (6.30)$$

then its Fourier transform is

$$X(f) = \frac{j}{2} [\delta(f + f_i) - \delta(f - f_i)]. \quad (6.31)$$

This result is shown in Figure 6.8.

When $f < 0$, it is multiplied by j and the result is $-0.5\delta(f + f_i)$. When $f > 0$, it is multiplied by $-j$ and the result is $-0.5\delta(f - f_i)$. This result is shown in Figure 6.9.

If one is familiar with Fourier transforms, it is easy to recognize that the above result is the Fourier transform of $-\cos(2\pi f_i t)$. Thus one can write the following relation:

$$H[\sin(2\pi f_i t)] = -\cos(2\pi f_i t), \quad (6.32)$$

where $H[x(t)]$ represents the Hilbert transform. This equation shows that the Hilbert transform of a sine function is a negative cosine function. As a result, the phase of the input signal is shifted by $-j$.

If the input signal is a cosine wave, using an approach similar to the one above, the Hilbert transform will change it to a sine wave as

$$H[\cos(2\pi f_i t)] = \sin(2\pi f_i t). \quad (6.33)$$

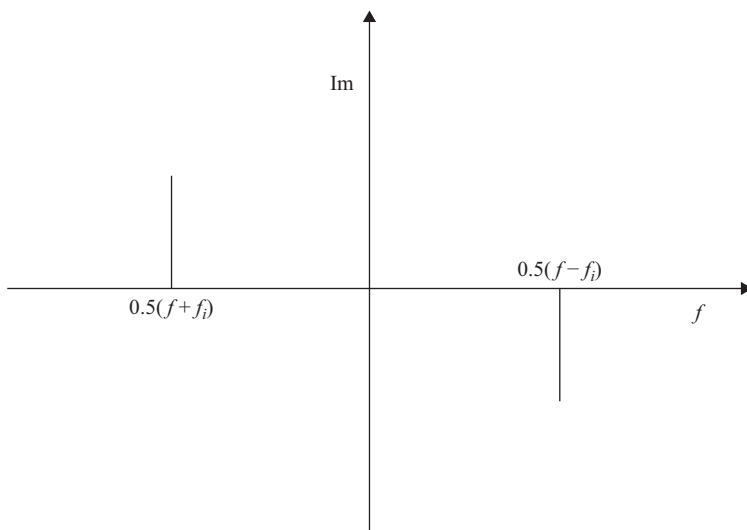


Figure 6.8 Fourier transform of a sine function.

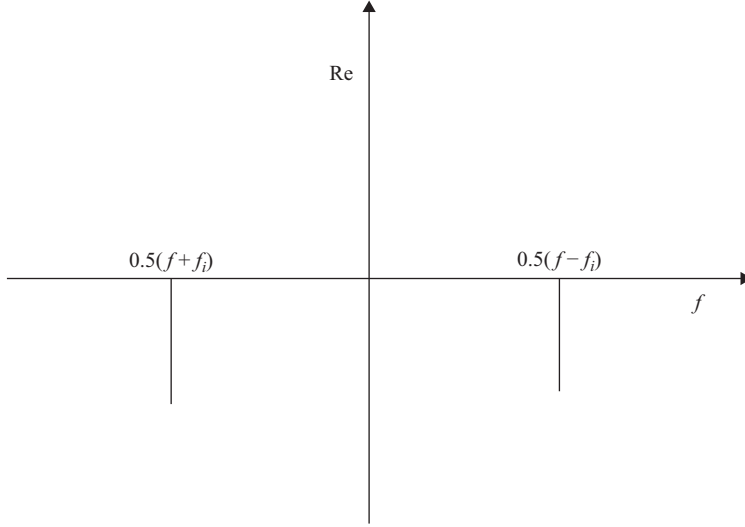


Figure 6.9 $X(f)H(f)$.

This relation also provides a phase shift of $-j$. Thus a Hilbert transform can provide a 90-deg phase shift without affecting the magnitude of the spectral component.

In this example, the Fourier transform of the input signal can be found easily. In a digital receiver, the input signal $x(t)$ is digitized and a fast Fourier transform (FFT) is used to determine $X(f)$. The Hilbert transform in the time domain $x^h(t)$ can be obtained from an inverse FFT as shown in (6.29) by using the definition of $H(f)$ in (6.25). However, the discrete Fourier transform is periodic in nature, which will affect the discrete Hilbert transform.

6.9 Discrete Hilbert Transform [10, 11, 14, 15]

In this section, the discrete Hilbert transform is discussed. The function $H(f)$ is extended from $-\infty$ to ∞ in the frequency domain. In the discrete domain, the number of data points must be limited (say, to $2M + 1$ points), which is equivalent to adding a rectangular window to the input signal. The z transform of $h(nt_s)$ is

$$H(z) = \sum_{n=-M}^M h(nt_s)z^{-n}$$

instead of

$$H(z) = \sum_{n=-\infty}^{\infty} h(nt_s)z^{-n}. \quad (6.34)$$

By definition, these two equations are not causal because the summation starts from a negative value. In order to implement the filter in practice, the above equations must be made causal.

The finite impulse response (FIR) filter design scheme will be used to achieve the discrete Hilbert transform as discussed in the following paragraphs.

1. Let us write the z transform of function $h(nt_s)$ as

$$\begin{aligned}
 H(z) &= \sum_{n=-\infty}^{\infty} h(nt_s) z^{-n} \\
 &= \sum_{n=-\infty}^{-1} h(nt_s) z^{-n} + h(0) + \sum_{n=1}^{\infty} h(nt_s) z^{-n} \\
 &= h(0) + \sum_{n=1}^{\infty} [h(-nt_s) z^n + h(nt_s) z^{-n}].
 \end{aligned} \tag{6.35}$$

Substituting $z = \exp(j2\pi ft_s)$, the result can be written as

$$\begin{aligned}
 H(e^{j2\pi ft_s}) &\equiv H_r(e^{j2\pi ft_s}) + jH_i(e^{j2\pi ft_s}) \\
 &= \sum_{n=-\infty}^{\infty} h(nt_s) e^{-j2\pi ft_s n} \\
 &= h(0) + \sum_{n=1}^{\infty} [h(-nt_s) \cos(2\pi nft_s) + jh(-nt_s) \sin(2\pi nft_s) \\
 &\quad + h(nt_s) \cos(2\pi nft_s) - jh(nt_s) \sin(2\pi nft_s)],
 \end{aligned} \tag{6.36}$$

where

$$\begin{aligned}
 H_r(e^{j2\pi ft_s}) &= h(0) + \sum_{n=1}^{\infty} [h(-nt_s) + h(nt_s)] \cos(2\pi nft_s) \\
 H_i(e^{j2\pi ft_s}) &= \sum_{n=1}^{\infty} [h(-nt_s) - h(nt_s)] \sin(2\pi nft_s).
 \end{aligned} \tag{6.37}$$

2. When the sampling frequency is f_s , the transfer function $H(f)$ is limited to the bandwidth of $f_s/2$. Due to the periodic property of sampling, the Hilbert transfer function is actually as shown in Figure 6.10. This function can be represented by the Fourier series as

$$H_i(e^{j2\pi ft_s}) = \sum_{n=1}^{\infty} b_n \sin(2\pi nft_s), \tag{6.38}$$

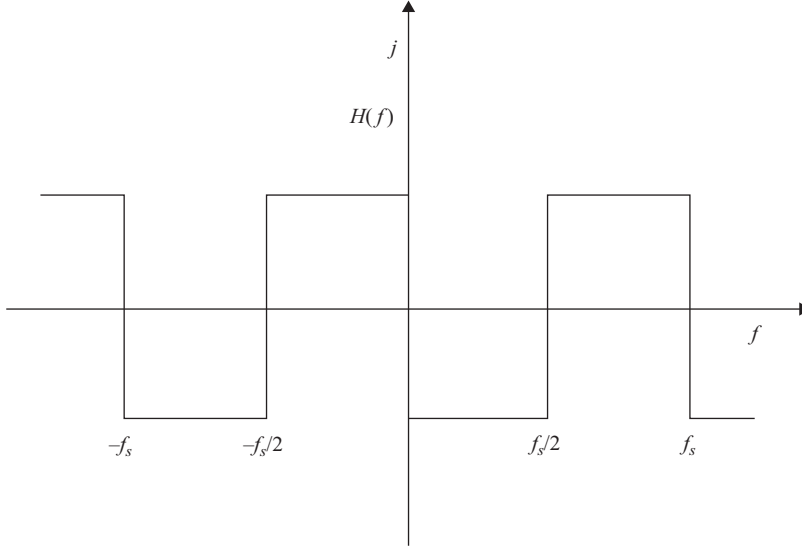


Figure 6.10 Periodical representation of $H(f)$.

where b_n can be found as

$$\begin{aligned}
 b_n &= \frac{2}{f_s} \int_{-f_s/2}^{f_s/2} H_i(e^{j2\pi n f t_s}) \sin(2\pi n f t_s) df \\
 &= \frac{2}{f_s} \left[\int_{-f_s/2}^0 \sin(2\pi n f t_s) df + \int_0^{f_s/2} -\sin(2\pi n f t_s) df \right] \\
 &= \frac{1}{n\pi} [-2 + 2 \cos(n\pi)] \\
 &= \begin{cases} 0 & n = \text{even} \\ \frac{-4}{n\pi} & n = \text{odd.} \end{cases}
 \end{aligned} \tag{6.39}$$

In (6.39), the relation of $f_s t_s = 1$ is used.

3. From the discussion of (6.25), the Hilbert transform in the continuous sense, it is obvious that the transfer function representing the Hilbert transform has only an imaginary part, therefore $H_r(f) = 0$ and $H_i(f) \neq 0$. This condition can be fulfilled if

$$h(0) = 0$$

and

$$h(-nt_s) = -h(nt_s) \tag{6.40}$$

Using this relationship, H_i in (6.37) can be written as

$$H_i(e^{j2\pi n f t_s}) = -2 \sum_{n=1}^{\infty} h(nt_s) \sin(2\pi n f t_s). \quad (6.41)$$

Comparing this result and (6.38), one can obtain

$$h(nt_s) = -\frac{b_n}{2}. \quad (6.42)$$

Using this relation and (6.39), one can find that

$$h(nt_s) = \begin{cases} 0 & n = \text{even} \\ \frac{2}{n\pi} & n = \text{odd} \end{cases} \quad (6.43)$$

$$h(-nt_s) = \begin{cases} 0 & n = \text{even} \\ -\frac{2}{n\pi} & n = \text{odd} \end{cases}$$

These are the desired $h(nt_s)$ values.

4. Causality—As mentioned before, the results obtained in (6.43) are not causal. In order to make the results causal, a simple shift in time domain can be used. The value of n in $h(nt_s)$ is windowed from $-M$ to M as shown in (6.34). The shift in time is equivalent to multiplying the first result in (6.34) by z^{-M} and substituting with $k = n + M$. The new result is

$$H(z) = \sum_{n=-M}^M h(nT)z^{-(n+M)} = \sum_{k=0}^{2M} h(kT - MT)z^{-k}. \quad (6.44)$$

This result is causal because the k value starts from zero instead of a negative value.

5. Windowing—Digitizing and limiting the input signal to a finite number of points is equivalent to providing a rectangular window in the time domain. Its equivalent effect in the frequency domain is convolving with a sinc function ($\sin x/x$), which will disrupt the phase relation and the output. In order to reduce the effect of the sinc function in the frequency domain, special windows $W(nt_s)$ are often used.

6.10 Examples of the Discrete Hilbert Transform

In this section, two examples will be used to demonstrate the discrete Hilbert transform. The first one uses an FIR filter with a rectangular window. The second one uses an FIR filter with a Hamming window. Let us choose an 11-lag filter to perform the Hilbert transform, which corresponds to $2M + 1 = 11$. The corresponding n value is from -5 to $+5$. The $h(n)$ values obtained from (6.43) are listed in Table 6.1. It is interesting to note that if $n = -6$ to 6 , the same results will be obtained because the two end values are zeros.

The second step is to convert $h(n)$ into a causal form. This can be accomplished by rearranging n from 0 to 10 as shown in row 3 of Table 6.1. The direct form of the filter is

Table 6.1 Standard $h(n)$ values.

n	-5	-4	-3	-2	-1	0	1	2	3	4	5
$h(n)$	$-2/5\pi$	0	$-2/3\pi$	0	$-2/\pi$	0	$2/\pi$	0	$2/3\pi$	0	$2/5\pi$
New n	0	1	2	3	4	5	6	7	8	9	10

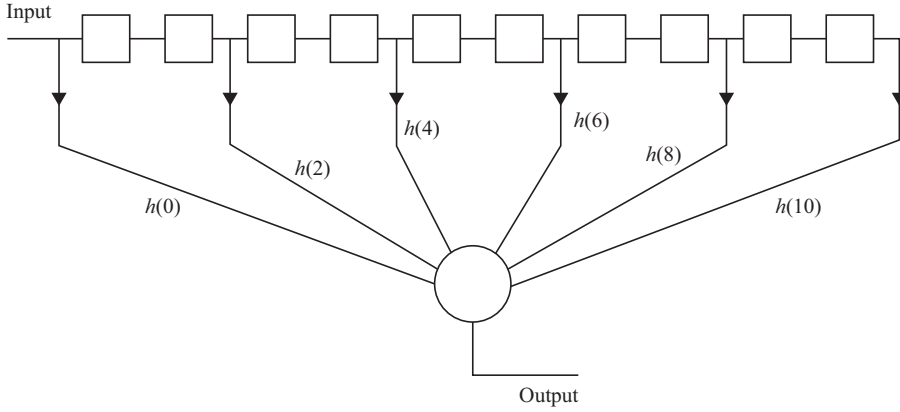


Figure 6.11 Direct form of the filter for the Hilbert transform.

shown in Figure 6.11. There are only six $h(n)$ values, but each has a delay time of two units. If the input to the filter is a sine wave, the output is obtained from the convolution of the input signal and $h(n)$. Figure 6.12(a) shows the input signal, with five data points removed from the beginning and end of the data to match the steady state of the filtered output. Figure 6.12(b) shows the output in the time domain.

Since the filter has a total delay of 10 units, it takes 10 units for the filter output to reach steady state, and the last 10 output data points are not accurate either because a portion of the input data is zeros. These data at the beginning and end of the output can be considered to be in transient states and are not included in the figure. In general, if the input is a pulse of n data points and the filter has m data points, the output will have $n + m - 1$ data points, and among them there are $n - m$ points in steady state.

The second example is adding a Hamming window to $h(n)$. The Hamming window is given in (3.111), thus

$$w(n) = 0.54 + 0.46 \cos \left[\frac{n\pi(n - N/2)}{N} \right], \quad (6.45)$$

where $N = 10$ and n is from 0 to 10. The filters $w(n)$ and $h(n)w(n)$ are listed in Table 6.2.

The output is shown in Figure 6.12(c). In this particular case, using a Hamming window makes little difference in the time domain. In the frequency domain, the Hamming window will limit the highest sidelobe to 43 dB from the mainlobe.

6.11 Narrowband *I* and *Q* Channels Through a Special Sampling Scheme [16–22]

Another way to build *I* and *Q* channels can be derived from a special sampling scheme. Let us start with a narrowband signal. If the input signal is real and narrowband with a known frequency f_i , it can be written as

$$V_i = A \sin(2\pi f_i t + \theta), \quad (6.46)$$

where A is the amplitude and θ is the initial phase of the input signal.

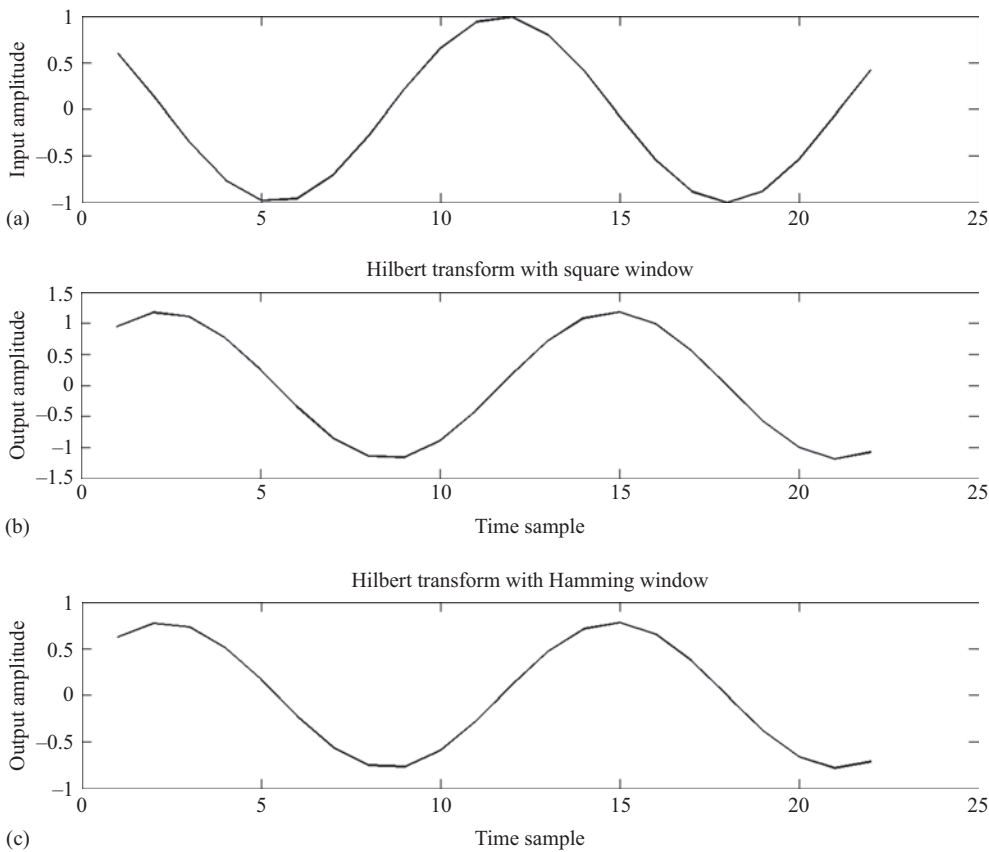


Figure 6.12 Input and output of a Hilbert transform filter: (a) input, (b) output (rectangular window), and (c) output (Hamming window).

Table 6.2 Filter values.

n	0	1	2	3	4	5	6	7	8	9	10
$h(n)$	$-2/5\pi$	0	$-2/3\pi$	0	$-2/\pi$	0	$2/\pi$	0	$2/3\pi$	0	$2/5\pi$
$w(n)$	0.080	0.168	0.398	0.682	0.912	1	0.912	0.682	0.398	0.168	0.080
$h(n)w(n)$	-0.010	0	-0.084	0	-0.581	0	0.589	0	0.084	0	0.010

If the sampling period t_s starts from zero and operates at twice the Nyquist sampling rate, or $4f_i$, which corresponds to $t_s = 1/4f_i$, the outputs starting from $t = 0$ are $V_i = A \sin \theta$, $A \cos \theta$, $-A \sin \theta$, $-A \cos \theta$, $A \sin \theta$, ... It is obvious that the odd and even terms are 90 deg out of phase. Thus one can treat the odd terms as the I channel and the even terms as the Q channel. This approach may be applicable to a narrowband system. In an EW system, the frequency of the input signal is unknown and hence this approach, in general, is not applicable.

6.12 Wideband I and Q Channels Through a Special Sampling Scheme [22, 23]

The following approach is somewhat similar to the scheme discussed in the last section, but it is for inputting a signal with unknown frequency. This concept can be explained through an analog approach. Let us use Figure 6.7(b) again. In that figure, the input signal is divided into two in-phase paths and the local oscillator signals applied to the two mixers are 90 deg out of phase. The outputs of the two mixers are

$$\begin{aligned}
 v_{if1} &= \sin(2\pi f_i t + \theta) \cdot \cos(2\pi f_o t) \\
 &= \frac{1}{2} \{ \sin[2\pi(f_i - f_o)t + \theta] + \sin[2\pi(f_i + f_o)t + \theta] \} \\
 v_{if2} &= \sin(2\pi f_i t + \theta) \cdot \sin(2\pi f_o t) \\
 &= \frac{1}{2} \{ \cos[2\pi(f_i - f_o)t + \theta] - \cos[2\pi(f_i + f_o)t + \theta] \}, \quad (6.47)
 \end{aligned}$$

where both the low and high frequencies are 90 deg out of phase. It is interesting to note that basically this is the same result as in (6.19). In analog design, at each output a lowpass filter is used to filter out the sum frequency term.

In the above equations, the local oscillator frequency f_o and its initial phase can be considered as known, but the input frequency f_i is unknown. Since the input phase and the sampling frequency of the mixers are known, one can choose the sampling time equal to $1/4f_o$ and start at zero phase. Under this condition, and substituting $t = 1/4f_o$ in (6.47), the two outputs are

$$\begin{aligned}
 V_{if1} &= \sin(\theta), 0, -\sin(2\pi f_i t_2 + \theta), 0, \sin(2\pi f_i t_4 + \theta) \\
 V_{if2} &= 0, \sin(2\pi f_i t_2 + \theta), 0, -\sin(2\pi f_i t_3 + \theta), 0. \quad (6.48)
 \end{aligned}$$

In this equation, $t_0 = 0$. According to (6.47), these two outputs are 90 deg out of phase. If these data are combined into one channel and arranged in chronological order in the time domain, neglecting the zeros, the result is

$$V_{if} = \sin(\theta), \sin(2\pi f_i t_1 + \theta), -\sin(2\pi f_i t_2 + \theta), \sin(2\pi f_i t_3 + \theta), \dots \quad (6.49)$$

These results can be obtained from a real signal. If the input signal is

$$V_{if} = \sin(2\pi f_i t + \theta) \quad (6.50)$$

the result in (6.49) can be obtained by simply making $t = t_0, t_1, t_2, t_3, \dots$, where $t_0 = 0$. The result is shown in Figure 6.13. In this figure the initial phase is assumed to be zero.

Now let us look at this problem from a reverse point of view. If a real signal is sampled at any time interval, the result will be the same as in (6.49). Zeros can be added into the sequence, which can then be separated into two channels to match the results in (6.48).

If the sampling frequency is f_s , from the Nyquist sampling theorem, the input bandwidth is limited to $f_s/2$. Based on the condition of obtaining (6.48), let us imagine that the frequency of the local oscillator is related to the sampling frequency as

$$f_o = f_s/4. \quad (6.51)$$

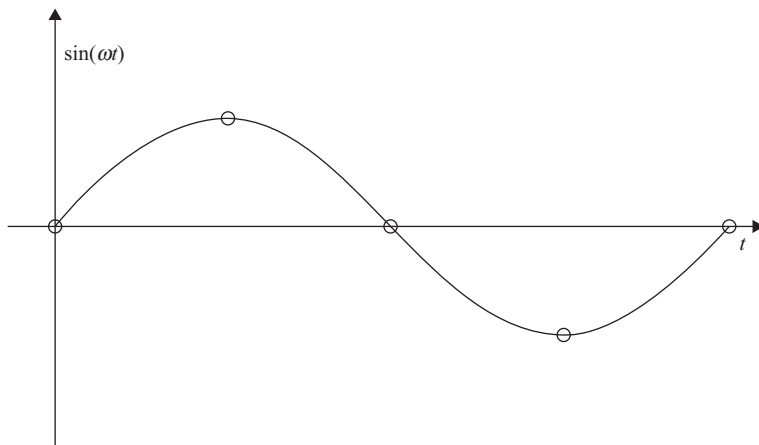


Figure 6.13 A sine wave is sampled at five points.

This approach will generate the results in (6.48), which can be converted into two channels that are 90 deg out of phase. The outputs of the two channels are

$$\begin{aligned} v_{if1} &= \frac{1}{2} \{ \sin[2\pi(f_i - f_o)t + \theta] + \sin[2\pi(f_i + f_o)t + \theta] \} \\ v_{if2} &= \frac{1}{2} \{ \cos[2\pi(f_i - f_o)t + \theta] - \cos[2\pi(f_i + f_o)t + \theta] \}. \end{aligned} \quad (6.52)$$

These are the same results as in (6.47), but not the desired results. The desired results are

$$\begin{aligned} v_{if1} &= \frac{1}{2} \sin[2\pi(f_i - f_o)t + \theta] \\ v_{if2} &= \frac{1}{2} \cos[2\pi(f_i - f_o)t + \theta]. \end{aligned} \quad (6.53)$$

In order to obtain these results, lowpass filters must be used to filter out the high-frequency terms, which are represented by the $f_i + f_o$ terms in (6.52). Since the input frequency is limited to $f_s/2$, the filter cutoff frequency should match this value.

Let us use an example to demonstrate this idea. If an input signal is sampled at 100 MHz, the input bandwidth is limited to 50 MHz. From (6.51), the corresponding local oscillator frequency is 25 MHz. If the outputs are $x(1)$, $x(2)$, $\dots$, $x(n)$, according to (6.48) these data can be divided into two groups with zeros added as

$$\begin{aligned} I(t) &= x(1), 0, -x(3), 0, x(5), 0, -x(7) \\ Q(t) &= 0, x(2), 0, -x(4), 0, x(6), 0. \end{aligned} \quad (6.54)$$

Both data must pass through a lowpass filter of $h(t)$ with the cutoff frequency at 50 MHz. Let us use a lowpass FIR filter to filter out the high-frequency components. The impulse response of the filter can be written as

$$h(t) = h(1), h(2), h(3), h(4), h(5), h(6), \dots \quad (6.55)$$

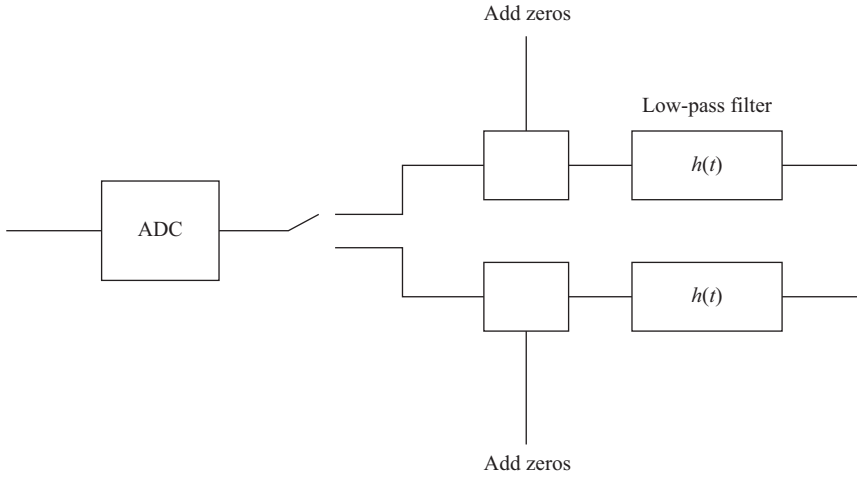


Figure 6.14 Digital mixing and filtering.

Two identical filters are needed, one for each output. The outputs of the *I* and *Q* channels can be obtained from the convolution of the input and the impulse response of the filter. The relation can be written as

$$\begin{aligned} y_I &= h(t) \otimes I(t) \\ y_Q &= h(t) \otimes Q(t). \end{aligned} \quad (6.56)$$

The above equation provides the desired results. This idea is represented in Figure 6.14. The input signal is sampled and separated alternatively into two data channels. At the output of each channel there is a lowpass filter. The next section will provide hardware considerations for the filter design.

6.13 Hardware Considerations for Filter Design for Wideband Digital *I-Q* Channels [23]

One important design requirement in implementing the approach in Section 6.12 is to reduce the operation speed of the lowpass filter. If the input data are sampled at 100 MHz, the filter must operate at the same speed in order to keep up with the input data rate. However, as shown in (6.54), half of the data are zeros. Taking these zeros into consideration, the filter operation speed can be reduced to $f_s/2$, or 50 MHz. The price one has to pay for such an approach is to double the number of filters. Instead of using one filter per output channel, two filters are required. The approach is discussed below.

If the results of (6.54) and (6.55) are substituted into (6.56), and from the definition of convolution, the results are

$$\begin{aligned}
 y_I(1) &= x(1)h(1) \\
 y_I(2) &= x(1)h(2) \\
 y_I(3) &= x(1)h(3) - x(3)h(1) \\
 y_I(4) &= x(1)h(4) - x(3)h(2) \\
 &\dots \\
 y_Q(1) &= 0 \\
 y_Q(2) &= x(2)h(2) \\
 y_Q(3) &= x(2)h(3) - x(4)h(1) \\
 y_Q(4) &= x(2)h(4) - x(4)h(2).
 \end{aligned} \tag{6.57}$$

In these equations, the outputs y_I and y_Q can be divided into odd and even time terms. The odd time term outputs are $y_I(1), y_I(3), \dots, y_Q(1), y_Q(3), \dots$, and the even term outputs are $y_I(2), y_I(4), \dots, y_Q(2), y_Q(4), \dots$. It is interesting to note that the odd outputs contain only odd terms of $h(t)$ (i.e., $h(1), h(3)$) and the even terms contain only even terms of $h(t)$ (i.e., $h(2), h(4)$). Let us use $h_o(t)$ and $h_e(t)$ to represent the odd and even terms of $h(t)$.

If $h(t)$ has N total coefficients, $h_o(t)$ and $h_e(t)$ will each contain $N/2$ coefficients. The output of I and Q can be subdivided into two channels and each channel will have a lowpass filter. There is a total of four filters, as shown in Figure 6.15. Since the filter in the I channel only operates on odd outputs (i.e., $x(1), x(3)$) and the Q channel filter only operates on even outputs (i.e., $x(2), x(4)$), the operating speed of these filters is only at half of the sampling frequency. In this approach zeros are not added to the output data, as shown in Figure 6.14.

A computer program is listed in Appendix 6.A to demonstrate this approach. The input is a sine wave. It should be noted that in (6.57), the data point $y_Q(1) = 0$. However, from the computer simulation, the first data point generated, $y_Q(1)$, is not zero. In order to make the result correct, a zero is added at the beginning of the $y_Q(t)$ to represent $y_Q(1)$. The outputs are shown in Figure 6.16. One should note that there are transient effects at both the input and the output of the pulse. Since each filter has a length of 16 points, the transient lasts the same length.

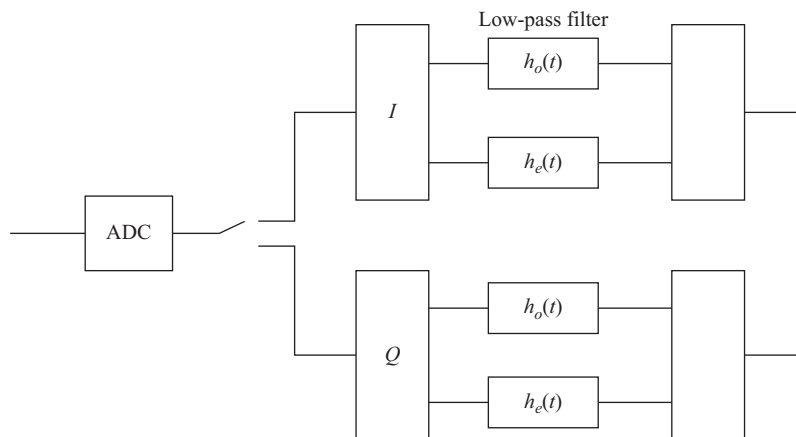


Figure 6.15 Digital mixing with modified filters.

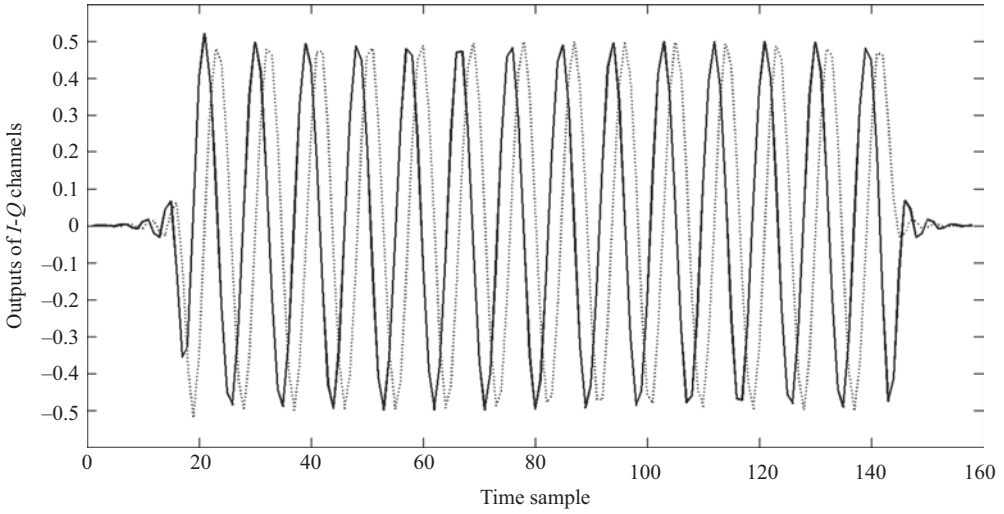


Figure 6.16 I-Q outputs from filters.

6.14 Digital Correction of I and Q Channel Imbalance

If the imbalance of two channels can be measured, they can be corrected. The correction scheme is through the Gram–Schmidt procedure, which can be stated as follows for a two-dimensional case [24, 25]: Any two vectors can be represented by two orthonormal vectors. This problem is solved in Churchill *et al.* [26] and the results are presented here.

In order to simplify this discussion, the amplitude error is included in the I channel and the phase error is included in the Q channel. Usually there is dc bias on the outputs and this bias must be removed from the outputs. After the dc bias is removed, the results can be written as

$$I_1 = (1 + \alpha)A \cos(2\pi f_{if}t)$$

and

$$Q_1 = A \sin(2\pi f_{if}t + \epsilon), \quad (6.58)$$

where α represents amplitude imbalance and ϵ represents phase imbalance. The corrected outputs can be written in matrix form as

$$\begin{bmatrix} I_2 \\ Q_2 \end{bmatrix} = \begin{bmatrix} E_1 & 0 \\ P & 1 \end{bmatrix} \begin{bmatrix} I_1 \\ Q_1 \end{bmatrix}, \quad (6.59)$$

where I_2 and Q_2 are orthogonal and balanced. Values of E_1 and P are solved as

$$\begin{aligned} E_1 &= \frac{\cos \epsilon}{1 + \alpha} \\ P &= \frac{-\sin \epsilon}{1 + \alpha}. \end{aligned} \quad (6.60)$$

If the values in (6.60) are substituted into (6.59), the results are

$$\begin{aligned} I_2 &= A \cos \epsilon \cos(2\pi f_{if} t) \\ Q_2 &= A \cos \epsilon \sin(2\pi f_{if} t). \end{aligned} \quad (6.61)$$

which indicate that the two outputs have the same amplitudes and they are 90 deg out of phase.

Since ϵ and α are quantities close to zero, E_1 is close to unity. It is conventional for digital arithmetic to make the scaling coefficient E_1 a small number. Thus a new E is often defined as $E_1 - 1$. The required correction coefficients for gain and phase become

$$\begin{aligned} E &= \frac{\cos \epsilon}{1 + \alpha} - 1 \\ P &= \frac{-\sin \epsilon}{1 + \alpha}. \end{aligned} \quad (6.62)$$

The following discussion is about finding the imbalance and from the collected data generating the correction coefficients in the above equation. This correction can be accomplished by using a testing signal of frequency f . The output of the I - Q channels can be written in complex form as

$$s_t(t) = (1 + \alpha) A \cos(2\pi ft + \psi) + a + j[A \sin(2\pi ft + \psi + \epsilon) + b], \quad (6.63)$$

where ψ is the initial phase of the input signal, ϵ is the imbalance in phase, and a and b are dc levels. The input frequency is sampled at frequency $f_s = 1/t_s$ where t_s is the unit sampling time. The sampling frequency must be four times the input frequency, which means $f_s = 4f$. Only four samples in the time domain are required to calculate the correction coefficients. They are

$$\begin{aligned} s_t(0) &= (1 + \alpha) A \cos \psi + a + j[A \sin(\psi + \epsilon) + b] \\ s_t(t_s) &= -(1 + \alpha) A \sin \psi + a + j[A \cos(\psi + \epsilon) + b] \\ s_t(2t_s) &= -(1 + \alpha) A \cos \psi + a + j[-A \sin(\psi + \epsilon) + b] \\ s_t(3t_s) &= (1 + \alpha) A \sin \psi + a + j[-A \cos(\psi + \epsilon) + b]. \end{aligned} \quad (6.64)$$

The corresponding FFT can be written as

$$S_t(k) = \sum_{n=0}^{N-1} s_t(t) e^{-j2\pi nk/N}. \quad (6.65)$$

Since the input frequency is f and the input is sampled at $4f$ and only four samples are obtained, the frequency domain and $S_t(k)$ has only four outputs. Three of the four frequency components are as follows: $S_t(0)$ is the dc component, $S_t(1)$ is the input frequency, and $S_t(3)$ is the image.

Substituting the results from (6.64) into this equation, the frequency domain output is

$$S_t(0) = 4(a + jb). \quad (6.66)$$

This information can be used to correct the dc bias.

The coefficients E and P for correction of gain and phase errors can be obtained from the frequency domain outputs at test signal frequency $S_t(1)$ and at the image frequency $S_t(3)$. The output at the test signal frequency is

$$S_t(1) = 2A[(1 + \alpha) + \cos \epsilon + j \sin \epsilon]e^{j\psi}. \quad (6.67)$$

The output of the filter at the image of the test signal frequency is

$$S_t(3) = 2A[(1 + \alpha) - \cos \epsilon + j \sin \epsilon]e^{-j\psi}. \quad (6.68)$$

It is easily shown that

$$S_t^*(1) + S_t(3) = 4A(1 + \alpha)e^{-j\psi}, \quad (6.69)$$

where $S_t^*(t)$ represents the conjugate of $S_t(t)$. From (6.68) and (6.69), one can find

$$\frac{2S_t(3)}{[S_t^*(1) + S_t(3)]} = 1 - \frac{\cos \epsilon}{1 + \alpha} + \frac{j \sin \epsilon}{1 + \alpha}. \quad (6.70)$$

From (6.62) and (6.69), the following results can be obtained:

$$\begin{aligned} E &= -\operatorname{Re} \left[\frac{2S_t(3)}{[S_t^*(1) + S_t(3)]} \right] \\ P &= -\operatorname{Im} \left[\frac{2S_t(3)}{[S_t^*(1) + S_t(3)]} \right]. \end{aligned} \quad (6.71)$$

These relations are used to estimate the correction factors. This correction method should be tested at different frequencies.

This correction can be applied to one frequency at a time. When the I and Q channels cover a wide bandwidth, the imbalance is a function of frequency. Since the signal frequency is unknown to the EW receiver, the imbalance compensation method discussed in this section might not be suitable. The wideband digital connection method is covered in the next section.

6.15 Wideband Digital Corrections of I and Q Channel Imbalance [27]

In Section 6.14, an imbalance compensation method based on a fixed imbalance value was described. An EW receiver needs to cover a broad bandwidth in which the amplitude and phase might be frequency dependent. To solve this issue, a wideband imbalance compensation scheme is necessary. Dr. Liou at the Air Force Research Laboratory developed a wideband imbalance compensation method for EW application [27] that will be introduced in this section.

Figure 6.17 is used to explain the amplitude and phase imbalance and its compensation. In Figure 6.17, a balanced complex signal is represented by vector $\mathbf{X}[\cos(\omega t), \sin(\omega t)]^T$, whose two components are in-phase and quadrature channels. The amplitude/phase imbalance caused by an imperfect downconverter is modeled by matrix $\mathbf{A}$. As shown in Figure 6.17, the in-phase and quadrature components of signal $\mathbf{Y}$ (output of matrix $\mathbf{A}$)

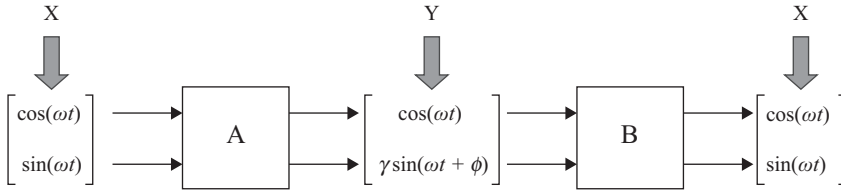


Figure 6.17 Amplitude and phase imbalance and its compensation [27].

have different amplitudes and their phase difference is not 90 deg. The relationship between $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{A}$ can be written as

$$\mathbf{Y} = \mathbf{A}\mathbf{X}, \quad (6.72)$$

where

$$\begin{aligned} \mathbf{X} &= [\cos(\omega t) \quad \sin(\omega t)]^T \\ \mathbf{Y} &= [\cos(\omega t) \quad \gamma \sin(\omega t + \phi)]^T. \end{aligned} \quad (6.73)$$

The matrix $\mathbf{A}$ can be derived as

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ \gamma \sin(\phi) & \gamma \cos(\phi) \end{bmatrix} \quad (6.74)$$

where γ and ϕ are amplitude and phase mismatches, respectively, at frequency ω .

An “inverse” matrix $\mathbf{B}$ is used to compensate for the imbalance. Let the input signal to the matrix $\mathbf{B}$ be the imbalanced signal $\mathbf{Y}$, and its output should be $\mathbf{X}$, the original balanced signal. The relationship between $\mathbf{Y}$, $\mathbf{X}$, and $\mathbf{B}$ is given by

$$\mathbf{X} = \mathbf{B}\mathbf{Y}. \quad (6.75)$$

The matrix $\mathbf{B}$ is the inverse of $\mathbf{A}$ and can be determined as

$$\mathbf{B} = \begin{bmatrix} 1 & 0 \\ -\tan(\phi) & 1/[\gamma \cos(\phi)] \end{bmatrix} \quad (6.76)$$

It should be pointed out that since the amplitude/phase imbalance is frequency dependent, the matrices $\mathbf{A}$ and $\mathbf{B}$ are also frequency dependent.

To convert the frequency domain vector presentation of imbalance compensation to a time domain complex-number operation, we first rewrite the $\mathbf{X}$ and $\mathbf{Y}$ vectors as

$$\mathbf{Y} = [I_{in} \quad Q_{in}]^T \quad \text{and} \quad \mathbf{X} = [I_{out} \quad Q_{out}]^T, \quad (6.77)$$

where I and Q with subscript “in” and “out” are the input and output in-phase and quadrature components, respectively. Let Z_{in} and Z_{out} be the input and output signals of the compensation system $\mathbf{B}$ in a complex-valued number representation:

$$Z_{in} = I_{in} + jQ_{in} \quad \text{and} \quad Z_{out} = I_{out} + jQ_{out}. \quad (6.78)$$

The $\mathbf{Y}$ and $\mathbf{X}$ in (6.77) can then be rewritten as

$$\mathbf{Y} = \begin{bmatrix} \frac{1}{2}(Z_{in} + Z_{in}^*) & \frac{1}{2j}(Z_{in} - Z_{in}^*) \end{bmatrix}^T \quad (6.79)$$

and

$$\mathbf{X} = \begin{bmatrix} \frac{1}{2}(Z_{out} + Z_{out}^*) & \frac{1}{2j}(Z_{out} - Z_{out}^*) \end{bmatrix}^T. \quad (6.80)$$

Substituting (6.79) and (6.80) into (6.75), the following relation can be derived:

$$Z_{out} = CZ_{in} + DZ_{in}^*, \quad (6.81)$$

where Z_{in}^* is the conjugate of Z_{in} and C and D are given by

$$C = \frac{1 - j\tan(\phi) + 1/(\gamma\cos(\phi))}{2} \quad (6.82)$$

$$D = \frac{1 - j\tan(\phi) - 1/(\gamma\cos(\phi))}{2} \quad (6.83)$$

C and D depend on the coupler's phase and amplitude mismatches at frequency ω .

Equation (6.81) is the “quasi-transfer” function of the imbalance compensator. It is called “quasi-transfer” because this function's input is a mixture of compensator input and its complex conjugate. Based on the frequency domain representation shown in (6.81) through (6.83), the two sets of time domain filter coefficients (one for the compensators input signal and the other for the compensator's input conjugate) can be derived by taking the inverse Fourier transform of the frequency-dependent C and D .

To implement the compensator with a digital filter, the values of C and D at discrete frequencies in the downconverter's working bandwidth are determined first. The inverse discrete Fourier transform of C and D at discrete frequency points are then taken. The input and output relations of the imbalance compensator in the time domain can be written as

$$z_{out,k} = \sum_{m=-M}^M c_m z_{in,k-m} + \sum_{m=-M}^M d_m z_{in,k+m}^*, \quad (6.84)$$

where the series of c and d are Fourier pairs of C and D , respectively, and $(2M + 1)$ is the tap number of FIR filters. The larger the filter tap number, the better the compensator performance at the expense of system complexity. The first term of the right-hand side of (6.84) is from the unconjugated term in (6.81) and the second term is from the conjugated term in (6.81). It should be noted that the second term has a positive time index in the summation instead of negative as seen in the first term. This is a result of the conjugated input.

To demonstrate the effectiveness of this algorithm, we consider a downconverter whose amplitude and phase imbalance are illustrated in Figure 6.18. The working bandwidth of the downconverter is assumed to be (0, 2.56 GHz). The amplitude and phase imbalance profile is arbitrarily chosen for the purpose of this demonstration. We used the amplitude and phase imbalance values in the downconverter's working bandwidth (resolution 20 MHz) to determine the value of vectors $\mathbf{C}$ and $\mathbf{D}$. The length of vectors $\mathbf{C}$ and $\mathbf{D}$ is 128 points. The inverse

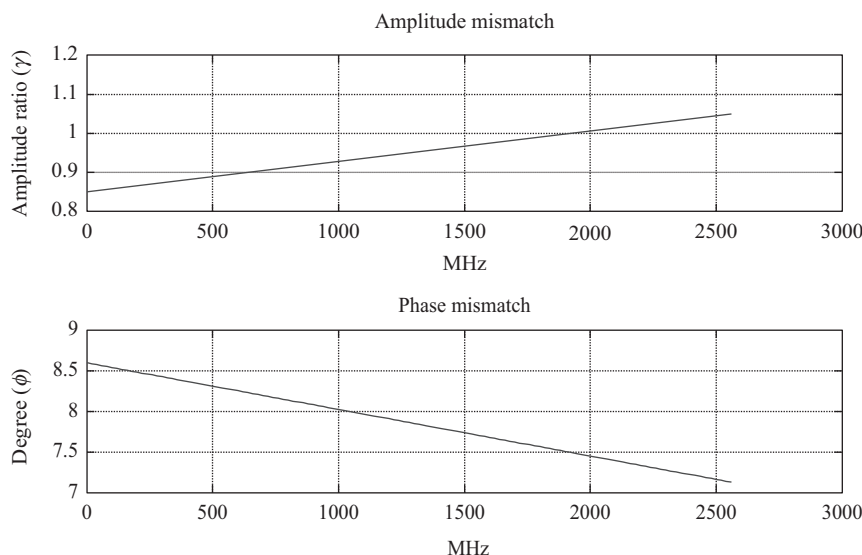


Figure 6.18 Downconverter's amplitude and phase imbalance spectrum used in the simulation.

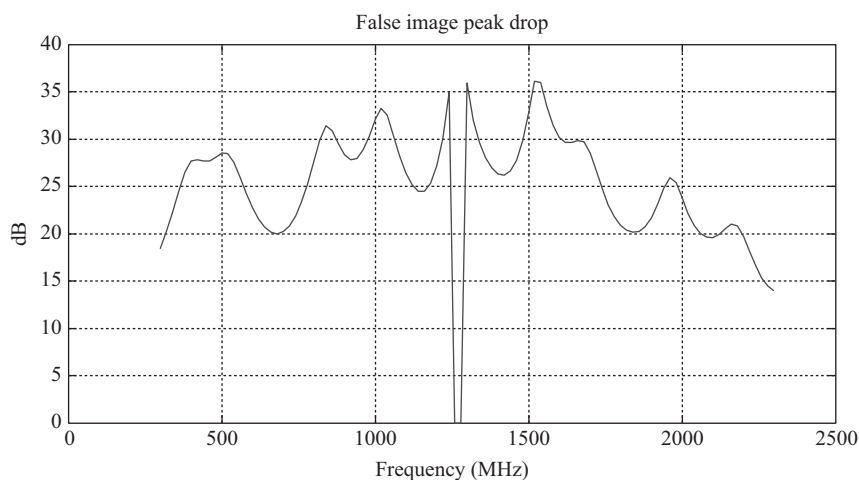


Figure 6.19 Image power drop across the downconverter's working bandwidth.

FFT of **C** and **D** are taken to get filter coefficients c and d . Since we would like to reduce the number of filter taps, only the values between index -5 and 5 are used (i.e., $M = 5$ in (6.84)). In other words, the filter length is 11.

We then use a cosine signal as the input signal and change its frequency so that the converter's output frequency changes from 300 MHz to 2.30 GHz. The outputs of down-converter are sampled at 2.56 GHz and the FFT is conducted upon the complex signals before and after the compensator. The transient period of the compensator output is removed before the FFT is conducted. The power difference between image signals before and after the compensator at different frequencies is calculated and the results are shown in Figure 6.19.

As shown in Figure 6.19, the compensator can successfully reduce image power, thus increasing the EW receiver's instantaneous dynamic range, except around 1.28 GHz (0 dB reduction), where the frequencies of the original signal and its image are identical.

References

- [1] East PW. 'Design techniques and performance of digital IFM'. *IEE Proceedings F: Communications, Radar and Signal Processing* 1982;**129**(3):154–163.
- [2] Tsui JBY. *Microwave Receivers with Electronic Warfare Applications*. New York: John Wiley & Sons; 1986.
- [3] Brown TT. 'Mixer harmonic chart'. *Electronic Buyers' Guide* 1953;**June**:R58–59.
- [4] Smith T, Wright J. *Spurious Performance of Microwave Double-Balanced Mixers*. Application note. Palo Alto, CA: Watkins-Johnson; August 1975.
- [5] Huang MY, Buskirk RL, Carlisle DE. 'Select mixer frequencies painlessly'. *Electronic Design* 1976;**24**(8):104–109.
- [6] Henderson BC. *Mixers: Part 1, Characteristics and Performance*. Tech-Notes. Palo Alto, CA: Watkins-Johnson; March/April 1981.
- [7] Henderson BC. *Predicting Intermodulation Suppression in Double-Balanced Mixers*. Tech-Notes. Palo Alto, CA: Watkins-Johnson; July/August 1983.
- [8] Papoulis A. *The Fourier Integral and its Applications*. New York: McGraw-Hill; 1962.
- [9] Engler H. Georgia Tech Research Institute, Atlanta, GA. Private communication.
- [10] Stremmler FG. *Introduction to Communication Systems*, 2nd ed. Reading, MA: Addison-Wesley; 1982.
- [11] Ziemer RE, Tranter WH, Fannin ER. *Signals and Systems: Continuous and Discrete*. New York: Macmillan; 1983.
- [12] Urkowitz H. 'Hilbert transforms of bandpass functions'. *Proceedings of the IRE* 1962;**50**:2143.
- [13] Bedrosian E. 'A product theorem for Hilbert transforms'. *Proceedings of the IEEE* 1963;**51**(5):868–869.
- [14] Rabiner LR. *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1975.
- [15] Oppenheim AV, Schaffer RW. *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1975.
- [16] Waters WM, Jarrett BR. 'Bandpass signal sampling and coherent detection'. *IEEE Transactions on Aerospace and Electronic Systems* 1982;**AES-18**(6):731–736.
- [17] Rice DW, Wu KH. 'Quadrature sampling with high dynamic range'. *IEEE Transactions on Aerospace and Electronic Systems* 1982;**AES-18**(6):736–739.
- [18] Rader CM. 'A simple method for sampling in-phase and quadrature components'. *IEEE Transactions on Aerospace and Electronic Systems* 1984;**AES-20**(6):821–824.
- [19] Waters WM, Jarrett BR. 'Tests of direct-sampling coherent detection with a laboratory analog-to-digital converter'. *IEEE Transactions on Aerospace and Electronic Systems* 1985;**AES-21**(3):430–432.

- [20] Mitchell RL. ‘Creating complex signal samples from a band-limited real signal’. *IEEE Transactions on Aerospace and Electronic Systems* 1989;**25**(3):425–427.
- [21] Liu H, Ghafoor A, Stockmann PH. ‘A new quadrature sampling and processing approach’. *IEEE Transactions on Aerospace and Electronic Systems* 1989;**25**(5):733–748.
- [22] Pellon LE. ‘A double Nyquist digital product detector for quadrature sampling’. *IEEE Transactions on Signal Processing* 1992;**40**(7):1670–1681.
- [23] Miniuk J, Shoemaker M. Naval Surface Warfare Center, Dahlgren, VA. Private communication.
- [24] Van Trees HL. *Detection, Estimation, and Modulation Theory, Part I*. New York: John Wiley & Sons; 1968.
- [25] Carlson AB. *Communication Systems: An Introduction to Signals and Noise in Electrical Communication*, 2nd ed. New York: McGraw-Hill; 1986.
- [26] Churchill FE, Ogar GW, Thompson BJ. ‘The correction of I and Q errors in a coherent processor’. *IEEE Transactions on Aerospace and Electronic Systems* 1981;**AES-17**(1):131–137.
- [27] Cheng C-H, Liou L, Lin D, Tsui J, Tai H-M. ‘Wideband in-phase/quadrature imbalance compensation using finite impulse response filter’. *IET Radar, Sonar & Navigation* 2014;**8**(7):797–804.

Appendix 6.A

```
% df8_16.m IQBK generates I and Q channels from fs/2
% clear
n1 = 128;
n=[0:n1-1];
% f = input('enter input frequency in MHz from 1 to 49 = ');
f=14*10^6;
x = sin(2*pi*f*1e-8*n+.01); % sampled at 100 MHz
% ***** generate I and Q channels *****
xx = reshape(x,4,n1/4);
xrm = [xx(1,1:n1/4); -xx(3,1:n1/4)];
xim = [xx(2,1:n1/4); -xx(4,1:n1/4)];
xr = reshape(xrm,1,n1/2);
xi = -1*reshape(xim,1,n1/2);
% ***** create a low pass filter *****
dbst = 70;
betat = .1102*(dbst-8.7);
windowt = kaiser(32,betat);
hkt = firl (31,0.5, windowt);
filtab = reshape(hkt,2,16);
filtat = filtab(1,1:16);
filtbt = filtab(2,1:16);
% ***** filtering *****
```

```

cnvlra = conv(filtat,xr);
cnvlrb = conv(filtbt,xr);
cnvlia = conv(filtat,xi);
cnvlib = conv(filtbt,xi);
I1 = [cnvlra; cnvlrb];
It = reshape(I1,1,2*length(cnvlra));
Q1 = [cnvlia; cnvlib];
Qt1 = reshape(Q1,1,2*length(cnvlra));
Qt = [0 Qt1(1:n1+30-1)]; % add 0 as the first point
% ***** plot the results *****
plot(1:n1+30,Qt, 1:n1+30,lt)
axis([0 160 -.6 .6])
xlabel('Time sample')
ylabel('Outputs of I Q channels')

```


Sensitivity and Detection Problems

7.1 Introduction

In this chapter, receiver sensitivity is discussed. The sensitivity of a receiver is defined as its ability to measure the weakest signal. The output of an analog-to-digital converter (ADC) is digital. This output impacts the measurement approach of input signals and thus receiver sensitivity. The conventional way to calculate sensitivity is through mathematical modeling. However, due to the nonlinear property ADCs, it is difficult to manipulate the calculations mathematically. As a result, the basic analysis is still based on an analog approach. If the ADC has a large number of bits, the result will be close to the analog approach. The threshold for detecting false alarms and signals in a digital receiver cannot be set arbitrarily in a continuous sense. The threshold can only be chosen from a finite number of levels.

This chapter discusses different types of detection schemes. The difference between the detection schemes of analog and digital receivers is emphasized. The possible advantages of a digital receiver from a detection point of view are discussed. The detection based on one data sample is discussed first. The one-sample case is then extended to multiple samples. Finally, frequency domain detection, which can be applied to digital receivers, is examined. Examples with computer programs are used to illustrate some of the detection schemes.

Before the general discussion, let us clarify the meaning of the word detection. This word has two distinct applications in a discussion of receivers. In the first sense, it refers to the process of converting a radio frequency (RF) signal or noise into a video signal. In the second sense, it refers to the process that determines whether a signal is present. In spite of this ambiguity, the meaning should be clear from the context.

7.2 Electronic Warfare (EW) Receiver Detection Approach

In this section, the detection procedure in an analog receiver is discussed. The detection is based on pulsed signals. Although this may not be true for all conventional receivers, such as communication receivers, it represents most EW receivers. In an analog receiver, the input RF signal is converted into a video signal by removing the RF through a crystal video detector. Sometimes the signal is converted into an intermediate frequency (IF) then converted to a video signal through a crystal video detector. A simple threshold comparator is used at the output of the video detector. If the video signal is higher than the threshold, the receiver will detect the signal as shown in Figure 7.1. This type of detection is achieved in the time domain because the video signal is in the time domain.

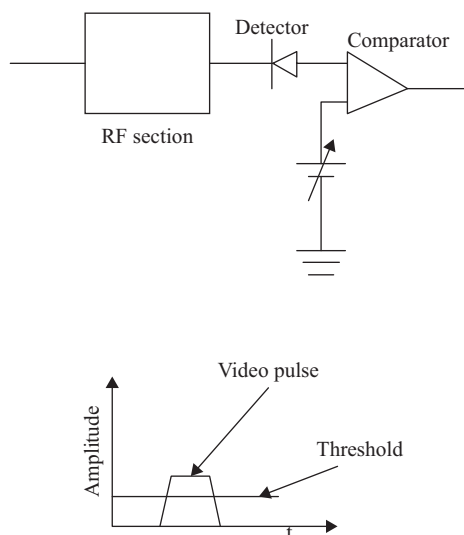


Figure 7.1 Detection circuit in an EW receiver.

In a radar receiver, since the input signal shape is known, it is possible to design the receiver with a matched filter to match the input signal and maximize receiver sensitivity. The receiver can also integrate many pulses to further improve receiver sensitivity.

In an EW receiver, since the input signal is unpredictable, it is impossible to design a receiver with a matched filter to obtain an optimum sensitivity. Usually an EW receiver is designed to match the shortest pulse anticipated. For example, if an EW receiver is designed to intercept pulses with a 100-ns minimum pulse width, the minimum filter bandwidth that determines the frequency data resolution is about 10 MHz (or 1/100 ns). The sensitivity of the receiver is determined by the noise floor, the noise figure (discussed in Chapter 5), and the threshold of the system. This minimum filter bandwidth determines the noise floor, thus determining the sensitivity of the receiver. Any signal with a pulse width of more than 100 ns will be detected with this sensitivity. Theoretically a receiver can be designed to intercept a longer pulse with narrow bandwidth to produce higher sensitivity. It is desirable to design an EW receiver with variable sensitivity (i.e., for a longer input pulse, the receiver will have a higher sensitivity).

An EW receiver is usually designed to receive the input signal on a pulse-by-pulse basis and convert the information on a pulse into a pulse descriptor word (PDW). This subject was discussed in Chapter 2. The PDWs of the received pulses are compared through a digital signal processor to deinterleave the pulses into a pulse train emitted by a certain radar. Thus it is unusual to design an EW receiver with multiple pulse integration to improve sensitivity.

7.3 Potential Detection Advantage in a Digital EW Receiver

In a digital receiver, the IF output is digitized at very high speed. Within one pulse, many data points can be collected. A fast Fourier transform (FFT) can be applied to the data to convert the ADC output to the frequency domain. Therefore the output can be detected either

in the time domain or in the frequency domain. The potential detection in both the time and frequency domains is discussed in the following sections.

7.3.1 Frequency Domain Detection

In this type of detection, FFT operation is performed on all the input data. In order to perform the FFT, the length of the input data points must be predetermined (e.g., 64 points chosen as a group). The FFT can be performed on the data points in a nonoverlapping mode as shown in Figure 7.2(a) or in a 50% overlapping mode as shown in Figure 7.2(b). This operation is sometimes called the short time Fourier transform (STFT) [1] and is discussed in detail in Chapter 10.

The frequency domain detection can be defined as the detection performed on each individual FFT output. Strictly speaking, if this detection is performed on a sequence of FFT outputs, one can consider that the detection is in the time domain. However, in this chapter let us limit the frequency domain detection to detection from one FFT output.

More overlapping in consecutive FFT outputs requires more operations, but less overlapping may lose information. For example, if a pulsed signal is 64 data points long and divided into two nonoverlapping consecutive data sets, the FFT outputs will be spread into several frequency bins and the amplitude will be lower than the expected value. Thus the sensitivity of the receiver will be lower. The highest sensitivity can be obtained by sliding the processing window across the data one point at a time, as shown in Figure 7.2(c). This approach is called a sliding FFT. The FFT outputs are compared with a certain threshold to determine whether there are signals, and the number of signals is then determined. In this approach, sidelobes must be neglected or eliminated.

The major advantage of frequency domain detection is that the FFT operating on a large number of data points pulls the signal out of the noise. The disadvantages are that many FFT

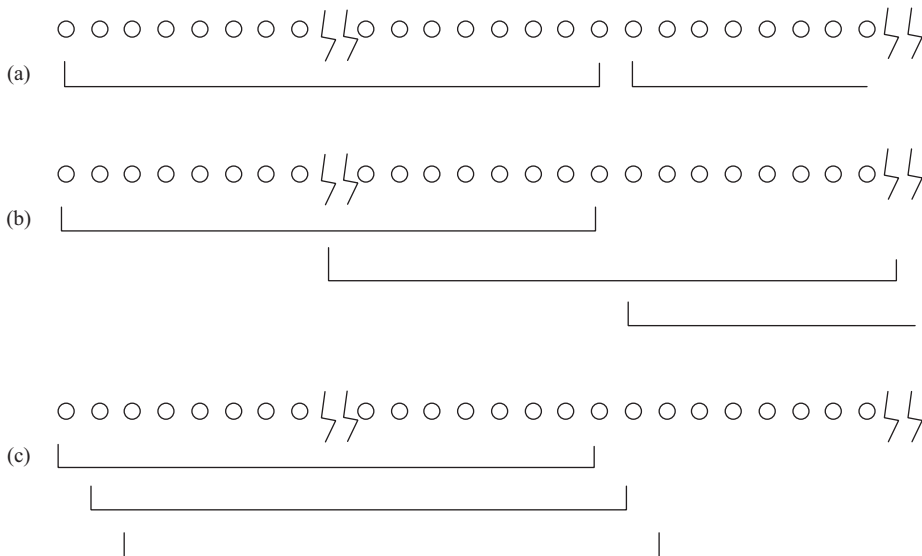


Figure 7.2 Data points grouping: (a) nonoverlapping, (b) 50% overlapping, and (c) one-point sliding.

operations are required and the FFT length is predetermined. In other words, the FFT speed must match the sampling speed. Even if the data points contain only noise, the FFT must be performed. Since the length of the FFT is predetermined, when the signal length does not match the FFT, the receiver cannot achieve optimum sensitivity. It is theoretically possible to perform the FFT with many different data lengths. However, this approach may excessively increase the complexity of signal processing.

7.3.2 Time Domain Detection

A simple time domain detection is relatively easy to perform. The amplitude of the signal is compared with a fixed threshold to determine whether there are signals in the data. Assume that the signal is collected in complex form. The amplitude of the signal can be obtained as

$$\begin{aligned}x_r &= A \cos(2\pi ft) \\x_i &= A \sin(2\pi ft) \\A &= \sqrt{x_r^2 + x_i^2},\end{aligned}\tag{7.1}$$

where x_r and x_i are the real and imaginary parts of the input signal and A and f are the amplitude and frequency of the input signal, respectively.

Obviously the time domain detection is easier than the frequency domain detection mentioned above. If the output data from the ADC can be detected first in the time domain, the FFT can be applied only to data strings with signals to obtain frequency information. When there is only noise in the data, FFT processing on the data can be avoided. If this scheme can be effectively implemented, the FFT speed may not be required to match the ADC sampling speed. However, the sensitivity of the receiver is determined by the time domain detection scheme.

In a digital receiver, the number of data points that can be obtained on one pulse depends on the pulse width (PW) and the sampling rate (f_s). Each individual data point can be compared with a threshold to obtain a sensitivity. However, the sensitivity can be improved by processing many data points together. The number of data points to be processed together must also be predetermined. However, since comparison in the time domain is rather simple, many different data lengths may be selected. If the input PW is close to one of these predetermined data lengths, the detection should be close to the matched filter for rectangular pulse shapes and high sensitivity can be achieved. Although this approach also increases the complexity of signal processing, it is more likely to be implemented than implementing FFT with various lengths.

Before the detection scheme can be meaningfully discussed, the threshold must be determined. The threshold setting determines the false alarm rate, which must be specified by the user of the receiver. First, the false alarm of the receiver is discussed.

7.4 False Alarm Time and Probability of False Alarm for One Data Sample

In an EW receiver, the false alarm time is often specified. The false alarm time is the average time the receiver takes to produce a false alarm when there is no input signal. For example, if

the false alarm time is 100 s, it means the receiver will produce on average one false alarm every 100 s. If the ADC sampling rate is f_s and the probability of false alarm on one single sample is P_{fas} , the false alarm time T_f can be written as

$$T_f = \frac{1}{P_{fas}f_s}$$

or

$$P_{fas} = \frac{1}{T_f f_s}. \quad (7.2)$$

In the above equation, it is assumed that every output sample of the ADC is compared with the threshold. If any output crosses the threshold, it is considered a false alarm. If the sampling frequency $f_s = 100$ MHz, in order to generate one false alarm every 100 s, the probability of false alarm $P_{fas} = 1 \times 10^{-10}$ is obtained from (7.2). Once the allowable probability of false alarm has been determined, the next step is to set the threshold at the output to generate such a probability of false alarm.

7.5 Threshold Setting for One Data Sample [2–6]

In this section, only one data point is compared with a fixed threshold. The probability density of noise $p(x)$ is assumed Gaussian, thus

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}, \quad (7.3)$$

where σ^2 is the variance of the noise. In order to generate an envelope for the input signal, I and Q channels are required. If (7.3) represents the output noise of the I channel, the Q channel can be similarly represented by

$$p(y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y^2}{2\sigma^2}}. \quad (7.4)$$

The probability density of the envelope is the product $p(x)p(y)$, which can be written as a function of r as

$$p(r) = \int_0^{2\pi} r p(x)p(y) d\phi = \int_0^{2\pi} \frac{r}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} d\phi = \frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}}, \quad (7.5)$$

where $r^2 = x^2 + y^2$ and $\phi = \tan^{-1}(y/x)$. This probability density function is referred to as the Rayleigh distribution.

The probability of false alarm can be written as

$$P_{fas} = \int_{r_1}^{\infty} p(r) dr = \int_{r_1}^{\infty} \frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}} dr = e^{-\frac{r_1^2}{2\sigma^2}}, \quad (7.6)$$

where r_1 is the threshold. The above equation can be written in a slightly different form as

$$-\frac{r_1^2}{2\sigma^2} = \ln(P_{fas})$$

or

$$r_1 = \sqrt{-2\sigma^2 \ln(P_{fas})}. \quad (7.7)$$

The value obtained from this equation, in general, will not coincide with one of the quantization levels of an ADC. One should choose a quantization level close to r_1 . A higher value will reduce the probability of false alarm and sensitivity, while a lower value will have the opposite effect.

In the above equation, if r_1 is chosen to match one of the quantization levels of the ADC and P_{fas} is determined by (7.2), the only variable that can be changed is σ . As discussed in Chapter 5, the noise is the combination of the quantization noise from the ADC and the noise from the RF amplifier in front of the ADC. Thus, from (5.41) and (5.49), the variance of the noise is

$$\sigma^2 = (1 + M) \frac{Q^2}{12R}. \quad (7.8)$$

In this equation, M is the ratio of the amplifier output noise to the quantization noise. From (5.58), one can see that the sensitivity of the receiver is degraded by

$$F_s - F = 10 \log \left(\frac{1 + M}{M} \right) \text{dB}, \quad (7.9)$$

where F_s and F are the noise figures of the amplifier chain without and with the ADC, respectively. If the M value is small, the sensitivity of the receiver will suffer.

If (7.8) is substituted into (7.7), the result is

$$\frac{r_1^2}{2\sigma^2} = \frac{(nQ)^2}{2(1 + M) \frac{Q^2}{12}} = -\ln(P_{fas}). \quad (7.10)$$

In this equation, r_1 is replaced by nQ , where n is an integer representing the threshold level.

The M value can be obtained as

$$M = - \left[\frac{6n^2}{\ln(P_{fas})} + 1 \right]. \quad (7.11)$$

The value of M can be adjusted by varying the amplifier gain in front of the ADC.

7.6 Probability of Detection for Single-Sample Detection [2–6]

When there is a signal in the data, the probability density function from the outputs of the I and Q channels can be written as

$$\begin{aligned} p(x) &= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu_x)^2}{2\sigma^2}} \\ p(y) &= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y-\mu_y)^2}{2\sigma^2}}, \end{aligned} \quad (7.12)$$

where μ_x and μ_y are the means of the Gaussian distribution and they are related to the input signal as

$$\begin{aligned} \mu_x &= A \cos \alpha \\ \mu_y &= A \sin \alpha, \end{aligned} \quad (7.13)$$

where α is the initial phase of the signal. Similarly x and y are related to r as

$$\begin{aligned} x &= r \cos \phi \\ y &= r \sin \phi. \end{aligned} \quad (7.14)$$

The joint density function is

$$\begin{aligned} p(r, \phi/\alpha) &= rp(x)p(y) \\ &= \frac{r}{2\pi\sigma^2} e^{-\frac{[r^2+A^2-2rA(\cos\alpha \cos\phi+\sin\alpha \sin\phi)]}{2\sigma^2}} \\ &= \frac{r}{2\pi\sigma^2} e^{-\frac{[r^2+A^2-2rA \cos(\alpha-\phi)]}{2\sigma^2}}. \end{aligned} \quad (7.15)$$

Integrating over ϕ , the probability density is

$$\begin{aligned} p(r|\alpha) &= \int_0^{2\pi} p(r, \phi|\alpha) d\phi \\ &= \frac{r}{\sigma^2} e^{-\frac{r^2+A^2}{2\sigma^2}} I_0\left(\frac{rA}{\sigma^2}\right) = p(r), \end{aligned} \quad (7.16)$$

where $I_0(x)$ is the modified Bessel function of zero order. It should be noted that this equation is independent of α , thus it is represented by $p(r)$. This distribution is referred to as the Rician distribution.

The probability of detection P_{ds} for a single sample can be calculated from

$$P_{ds} = \int_{r_1}^{\infty} p(r) dr = 1 - \int_0^{r_1} p(r) dr, \quad (7.17)$$

where r_1 is the threshold. The second portion of the integral is often calculated because the limit ∞ can be avoided and better accuracy can be expected. Unfortunately the above integration cannot be evaluated analytically. Only numerical integration can be used to obtain the result.

This result is for a single digitized data point. Sometimes the false alarm can be based on multiple data points, thus the detection problem must also be analyzed for multiple data points. The following sections discuss the multiple sample detection problem.

7.7 Detection Based on Multiple Data Samples

The sampling speed is usually in the hundreds of megahertz and the corresponding sampling time is a few nanoseconds. It is not necessary to make a decision based on one data sample. A more reasonable way is to detect the input signal through the entire PW, which contains multiple data samples. In order to detect the input signal based on multiple data samples, a new set of equations must be derived. These are the probability of false alarm and the probability of detection based on multiple samples. These relations can be obtained from equations of a single data point (discussed in the previous sections). Several detection approaches are discussed in this section.

The relation between the false alarm time and the probability of false alarm given in (7.2) can be modified to accommodate multiple samples. Suppose a total of N samples are used to determine the probability of false alarm and the probability of detection. In order to keep the desired false alarm time, the probability of false alarm can be increased by the factor N , because instead of making a decision every sampling time, $t_s = 1/f_s$, it makes a decision every Nt_s times. Thus the probability of false alarm can be written as

$$P_{fam} = \frac{N}{T_{fa} f_s}, \quad (7.18)$$

where P_{fam} is used to represent the probability of false alarm for multiple samples. Use of multiple samples to set the threshold is similar to narrowing the video bandwidth in an analog microwave receiver. The equivalent video bandwidth can be considered as f_s/N . In an analog receiver, the video bandwidth must match the minimum anticipated PW. In a digital receiver, the equivalent video bandwidth should also match the minimum PW. However, if one can design a receiver with several different N values, the receiver can match many different PWs. This approach will improve the sensitivity of the receiver on various PWs.

The next question to be answered is how to make a decision on every N sample. Two approaches will be presented:

1. Use the result derived from the single detection scheme to form an L -out-of- N approach. Under this condition, a certain quantization level will be used as the threshold. By selecting an L value that is smaller than N , if within the N samples L or more samples cross the threshold, then an input signal is detected.
2. The second approach is based on the sum of N samples. This approach is more akin to traditional radar detection where the sum over N pulses is used to make a decision. Both approaches are discussed in the following sections.

7.8 Detection Scheme for Multiple Samples (L -out-of- N) [7–9]

In this section, the L -out-of- N approach is discussed. First, a given false alarm time is used to generate the probability of false alarm P_{fa} . From the value of P_{fa} , a threshold is selected.

This false alarm can be treated as a binomial distribution problem. In order to generate the probability of false alarm from N consecutive samples, a total number of L or more samples must cross the threshold. If the probability of one sample crossing the threshold (a certain quantization level) is p , the probability of not crossing the threshold is $(1 - p)$. The probability of exactly L samples crossing the threshold is [7-9]

$$p(L|N) = \frac{N!}{L!(N-L)!} p^L (1-p)^{N-L}, \quad (7.19)$$

where $p(L|N)$ represents the probability that L samples are above the threshold.

The $p(L|N)$ value in the above equation can be used to represent either the false alarm rate or the probability of detection, depending on the p value. If p_{fas} is used to replace p , $p(L|N)$ represents the probability of false alarm. If p_{ds} is used to replace p , $p(L|N)$ represents the probability of detection.

Let us use $p(L + |N)$ to represent the probability that at least L samples cross the threshold (i.e., from L to N samples crossing the threshold). The probability $p(L + |N)$ can be written as

$$p(L + |N) = \sum_{i=L}^N p(i|N). \quad (7.20)$$

This equation can be used to determine the desired probabilities, either false alarm or detection.

The calculation of the probability of detection from multiple samples is quite similar to the approach mentioned in the previous section. First, the desired P_{fam} must be determined. Equations (7.19) and (7.20) will be used to match the desired P_{fam} . However, there are many combinations of L and N values, and it is difficult to solve for unique values of L , N , and P_{fas} . As a result, it is suggested to first select an N value and a value of P_{fas} . From (7.19) and (7.20), find a value of L that will generate a P_{fam} value close to the desired value. In this approach, the exact value of P_{fam} cannot be achieved, but a value close to it can be obtained. Examples are shown in Section 7.13 to illustrate the use of these equations.

7.9 Probability Density Function and Characteristic Function [9-13]

The next detection approach is to sum the output of N data points and compare the result with one threshold. In order to perform this analysis, the probability density functions of these N data points must be formulated. These probability functions are used to produce the false alarm and detection probabilities. Finally, the detection scheme is discussed. In this section, the relation of the probability density function and characteristic function is presented. This relation is used to produce probability density functions.

In order to determine the probability density functions of a sum of random variables, it is convenient to define a characteristic function. The characteristic function is defined as

$$C(\omega) \equiv \int_{-\infty}^{\infty} p(x) e^{j\omega x} dx, \quad (7.21)$$

where $p(x)$ is the probability density function and ω is an arbitrarily chosen variable. The function is identical to the inverse Fourier transform, thus the properties of the Fourier transform can be used in the following discussion.

The probability density function of a sum of n mutually independent random variables is given by the $n - 1$ -fold convolution of their individual probability density function $p_i(x)$, which can be written mathematically as [9]

$$p_n(x) = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p_{n-1}(x - x_{n-1}) \dots p_2(x - x_2) p_1(x_1) dx_1 dx_2 \dots dx_{n-1}, \quad (7.22)$$

where x_i are dummy variables. From Section 3.4, one should realize that the above equation is the convolution of $p_1, p_2, \dots$ in the x domain. Convolution in the x domain is equivalent to the product of their characteristic functions in the ω domain as

$$C_n(\omega) = C_{n-1}(\omega) \dots C_2(\omega) C_1(\omega) \quad (7.23)$$

since $C(\omega)$ is the inverse Fourier transform of $p(x)$ as shown in (7.21). The inverse transform of this equation gives the desired combined probability density function as

$$p_n(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} C_n(\omega) e^{-j\omega x} d\omega. \quad (7.24)$$

It is often easier to use the characteristic function to obtain the desired probability density function than carry out the integral shown in (7.22).

7.10 Probability Density Function of Sum Samples with a Square Law Detector [2, 9–13]

In Marcum [2] it appears that the mathematical analysis for an envelope detector is more complicated than that for a square law detector. However, the results from these two detectors are very close. In a square law detector, the output samples are obtained from a squaring operation of the digitized data. The square law detector is presented here and the result should be applicable to both envelope and square law detectors.

This discussion is again based on Marcum [2]. When there is only noise, the probability density function in (7.5) will be used here. For a square law detector, $z = r^2$ can be used to replace r , and $dz/dr = 2r$. This relation between $p(z)$ and $p(r)$ can be written as

$$p(z) dz = p(r) dr$$

or

$$p(z) = p(r) \frac{dr}{dz} = \frac{p(r)}{dz/dr}. \quad (7.25)$$

Thus the corresponding probability density function $p_{1n}(z)$ is

$$p_{1n}(z) = \frac{p(r)}{2r} = \frac{1}{2\sigma^2} e^{-\frac{r^2}{2\sigma^2}} = \frac{1}{2\sigma^2} e^{-\frac{z}{2\sigma^2}}. \quad (7.26)$$

This is the probability density of a square law detector. The subscript $1n$ of $p(z)$ represents one data point with only noise as the input. In order to find the probability density of N samples, the characteristic function will be used.

It is important to note that the integration presented in the following sections is difficult to perform. However, all the integrals can be found in Campbell and Foster [12]. In order to match the formulas in the reference, the following changes are made. In the conventional sense, the characteristic equation is defined as the inverse Fourier transform as shown in (7.22). The probability density function is obtained through the Fourier transform. In the following sections, the characteristic function is redefined as the Fourier transform and the probability density function is obtained through the inverse Fourier transform. Since the operations are performed in pairs, the result will be the same.

The characteristic function of (7.26) can be written as

$$C_{1n}(\omega) = \int_0^{\infty} \frac{1}{2\sigma^2} e^{-\frac{z}{2\sigma^2}} e^{-j\omega z} dz = \frac{1}{j2\omega\sigma^2 + 1}. \quad (7.27)$$

If N samples are used to determine the overall probability density function, the characteristic function can be written as

$$C_{Nn} = [C_{1n}(\omega)]^N = \left[\frac{1}{j2\omega\sigma^2 + 1} \right]^N = \left(\frac{1}{2\sigma^2} \right)^N \left[\frac{1}{j\omega + \frac{1}{2\sigma^2}} \right]^N. \quad (7.28)$$

The last expression can be used directly from the inverse Fourier transform formula (no. 431, p. 44 [12]). The result is

$$P_{Nn}(z) = \frac{z^{N-1} e^{-\frac{z}{2\sigma^2}}}{(2\sigma^2)(N-1)!}. \quad (7.29)$$

The subscript Nn represents the probability density of N data points of data with only noise. This equation will be used to find the false alarm rate.

If there is a signal present, the probability density function can be obtained from (7.16) in a similar manner as from (7.25) and (7.26). The result is

$$p_{1s}(z) = \frac{1}{2\sigma^2} e^{-\frac{z+A^2}{2\sigma^2}} I_0\left(\frac{\sqrt{z}A}{\sigma^2}\right). \quad (7.30)$$

This is the probability density of one sample from a square law detector with a signal. The characteristic function $C_{1s}(\omega)$ of this equation can be written as in Campbell and Foster (no. 655.1, p. 79 [12]). The Fourier transform result is

$$C_{1s}(\omega) = \frac{1}{2\sigma^2} e^{-\frac{A^2}{2\sigma^2}} \int_0^{\infty} e^{-\frac{z}{2\sigma^2}} I_0\left(\frac{A\sqrt{z}}{\sigma^2}\right) e^{-j\omega z} dz = \frac{1}{j2\omega\sigma^2 + 1} e^{-\frac{jA^2\omega}{j2\omega\sigma^2 + 1}}, \quad (7.31)$$

where the subscript 1s is a single data point with a signal. For the N sample case, the characteristic function can be written as

$$C_{Ns}(\omega) = [C_{1s}(\omega)]^N = \frac{1}{(j2\omega\sigma^2 + 1)^N} e^{-\frac{jN\omega A^2}{j2\omega\sigma^2 + 1}}$$

$$= \left(\frac{1}{2\sigma^2}\right)^N e^{-\frac{NA^2}{2\sigma^2}} \frac{1}{\left(j\omega + \frac{1}{2\sigma^2}\right)^N} e^{-\left[\frac{1}{\frac{4\sigma^4}{NA^2}\left(j\omega + \frac{1}{2\sigma^2}\right)}\right]}. \quad (7.32)$$

The last expression matches the formula in Campbell and Foster (no. 650.0, p. 77 [12]). The probability density function can be obtained from the inverse Fourier transform as

$$p_{Ns}(z) = \frac{1}{2\sigma^2} \left(\frac{z}{NA^2}\right) e^{-\frac{NA^2+z}{2\sigma^2}} I_{N-1}\left(\frac{A\sqrt{Nz}}{\sigma^2}\right), \quad (7.33)$$

where I_{N-1} is the modified Bessel function of the first kind. The results in (7.29) and (7.33) are used to find the probability of false alarm and the probability of detection.

7.11 Detection of Multiple Samples Based on Summation [2, 9–20]

In this approach, N samples are used to determine the probability of false alarm. The difference between this method and the L -out-of- N method is that the output from each sample is added together. After N samples of data are collected, the result is compared with one threshold. This case is similar to the pulsed radar detection problem [2]. A brief discussion will be presented here. The probability of false alarm and probability of detection can be determined through the same procedure as discussed for the single data point case.

The first step is to find the probability of false alarm P_{fam} from (7.18). This value will be used to find the threshold. Once the threshold is found, the probability of detection can be determined. The threshold can be found in a similar way as that shown in (7.6), with (7.29) used as the probability density function. The result is

$$P_{fam} = \int_{r_1}^{\infty} p_{Ns}(z) dz. \quad (7.34)$$

In this equation, P_{fam} is given, thus threshold r_1 can be found.

The probability of detection can be obtained in a similar manner as that shown in (7.18). The result is

$$P_{dm} = \int_{r_1}^{\infty} p_{Ns}(z) dz = 1 - \int_0^{r_1} p_{Ns}(z) dz. \quad (7.35)$$

There are no closed-form solutions to these two equations. Numerical integrals will be used to find the results. Several examples are presented in the following sections to demonstrate the procedure to find the false alarm and probability of detection.

In actual application, N can have different values with different thresholds. Detection can be declared as soon as one or more thresholds are crossed. The value of N can even start from one.

7.12 An Example of Single-Sample Detection

The following example is used to demonstrate the application of the equations to find the probability of false alarm and probability of detection of a digital receiver. The design examples in Sections 5.22 and 5.23 will be used here. Let us assume that the input bandwidth is 125 MHz and the desired false alarm time is about 100 s, which means the receiver is allowed to generate on average one false alarm every 100 s. The sampling rate is 250 MHz and the ADC is 8 bits, thus

$$\begin{aligned} T_f &= 100 \text{ s} \\ f_s &= 250 \text{ MHz} \\ b &= 8 \text{ bits.} \end{aligned}$$

In this example, let us use only one sample to detect the existence of the input signal. From (7.2), the probability of false alarm is

$$P_{fas} = \frac{1}{T_f f_s} = 4 \times 10^{-11}. \quad (7.36)$$

From (7.7), the threshold should be set as

$$-\frac{r_1^2}{2\sigma^2} = -23.94$$

or

$$\frac{r_1^2}{\sigma^2} = 47.88. \quad (7.37)$$

It should be noted that the threshold r_1 is an integer number of Q (or $r_1 = nQ$), where n is an integer and Q is the quantized unit. If the noise from (7.8) is substituted into the above equation, the result is

$$\frac{(nQ)^2}{\frac{(1+M)Q^2}{12}} = \frac{12n^2}{1+M} = 47.88, \quad (7.38)$$

where M is the amplifier output noise measured in units of quantization noise (see (5.49)). The M value is approximately equal to 16 (from Section 5.23). The corresponding n value is 8.24. There are three ways to choose the M and n values:

1. Change the M value to make $n = 8$ or 9 to fulfill the result of this equation. This approach will change the gain of the front end slightly.
2. Keep $M = 16$ and choose $n = 8$.
3. Keep $M = 16$ and choose $n = 9$.

Since the mathematical manipulation of all the approaches is exactly the same, keeping $M = 16$ and choosing $n = 8$ and $n = 9$ will be used in the following discussion.

With $M = 16$, these choices of n will reflect

$$\frac{12n^2}{1+M} = \begin{cases} 45.18 & \text{for } n = 8 \\ 57.18 & \text{for } n = 9 \end{cases} \quad (7.39)$$

The corresponding probability of false alarm is

$$P_{fas} = \begin{cases} e^{-22.59} = 1.55 \times 10^{-10} & n = 8 \\ e^{-28.59} = 3.84 \times 10^{-13} & n = 9 \end{cases} \quad (7.40)$$

$$T_f = \frac{1}{P_{fas}f_s} = \begin{cases} 25.8 \text{ sec} & n = 8 \\ 10,416 \text{ sec} & n = 9. \end{cases}$$

One can see that if $n = 8$ is chosen, the false alarm time is worse than the desired value, although it might still be acceptable. If $n = 9$ is chosen, the false alarm time is much better than the desired value. In this discussion, both cases will be carried out. The last step is to find the probability of detection from (7.16) and (7.17). The corresponding r_1 can be found from (7.7).

The Rayleigh and Rician distributions are shown in Figure 7.3. The Rayleigh distribution is labeled noise only and the Rician distribution is at a signal-to-noise (S/N) ratio of 10 dB. An r_1 is arbitrarily chosen. The area on the right side of the noise-only curve represents the probability of false alarm. The area on the right side of the $S/N = 10$ -dB curve represents the probability of detection.

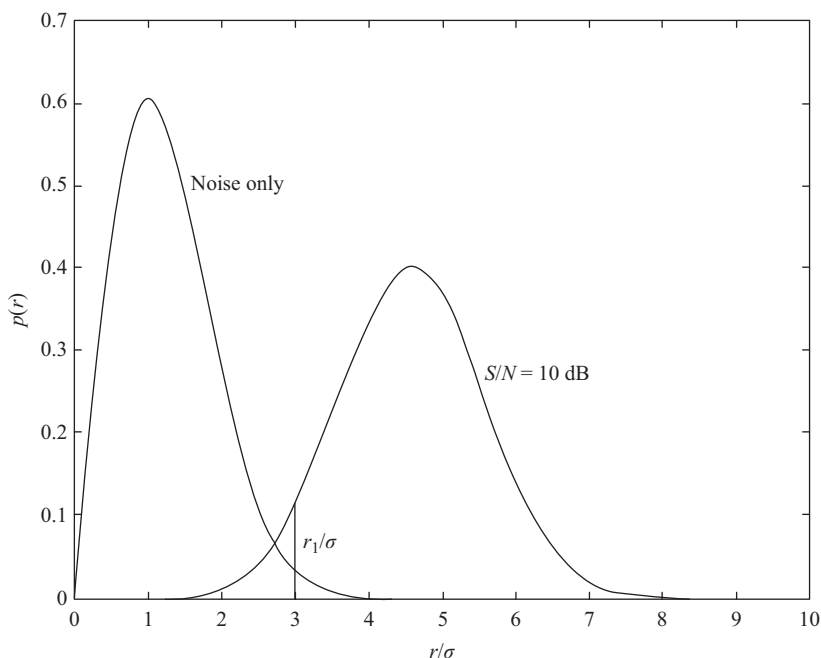


Figure 7.3 Rayleigh and Rician distributions.

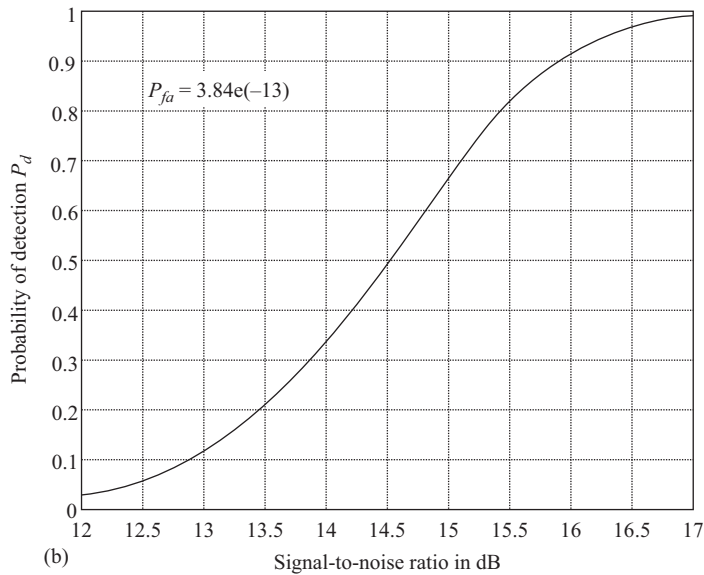
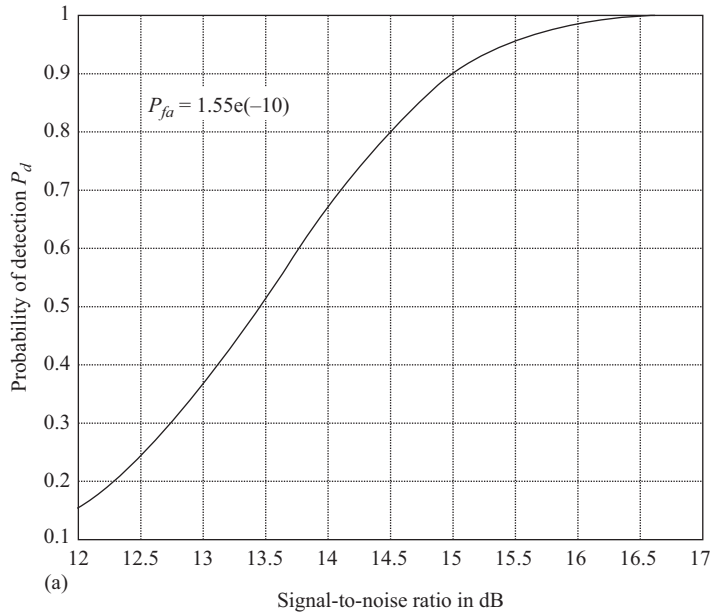


Figure 7.4 Probability of detection versus S/N : (a) $n = 8$ and (b) $n = 9$.

A combination of several programs is used to generate Figure 7.4 and they are listed in the Appendix. Figures 7.4(a) and 7.4(b) show the probability of detection versus S/N for $n = 8$ and 9, respectively. In order to produce a 90% probability of detection, the S/N ratio required is approximately 14.95 dB for $T_f = 25.8$ s and 15.8 dB for $T_f = 10,416$ s. This is the

trade-off between sensitivity and false alarm rate. Accepting a higher false alarm rate implies the receiver can have higher sensitivity.

7.13 An Example of Multiple-Sample (L -out-of- N) Detection

In this example, the requirement is the same as the example in the last section. The only difference is the use of 64 points ($N = 64$) to determine the false alarm rate and probability of detection. The amplifier in front of the ADC is kept the same. In other words, the amplification factor $M = 16$ is used.

Since $N = 64$, from (7.18) the corresponding probability of false alarm is $P_{fam} = 2.56 \times 10^{-9}$. In the next step, the L value is determined. However, the L value is determined by the threshold level. Many quantization levels can be selected as thresholds. In this example, let us try to use the first quantization level as the detection threshold. If a different level is selected as the threshold, the procedure will be exactly the same. The first step is to find the probability of crossing the threshold by noise alone. The result can be obtained from (7.10) with $n = 1$ as

$$P_{fas}(1) = e^{-\frac{6}{1+M}} = e^{-\frac{6}{1+16}} = 0.7026, \quad (7.41)$$

where $P_{fas}(1)$ is the probability of the noise crossing the first quantization level on a one-sample basis. Thus the probability p in (7.19) can be written as $p = P_{fas}(1)$. In order to determine how many samples are needed to cross the threshold in the total of 64 samples, (7.19) and (7.20) are used. In this calculation, because the threshold must be one of the quantization levels, the result obtained will be close to the desired value and the exact false alarm time T_f cannot be achieved.

The calculation is illustrated as follows. Using the result from (7.41) as the p value in (7.20), the results shown in Table 7.1 are obtained from the program `binomial.m` (Appendix 7.I).

For $L = 63$, the probability is 4.351×10^{-9} , which is greater than the desired value of 2.56×10^{-10} . These results can be obtained from a trial-and-error approach by using the program `df9_4.m` in Appendix 7.D. This means the number of false alarms generated is greater than the design goal. If only $L = 64$ is included, the overall probability of false alarm is $P_{fam} = 1.549 \times 10^{-10}$, which is less than the desired value of 2.56×10^{-9} . Therefore if the first quantization level is chosen as the threshold, all 64 samples should cross the threshold for it to be considered that a signal is detected. Thus this example turns out to be a very special case because $L = N$. In general, L will be less than N .

Now let us consider the probability of detection and use a 90% probability of detection as a criterion. A slightly different procedure is used here. We want to find the $p(i|N)$ value in (7.20) that can generate $p(L + |N) = P_{dm} = 0.9$. However, there is no direct solution.

Table 7.1 Results from (7.41) and (7.20).

L	$P(L + N)$
63	4.351×10^{-9}
64	1.549×10^{-10}

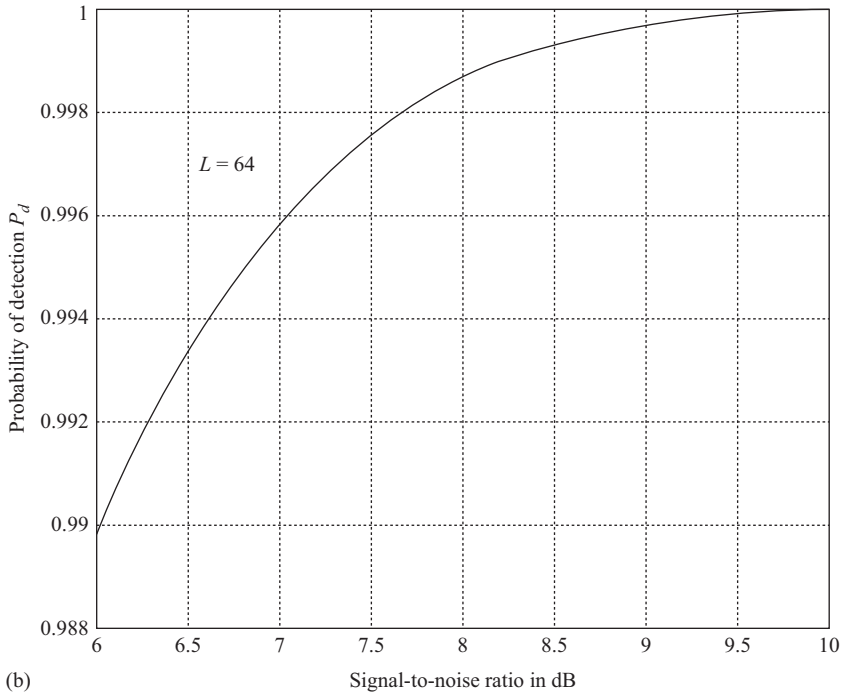
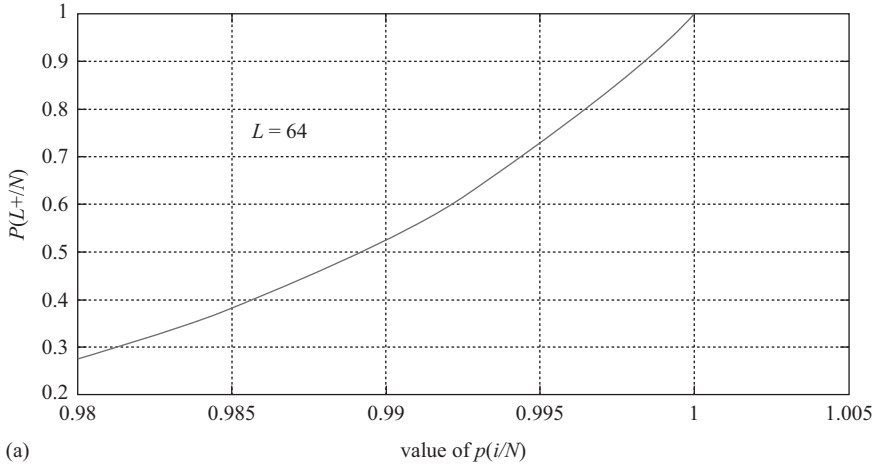


Figure 7.5 Using the first quantization level as the threshold: (a) P_{dm} versus p and (b) P_{dm} versus S/N .

A plot of $p = p(i|N)$ versus $P(L + |N) = P_{dm}$ is shown in Figure 7.5(a). In this special case of $L = N$, this curve is generated through $P_{dm} = p^{64}$. If $L < N$, this curve can be plotted by the program `df9_5.m` in Appendix 7.E. By further refining the value of $p(i|N)$, it is found that $p(i|N) = (0.90)^{1/64} = 0.998355$ will generate the desired P_{dm} .

The last step is to find the S/N ratio of a single sample crossing that will generate a P_{dm} of 0.998355 with the probability of false alarm of 0.7026 given by (7.41). The procedure for finding the probability of detection is exactly the same as that discussed in the previous section. The result is plotted in Figure 7.5(b). The S/N ratio required is about 8 dB, which is much less than the 15 dB required by the single-sample detection scheme. This indicates that the sensitivity of the receiver can be improved. It should be noted that the false alarm time is also improved since the P_{fam} obtained is 1.549×10^{-10} , which corresponds to a false alarm time of 1653 s.

This improvement is obtained by integrating the incoming data points. If the signal is longer than L , this approach will detect the input with a high sensitivity. If the signal length is less than L data points, the chance of detecting the signal is extremely small because the chance of crossing the first threshold by noise alone is small. The selection of N should be based on the minimum PW the receiver is designed to intercept.

7.14 Selection of Threshold Level

In the last section, the first level was arbitrarily chosen as the threshold. Other levels can also be used as the threshold, and the approach is the same as discussed in the previous section. Since the first quantization level used as the threshold turns out to be a special case, the detailed procedure is presented again.

With the second quantization level as the threshold, the probability of crossing the second level by noise alone is

$$P_{fas}(2) = e^{-\frac{6n^2}{1+M}} = e^{-\frac{24}{17}} = .2437, \quad (7.42)$$

where $n = 2$ and $M = 16$. In order to generate a probability of false alarm less than 2.56×10^{-9} but close to it, $L = 39$ is chosen. The value is obtained numerically through trial and error from (7.19) and (7.20) or from the program `binomial.m` (Appendix 7.I). This means 39 samples out of 64 must cross the threshold to be detected as a signal. The corresponding probability of false alarm from (7.20) is

$$p_{fam} = p(L + |N|) = \sum_{i=39}^{64} p(i|64) = 5.6835 \times 10^{-10}. \quad (7.43)$$

Next, the probability of detection is found. In order to generate $P_{dm} = 0.90$, $p = 0.67698$ is required. This result is obtained from Figure 7.6(a), generated from program `df9_5.m` (Appendix 7.E). Finally, the required S/N ratio for one sample to cross the second threshold ($n = 2$) with a probability of false alarm of 0.2437 is obtained. The result is shown in Figure 7.6(b), from program `df9_4.m` (Appendix 7.D). The required S/N ratio value to obtain $p = 0.676$ is about 2.3 dB, which is much less than the 8 dB required before.

In a very similar manner, the third, fourth, fifth, and sixth levels are used for the threshold. The seventh level is too high for the threshold because the false alarm generated is below the desired value. The results are listed in Table 7.2, including the results from single-sample detection.

These results are obtained from program `ta9_1.m` (Appendix 7.A). In this program, the N , L , and P_{fam} values are given and it will find the correct S/N ratio. From these results, it is

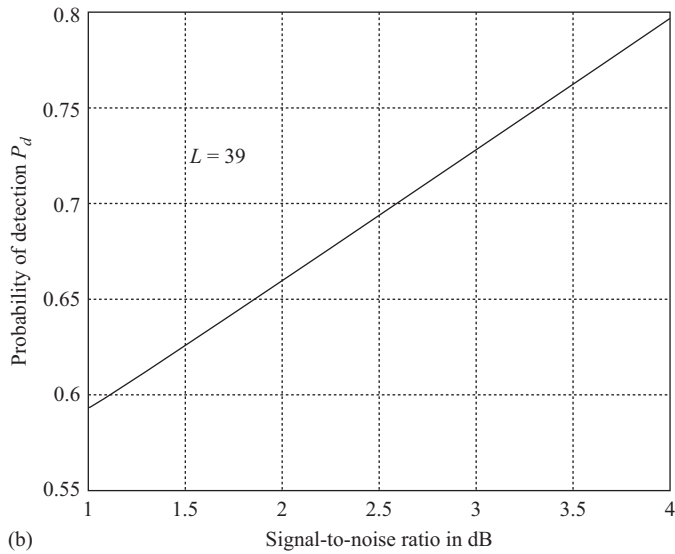
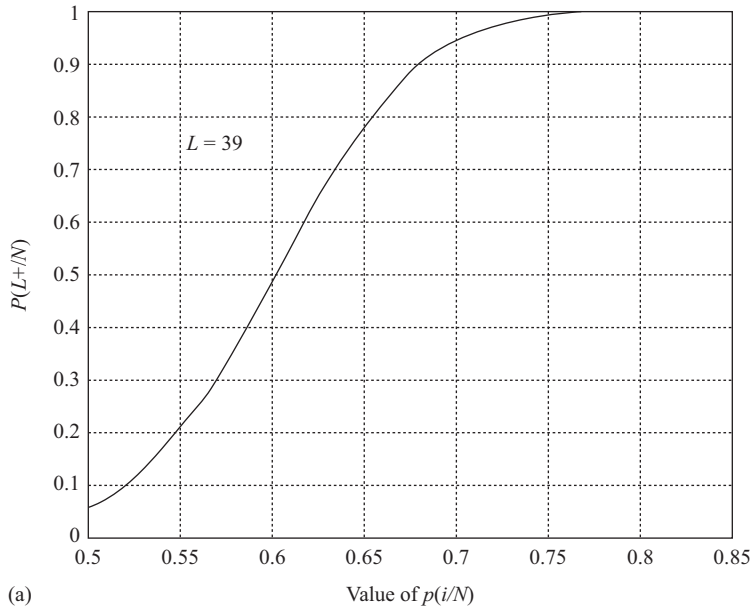


Figure 7.6 Use of the second quantization level as the threshold: (a) P_{dm} versus P and (b) P_{dm} versus S/N .

obvious that multiple-sampled cases provide better sensitivity, which is well known in radar detection. However, among the multiple-sampled cases, it is difficult to make a very accurate comparison since the probability of false alarm in each case is different. However, one can see that the best result appears at $n = 2$. Under this condition, $L = 39$.

Table 7.2 Results from different detection schemes.

N	n	L	P_{fas}	P_{fam}	T (s)	S/N (dB)
1	8			1.55×10^{-10}	25.8	14.95
	9			3.84×10^{-13}	10,417	15.84
64	1	64	0.7026	1.55×10^{-10}	1652	7.78
	2	39	0.2437	5.68×10^{-10}	451	2.22
	3	17	0.0417	7.41×10^{-10}	345	2.25
	4	8	3.52×10^{-3}	8.91×10^{-11}	2873	3.92
	5	4	1.47×10^{-4}	2.96×10^{-10}	865	5.66
	6	3	3.03×10^{-6}	1.16×10^{-12}	220,690	7.74
Summation				2.295×10^{-9}	111.5	0.79

The last row in the table is the result from the summation method. The required S/N ratio is less than the L -out-of- N method. The calculation is presented in Section 7.16.

In general, when a certain quantization level is chosen as the threshold, if the corresponding L is close to $N/2$, the best sensitivity can be expected. This point is further discussed in the next section.

7.15 Optimizing Selection of the Threshold [10]

In this section, it is demonstrated that the optimum threshold selected is close to $L = N/2$ in the L -out-of- N scheme. Since the mathematic operations involved are rather tedious, it is difficult to prove analytically. Thus this point will be illustrated through a numerical approach. Hopefully this illustration will also make the steps in the previous section clearer.

In order to compare the sensitivity at different threshold levels, the probability of false alarm must be kept the same. It has been shown in previous sections that if the threshold levels are selected in a discrete manner, this goal cannot be accomplished. Thus the threshold is selected in a continuous sense. The basic procedure is to find the L value from 1 to 64. At each L value, the required S/N ratio is obtained to generate the same probability of detection with the same probability of false alarm. Let us still use $N = 64$ samples, the same sampling frequency of 250 MHz, the overall probability of false alarm of 2.56×10^{-9} , and the probability of detection of 90%.

The procedure can be divided into three steps for each given L value.

1. From the overall given probability of false alarm (from all 64 samples), find each individual probability of false alarm required to generate the desired value. The approach uses (7.20) and (7.21) by adjusting the p value to generate the desired $P(L + |N) = 2.56 \times 10^{-9}$. A $\pm 0.1\%$ error in $P(L + |N)$ is used in this calculation.
2. From the overall given probability of detection (from all 64 samples), find each individual probability of detection required to generate the desired value. The procedure is exactly the same as in step 1. The only difference is that in the case of $P(L + |N) = 0.900$, the same $\pm 0.1\%$ error in $P(L + |N)$ is used in this calculation.

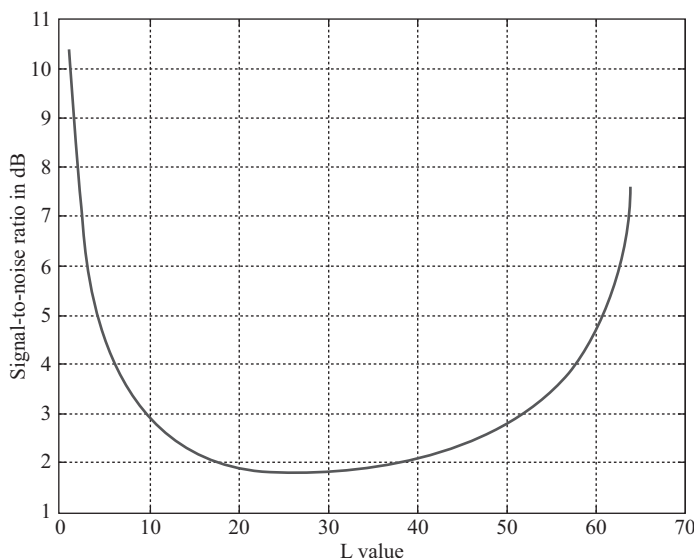


Figure 7.7 S/N ratio versus L value.

3. Find the required S/N ratio for each individual sample. In this approach, (7.17) and (7.18) are used. The S/N ratio is adjusted to match the desired probability of detection. The minimum step size in S/N is 0.01 dB. The program `ta9_1.m` (Appendix 7.A) is used to perform these calculations. The result is shown in Figure 7.7. The minimum S/N ratio required in this calculation is 1.85 dB, which occurs at both $N = 27$ and $N = 28$. The curve at the minimum is rather flat, thus it is not critical to determine the minimum S/N ratio.

From this illustration, one can understand the rule to choose the proper quantization level as the threshold. In the L -out-of- N method, if the threshold can be chosen such that the L value is about half of N , it is a reasonable approach. Similar results are shown in Davenport [9].

7.16 An Example of N Sample Detection (Summation Method)

This example illustrates the summation method discussed in Section 7.11. As discussed earlier, a threshold that generates the desired probability of false alarm must be found first. The threshold can be found from the numerical integration of (7.29). In order to obtain the desired probability of false alarm of 2.56×10^{-9} , the threshold is at 244.5562. However, because the data are quantized, 245 is used as the threshold. Under this condition, the probability is 2.295×10^{-9} . The probability of detection can be found from (7.33) and (7.34). In the integral, the modified Bessel function $I_n(x)$ is approximated by [2]

$$I_n(x) \underset{x \gg n}{\approx} \frac{e^x}{\sqrt{2\pi x}}. \quad (7.44)$$

The program `ta9_la.m` (Appendix 7.B) can be used to calculate the result. The required inputs to the program are the values of the probability of false alarm. The result, listed in the last row of Table 7.2, is that a S/N ratio of 0.79 dB is required.

It should be noted that the summation is made on the outputs of a square law detector and digitization is not taken into consideration.

Comparing the summation and the L -out-of- N methods, two advantages can be found for the summation method: 1) the summation method has higher sensitivity and 2) if the signal is shorter than 64 data points and the signal is strong, the summation method can detect the signal more effectively than the L -out-of- N approach.

Once a sampled datum crosses a threshold in the L -out-of- N approach, the amplitude information is lost and is no longer taken into consideration. In contrast, with the summation approach the amplitudes of the entire signal are accumulated and taken into consideration.

7.17 Introduction to Frequency Domain Detection

In the previous section it is assumed that detection is performed in the time domain. When the signal is detected, the FFT can be used to determine the frequency of the input signal. Now let us assume that an N -points FFT can be performed just fast enough to match the digitizing speed. Under this condition, the input data is processed by the N -points FFT continuously, but without data overlapping. It is desirable to match the N points to the minimum anticipated PW. However, a pulse signal with a minimum PW might be divided into two different FFT windows, and this signal will not be detected with the full sensitivity of the receiver. If the input signal is longer than N points, it is difficult to change the FFT length to match the input signal and improve the sensitivity on long pulses.

After the FFT operation, one must determine 1) whether there is any signal in the windowed data and 2) the number of signals present in the data and their frequencies. An important technical challenge is to avoid sidelobes generated by a strong signal and detect weak signals in the presence of strong ones. Of course, this is also a challenge to other types of EW receivers with simultaneous signal detection capability.

It appears that the problem in frequency domain detection is much more complicated than for time domain detection. One of the major problems is that one does not know when the signal appears in the data points in the time domain. For example, if the total length of the FFT is 64 points, 64 points of time domain data will be processed. If all the data points in this time frame contain signals, the output of the FFT should be very well behaved and detection should be relatively easy.

If only a small portion of the time frame data contain signals, the mainlobe will be very wide (reciprocal of signal length). One extreme case is that the time window collects only one point of data from the leading edge of a strong pulse. The results are shown in Figure 7.8.

Figure 7.8(a) shows noise alone in the time domain and Figure 7.8(b) shows the corresponding power spectrum from the FFT output. Figure 7.8(c) shows there is one data point containing a signal and Figure 7.8(d) shows its corresponding output in the frequency domain. From these figures, it is obvious that when there is even one signal datum in the time domain, the amplitude of the output of the frequency domain increases. If a fixed threshold is used to determine whether there is a signal, the threshold might be crossed at many

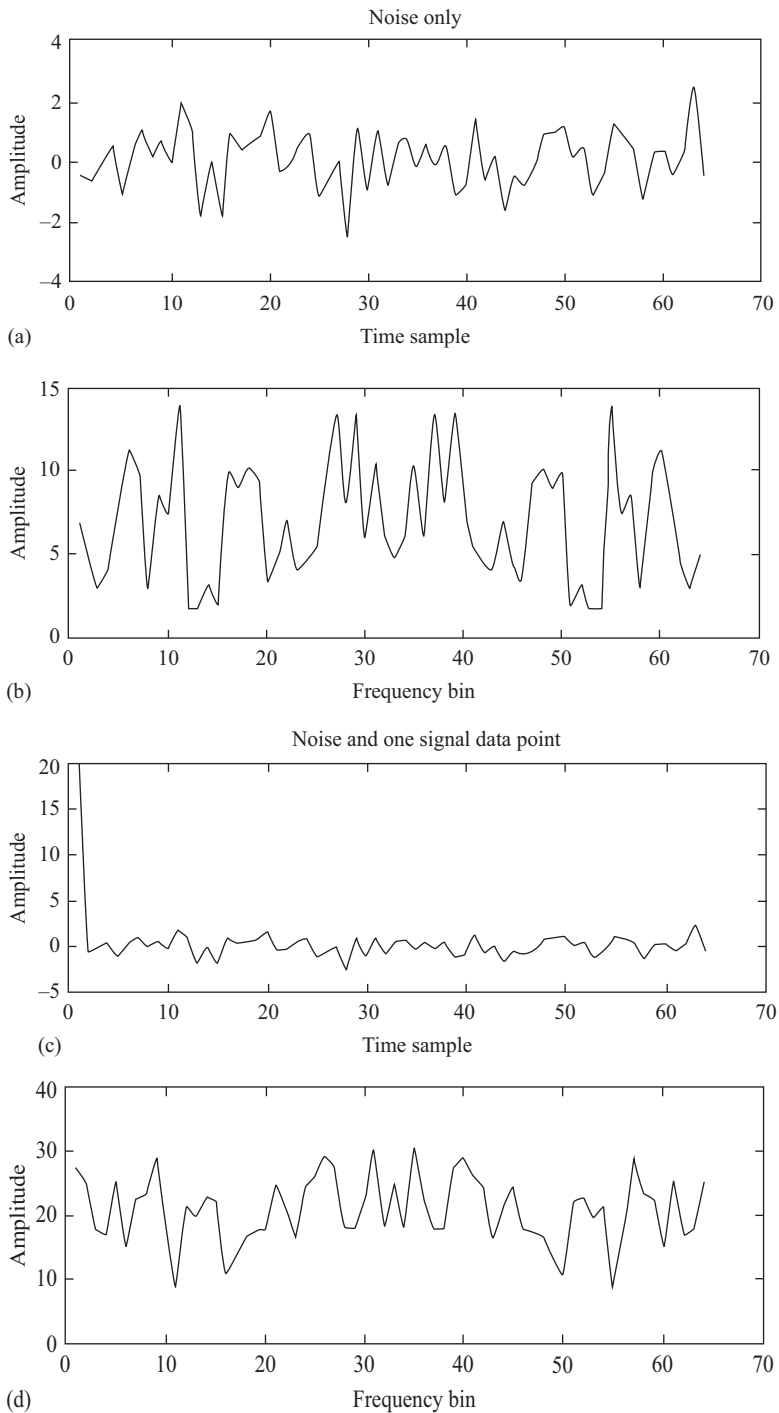


Figure 7.8 Time and frequency domain responses: (a) noise alone in the time domain, (b) noise alone in the frequency domain, (c) noise and one datum containing signal (time), and (d) noise and one datum containing signal (frequency).

frequency components. In addition, the peaks in the frequency domain do not necessarily correspond to the correct frequencies of the input signals because the one sample can create a very wide spectrum.

If most of the data in the time domain contain signals, the peaks in the frequency domain represent the frequency of the input signals. The following sections discuss the detection of signals in the frequency domain.

7.18 A Suggested Approach to Frequency Domain Detection

In this discussion it is assumed that the FFT length is shorter than the shortest anticipated pulse. The FFT is performed consecutively without data overlap. In other words, a signal can be divided into several (at least two) time frames.

One possible solution to the frequency domain detection problem is to use a procedure similar to that used for a microscan receiver [14]. The input signal of a microwave receiver is divided into many segments (scan time) in the time domain. Each time segment contains many serial outputs, representing the input frequency. The receiver is usually designed in such a way that the scan time is equal to or less than the minimum PW to be processed. If one compares a microscan receiver with digital receiver with fixed FFT length operation, the results will be quite similar.

One way to obtain the frequency information in a microscan receiver is to generate frequency information from two consecutive scan times. There are five possibilities for dividing the data into two consecutive time segments.

1. The pulse is equal to or longer than twice the minimum PW. In a digital receiver, this situation corresponds to a $PW > 2N$. Under this condition, the first time frame is likely to be filled with data partially containing signals, but all the data in the second time frame will contain signals. Thus the second time frame should be used to determine the input frequency.
2. In cases 2 through 5, it is assumed that the PW is equal to the window time, or $PW = N$. If the signal fills one entire time frame, the frequency calculated from this time window will be well defined.
3. The signal can be divided equally between two time frames. Under this condition, the power spectrums obtained from these two time windows are the same. Either the first or the second time segment can be used to determine the frequency.
4. The input signal is split into two consecutive time windows and the first time frame contains more signal data.
5. The input signal is split into two consecutive time windows and the second time frame contains more signal data.

In a microscan receiver, the output information is generated on a pulse-by-pulse basis rather than a scan-by-scan basis. Usually the frequency components from two consecutive time frames are compared. If the two time windows contain the same frequency (or are very close in frequency), it is considered that they belong to the same signal. The frame with the stronger signal will be used to determine the signal frequency. In all five cases mentioned above, this approach produces reasonable frequency readings. An identical approach can be used in a digital receiver.

7.19 Probability of False Alarm in the Frequency Domain

The calculation of the probability of false alarm is quite similar to the time domain approach. The probability of false alarm is calculated as shown in Section 7.5. The only difference is that the amplitude of the frequency component is used instead of the sampled amplitude in the time domain. The first step is to find the noise distribution in the frequency domain. Since the FFT is a linear operation, the noise distribution in the frequency domain is similar to the time domain distribution. This can be proven mathematically from the noise power spectrum. The noise is assumed to be uncorrelated in the time domain and the noise power in the time domain can be written as

$$E[x_n x_n] = \sigma^2, \quad (7.45)$$

where $E[\]$ represents the expectation value. The corresponding noise power in frequency domain σ_f can be obtained from the expectation value of the spectrum components X_k and X_l as follows:

$$\begin{aligned} \sigma_f^2 &= E \left[\left(\sum_m^{N-1} x_m e^{-\frac{j2\pi mk}{N}} \right) \left(\sum_n^{N-1} x_n e^{-\frac{j2\pi nl}{N}} \right)^* \right] \\ &= \sum_m^{N-1} \sum_n^{N-1} E[x_m x_n^*] e^{-\frac{j2\pi(mk-nl)}{N}} = \sum_n^{N-1} E[x_n x_n^*] e^{-\frac{j2\pi n(k-l)}{N}} \\ &= \sigma^2 \sum_n^{N-1} e^{-\frac{j2\pi n(k-l)}{N}} = \begin{cases} 0 & \text{for } k \neq l \\ N\sigma^2 & \text{for } k = l \end{cases}, \end{aligned} \quad (7.46)$$

where $*$ presents the complex conjugate. In the above equation, the noise spectrum is represented by its Fourier transform. This equation shows that the noise has the same distribution, but the variance is increased by N . This is reasonable because in order to calculate the noise power spectrum, all the N points in the time domain are used.

This result implies the distribution of the power spectrum is Rayleigh. The probability density function can be written in the same form as (7.5) by replacing σ^2 with $N\sigma^2$. This result can also be obtained from the real and imaginary parts of the FFT. They can be expressed as

$$\begin{aligned} p_f(x) &= \frac{1}{\sqrt{2\pi N\sigma}} e^{-\frac{x^2}{2N\sigma^2}} \\ p_f(y) &= \frac{1}{\sqrt{2\pi N\sigma}} e^{-\frac{y^2}{2N\sigma^2}} \\ p_f(r) &= \int_0^{2\pi} r p_f(x) p_f(y) d\phi = \frac{r}{N\sigma^2} e^{-\frac{r^2}{2N\sigma^2}}, \end{aligned} \quad (7.47)$$

where the subscript f represents the probability density in the frequency domain. The probability of false alarm is

$$P_{faf} = \int_{r_1}^{\infty} p_f(r) dr = e^{-\frac{r_1^2}{2N\sigma^2}}. \quad (7.48)$$

For a given probability of false alarm, (7.48) can be used to set the threshold r_1 . If the false alarm time T_f is given, the corresponding probability of false alarm in the frequency domain P_{faf} is

$$P_{faf} = \frac{N}{T_f f_s}, \quad (7.49)$$

where N is the total number of points in the FFT and f_s is the sampling frequency. Since a decision is made every N samples, this result is identical to (7.18).

7.20 Input Signal Conditions in Frequency Domain Detection

The probability of detection in the frequency domain depends on the input signal conditions. For example, if the input power is constant and the frequency matches one of the frequency bins after the FFT, its power spectrum will be high compared with the input frequency between two frequency bins. Four signal conditions will be discussed here. In the first two conditions, the signals fill up the time window, with the input frequency on a frequency bin and a frequency at the middle of two bins. In the last two conditions, the signal only fills half the time window, with the input frequency on a frequency bin and at the middle of two bins. The input signal is assumed to be a complex sinusoidal wave, which can be written as

$$x(n) = Ae^{\frac{j2\pi nk_o}{N}}, \quad (7.50)$$

where A is the amplitude of the signal, k_o is the frequency, and N is the length of the FFT used to process this signal.

The four signal conditions can be written as follows.

1. Signals fill up the time domain window and the input frequency matches one of the spectrum lines. This condition will produce the highest spectrum output and thus the highest probability of detection. Under this condition, $k_o = k_i$, where k_i is a certain spectrum component. The highest output of the FFT is

$$X(k_i) = \sum_{n=0}^{N-1} Ae^{\frac{j2\pi nk_o}{N}} e^{-\frac{j2\pi nk_i}{N}} = A \sum_{n=0}^{N-1} e^{\frac{j2\pi n(k_o - k_i)}{N}} = NA. \quad (7.51)$$

2. Signals fill up the time domain window and the input frequency is exactly at the center of two frequency lines. Under this condition, $k_o = k_i + 0.5$ and there are two highest outputs with the same amplitude. The spectrum output $X(k_i)$ is

$$X(k_i) = A \sum_{n=0}^{N-1} e^{\frac{j2\pi n(k_o - k_i)}{N}} = A \sum_{n=0}^{N-1} e^{\frac{j\pi n}{N}} = A \frac{1 - e^{j\pi}}{1 - e^{j\pi/N}}. \quad (7.52)$$

Its amplitude is

$$|X(k_i)| = \frac{2A}{\sqrt{2 - 2 \cos \frac{\pi}{N}}}. \quad (7.53)$$

It should be noted that the next frequency component $X(k_{i+1})$ has the same amplitude. However, if noise is present, these two amplitudes may be different. To determine the probability of detection, both frequency components will be considered.

3. Signals fill only half of the time domain window and the input frequency is one of the spectrum lines. Under this condition, $k_o = k_i$ and the highest FFT output is

$$X(k_i) = A \sum_{n=0}^{\frac{N}{2}-1} e^{\frac{j2\pi n(k_o - k_i)}{N}} = \frac{NA}{2}. \quad (7.54)$$

4. Signals fill only half of the time domain window and the input frequency is exactly at the center of two frequency lines. This condition produces the worst detection probability. Under this condition, $k_o = k_i + 0.5$ and the $X(k_i)$ component is

$$X(k_i) = A \sum_{n=0}^{\frac{N}{2}-1} e^{\frac{j2\pi n(k_o - k_i)}{N}} = A \sum_{n=0}^{\frac{N}{2}-1} e^{\frac{j\pi n}{N}} = A \frac{1 - e^{j\pi/2}}{1 - e^{j\pi/N}}. \quad (7.55)$$

Its amplitude is

$$|X(k_i)| = \frac{A}{\sqrt{1 - \cos\left(\frac{\pi}{N}\right)}} \quad (7.56)$$

and $X(k_{i+1})$ has the same amplitude.

Cases 1 and 4 provide the highest and lowest probability of detection, respectively. The general probability of detection should fall between the highest and lowest values. The following sections calculate the probability of detection of the four cases.

7.21 Probability of Detection in the Frequency Domain

The probability of detection is discussed in this section. The approach to solving this problem is similar to the solution for that in the time domain. If there are signals in the output data, the probability density function has the same form as (7.16) by replacing σ^2 by $N\sigma^2$:

$$p(r) = \frac{r}{N\sigma^2} e^{-\frac{r^2 + x_m^2}{2N\sigma^2}} I_0\left(\frac{rX_m}{N\sigma^2}\right). \quad (7.57)$$

This result will be used to find the probability of detection.

The probability of detection for signal conditions 1 and 3, where the input frequency is one of the spectrum lines, can be found as

$$P_{df} = \int_{r_1}^{\infty} p(r) dr = 1 - \int_0^{r_1} p(r) dr, \quad (7.58)$$

where P_{df} represents the probability of detection in the frequency domain and $p(r)$ is the probability density function from (7.57).

For signal conditions 2 and 4, where the input frequency is at the center of two adjacent spectrum lines, the probability of detection should be considered slightly different. If any one of the two adjacent spectrum lines crosses the threshold or both the spectrum lines cross

the threshold, it is considered that there is an input signal. The probability of detection of two adjacent spectrum lines can be found by summing the probability of all three cases: 1) $X(k_i)$ crosses the threshold, 2) $X(k_{i+1})$ crosses the threshold, and 3) both of them cross the threshold.

Here, a slightly different approach is used. First, the chance of the signal not crossing the threshold is evaluated, which can be written as

$$P_{uf} = 1 - P_{df} = 1 - \int_{r_1}^{\infty} p(r) dr = \int_0^{r_1} p(r) dr, \quad (7.59)$$

where P_{uf} is the probability that one of the outputs does not cross the threshold. The probability that both the power spectrum lines do not cross the threshold is P_{uf}^2 . Thus the probability of either one or both the spectrum lines crossing the threshold is

$$P_{df2} = 1 - P_{uf}^2 = 1 - \left[\int_0^{r_1} p(r) dr \right]^2 = 1 - (1 - P_{df})^2 = P_{df}(2 - P_{df})$$

or

$$P_{df} = 1 - \sqrt{1 - P_{df2}}. \quad (7.60)$$

In the above equation, P_{df2} is used to represent the probability of detection from two adjacent spectrum lines to avoid the confusion with P_{df} .

The last step is to relate the frequency outputs to the time domain information. Since the S/N ratio in the frequency domain is used to find the probability of false alarm and the probability of detection, this information should be related back to the time domain S/N ratio. This calculation is also considered in four different signal conditions, as mentioned in the last section.

1. For case 1, the highest spectrum line is $X_m = X(k_i) = NA$ and the S/N ratio in the time and frequency domains can be related by

$$\left(\frac{S}{N}\right)_f = \frac{X_m^2}{2\sigma_f^2} = \frac{(NA)^2}{2N\sigma^2} = N \frac{A^2}{2\sigma^2} = N \left(\frac{S}{N}\right)$$

or

$$\left(\frac{S}{N}\right) = \frac{1}{N} \left(\frac{S}{N}\right)_f. \quad (7.61)$$

In this equation, it should be noted that N is used to represent both the total points of the FFT operation and the noise. When it represents noise, it associates with signal S . The subscript f is used to represent notations in the frequency domain, and no subscript is associated with notations in the time domain. The noise in the frequency domain σ_f is obtained from (7.46).

2. For case 2, $X_m = 2A[2 - 2 \cos(\pi/N)]^{-1/2}$, the S/N ratio in the time and frequency domains is related by

$$\left(\frac{S}{N}\right)_f = \frac{X_m^2}{2\sigma_f^2} = \frac{2}{N[1 - \cos(\frac{\pi}{N})]} \frac{A^2}{2\sigma^2} = \frac{2}{N[1 - \cos(\frac{\pi}{N})]} \left(\frac{S}{N}\right)$$

or

$$\left(\frac{S}{N}\right) = \frac{N[1 - \cos(\frac{\pi}{N})]}{2} \left(\frac{S}{N}\right)_f. \quad (7.62)$$

3. For case 3, $X_m = NA/2$, the relation is similar to case 1, and the result is

$$\left(\frac{S}{N}\right)_f = \frac{N}{2} \left(\frac{S}{N}\right)$$

or

$$\left(\frac{S}{N}\right) = \frac{2}{N} \left(\frac{S}{N}\right)_f. \quad (7.63)$$

4. For case 4, $X_m = A[1 - \cos(\pi/N)]^{-1/2}$, the result is similar to case 2, and that result is

$$\left(\frac{S}{N}\right)_f = \frac{1}{N[1 - \cos(\frac{\pi}{N})]} \left(\frac{S}{N}\right)$$

or

$$\left(\frac{S}{N}\right) = N[1 - \cos(\frac{\pi}{N})] \left(\frac{S}{N}\right)_f. \quad (7.64)$$

These four equations can be used to find the S/N ratio in the time domain once the S/N ratio in the frequency domain is obtained. The following two examples will be used to demonstrate the calculation.

7.22 Examples of Frequency Domain Detection

In this section, the probability of false alarm and detection in the frequency domain will be illustrated by two examples. In the first example, the best detection case is considered when all the data contain signal and the frequency is coincident with one of the output frequency bins. In the second example, the worst case will be considered, which means the data are half-filled with signal and the frequency is between two frequency bins. For simplicity, the quantization effect is neglected in this discussion. The result will be compared with the result of time domain detection. Thus the example in the time domain will be used. The information is given as

- $T_f = 100$ sec
- $f_s = 250$ MHz
- $N = 64$
- $b = 8$ bits.

The probability of false alarm is calculated previously from (7.18) as

$$P_{fa} = 2.56 \times 10^{-9}$$

The noise from the amplifier in front of the ADC is determined as $M = 16$ from the time domain calculation. The corresponding variance of the noise in the frequency domain is $N\sigma^2$.

First, a threshold must be found to provide the desired probability of false alarm. The threshold is set at

$$e^{-\frac{r_1^2}{2\sigma_f^2}} = 2.56 \times 10^{-9}$$

or

$$\frac{r_1}{\sigma_f} = 6.2902. \quad (7.65)$$

This result is obtained from (7.6).

1. In the first example, the input signal will fill up the window and the frequency matches one of the output frequency bins. Under this condition, using (7.58) and program df9_4.m (Appendix 7.D) with $P_{fa} = 2.56 \times 10^{-9}$ one can find that $(S/N)_f = 14.5$ dB is required to generate a 90% probability of detection. The corresponding (S/N) in the time domain can be obtained from (7.61) in decibel form as

$$\begin{aligned} \left(\frac{S}{N}\right) &= -10 \log(N) + \left(\frac{S}{N}\right)_f = -10 \log(64) + 14.5 \\ &= -18.1 + 14.5 = -3.6 \text{ dB}. \end{aligned} \quad (7.66)$$

This result is even better than the summation method in time domain detection, which requires about 0.79 dB. The reason is that in the time domain detection, the signal can be considered summed incoherently, while the FFT integrates the signal coherently.

2. In the second example, the input signal only fills half the window and the frequency is at the center of two frequency bins. In other words, this is the worst signal condition according to the previous discussion.

For $P_{fa} = 2.56 \times 10^{-9}$, the threshold $r_1/\sigma_f = 6.2902$, which is the same as the previous example. For a 90% probability of detection $P_{df2} = 0.9$, the corresponding P_{df} can be found from (7.60) as

$$P_{df} = 1 - \sqrt{1 - P_{df2}} = 0.6838. \quad (7.67)$$

To achieve this probability of detection, a $(S/N)_f = 13.5$ dB is needed. The (S/N) in time domain can be found from (7.64), written in decibel form as

$$\begin{aligned} \left(\frac{S}{N}\right) &= 10 \log(N) + 10 \log\left[1 - \cos\left(\frac{\pi}{N}\right)\right] + \left(\frac{S}{N}\right)_f \\ &= 18.1 - 29.2 + 13.5 = 2.4 \text{ dB}. \end{aligned} \quad (7.68)$$

This result is very close to the S/N required for the L -out-of- N method in time domain detection. In general, the S/N required should be between the most optimum signal conditions and the worst case. In other words, the S/N required is between -3.6 and 2.4 dB.

7.23 Comments on Frequency Domain Detection

The discussion on frequency domain detection is limited to determine the sensitivity of a receiver. It is difficult to generalize this approach to process strong signals, and especially to detect time-coincident (simultaneous) signals. In general, the detection of simultaneous signals through a simple threshold crossing is not satisfactory. The main peaks of the spectrum will be detected, but the sidelobes should be neglected. Special effort and special algorithms are needed to process simultaneous signals. The major effort in designing analog EW receivers with simultaneous signals detection capability is concentrated on picking the true input signals and avoiding the generation of false information from the sidelobes of the signals. It is anticipated that in a digital EW receiver, a similar problem will be encountered.

References

- [1] Chen VC, Ling H. *Time-Frequency Transforms for Radar Imaging and Signal Analysis*. Norwood, MA: Artech House; 2002.
- [2] Marcum J. 'A statistical theory of target detection by pulsed radar'. *IRE Transactions on Information Theory* 1960;**6**(2):59–267.
- [3] Robertson GH. 'Operating characteristics for a linear detector of CW signals in narrow-band Gaussian noise'. *Bell System Technical Journal* 1967;**46**(4):755–774.
- [4] Van Trees HL. *Detection, Estimation, and Modulation Theory, Part I*. New York: John Wiley & Sons; 1968.
- [5] DiFranco JV, Rubin WL. *Radar Detection*. Englewood Cliffs, NJ: Prentice Hall; 1968.
- [6] Whalen AD. *Detection of Signals in Noise*. New York: Academic Press; 1971.
- [7] Papoulis A. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill; 1965.
- [8] Drake AW. *Fundamentals of Applied Probability Theory*. New York: McGraw-Hill; 1967.
- [9] Davenport WB Jr. *Probability and Random Processes: An Introduction for Applied Scientists and Engineers*. New York: McGraw-Hill; 1970.
- [10] Schwartz M, Shaw L. *Signal Processing: Discrete Spectral Analysis, Detection and Estimation*. New York: McGraw-Hill; 1975.
- [11] Scharf LL. *Statistical Signal Processing, Detection, Estimation, and Time Series Analysis*. Reading, MA: Addison-Wesley; 1991.
- [12] Campbell GA, Foster RM. *Fourier Integrals for Practical Applications*. New York: Van Nostrand Reinhold; 1948.
- [13] Shaw A, Xia W. *Wright State University, Dayton, OH. Private communication*.
- [14] Tsui JBY. *Microwave Receivers with Electronic Warfare Applications*. New York: John Wiley & Sons; 1986.
- [15] Press WH, Flannery BP, Teukolsky SA, Vetterling WT. *Numerical Recipes*. Cambridge: Cambridge University Press; 1986.
- [16] Hansen VG. 'Optimization and performance of multilevel quantization in automatic detectors'. *IEEE Transactions on Aerospace and Electronic Systems* 1974;**AES-10**(2):274–280.

- [17] Knight WC, Pridham RG, Kay SM. 'Digital signal processing for sonar'. *Proceedings of the IEEE* 1981;**69**(11):1451–1506.
- [18] Rohling H. 'Radar CFAR thresholding in clutter and multiple target situations'. *IEEE Transactions on Aerospace and Electronic Systems* 1983;**AES-19**(4):608–621.
- [19] Gandhi PP, Kassam SA. 'Analysis of CFAR processors in homogeneous background'. *IEEE Transactions on Aerospace and Electronic Systems* 1988;**24**(4):427–445.
- [20] Polydoros A, Nikias CL. 'Detection of unknown-frequency sinusoids in noise: spectral versus correlation domain'. *IEEE Transactions on Acoustics, Speech and Signal Processing* 1987;**35**(6):897–900.

Appendix 7.A

```
% ta9_1.m for table calculation
%find SNR using Tsui (binomial) method

clear;

r=[8:.1:20]';
global snr var st_dev

ns=input(' How Many Samples N ? ');
%q=input(' Q = ? ');
q=1;
%m=input(' M = ? ');
m=16;
pfa=input(' Probability of False Alarm = ? ');
%pd=input(' Pd = ? ');
pd=0.9;
l=input(' L = ? ');
var=(1+m)*(q.^2)/12;

if ns==1,
    n=sqrt(-1*(1+m)*log(pfa)/6);
else
    pk=inverse(ns,round(l),pfa);
    n=sqrt(-1*(1+m)*log(pk)/6);
end

st_dev=1;
snr=100;
for ii=1:2:5
```

```

for jj=1:10
    snr=snr-(10.^(2-ii));
    if ns==1,
        p_DET=1-quad('pd1',0,sqrt(abs(2*var*log(pfa))));
    else
        det=1-quad('pd1',0,sqrt(abs(2*var*log(pk))));
        p_DET=binomial(ns,round(1),det);
    end
    if (p_DET<=pd),break,end
end
for j=1:10,
    snr=snr+(10.^(1-ii));
    if ns==1,
        p_DET=1-quad('pd1',0,sqrt(abs(2*var*log(pfa))));
    else
        det=1-quad('pd1',0,sqrt(abs(2*var*log(pk))));
        p_DET=binomial(ns,round(1),det);
    end
    if (p_DET>=pd),break,end
end
end
snr

```

Appendix 7.B

%ta9_1a.m the summing method

%Find SNR using Square-Law detection from known ns,pfa and pd

```

clear;
r= [8:.1:20]';
global snr var st_dev ns pfa

ns=input(' How Many Samples N ? ');
pfa=input(' Probability of False Alarm = ? ');
%pd=input(' Pd = ? ');
pd=0.9;
var=1;
d_square=msr1;
st_dev=1;
snr=1;

```

```

for ii=1:2:7
    for jj=1:20,
        snr=snr-(10.^(1-ii));
        if ns==1,
            p_DET=1-quad('pd1',0,sqrt(abs(2*var*log(pfa))));
        else
            p_DET=1-quad('pd3',0,d_square);
        end
        if (p_DET<=pd),break,end
    end
    for jj=1:20,
        snr=snr+(10.^(-ii));
        if ns==1
            p_DET=1-quad('pd1',0,sqrt(abs(2*var*log(pfa))));
        else
            p_DET=1-quad('pd3',0,d_square);
        end
        if (p_DET>=pd),break,end
    end
end
end

snr

```

Appendix 7.C

```

% df9_3.m
clf
r = [0:.01:10];
out1 = rayleigh(r);
plot(r,out1)
hold on
out2 = ricianns(r,10);
plot(r, out2)
plot([3, 3], [0, .11])
xlabel('r')
ylabel('P(r)')
text(1.5, .55, 'noise only')
text(5.5, .35, 'S/N = 10 dB')
text(3.1, .08, 'r1')

```

Appendix 7.D

```
% df9_4.m RIC1_INT This program generates prob of detection given the
% prob of false alarm rate and signal-to-noise ratio. Plot Pd vs. snr
clear
clf

snrends = input('enter starting and ending snr values [a b] = ');
%[12 17]
r = [8:.1: 20]';
global snr

p_fa = input('enter probability of false alarm p_fa = ');
%1.55e-10;3.84e-13
threshold = sqrt(abs(2*log(p_fa)));
l=0;
for snr = snrends(1,1):.1:snrends(1,2);
    l = l+1;
    p_d(l) = 1- quad('rician', 0, threshold);
end
plot([snrends(1,1) : .1 : snrends(1,2)], p_d)
grid
xlabel('Signal-to-noise ratio in dB')
ylabel('Probability of detection Pd')
% text(12.5, .85, 'pfa = 1.55e(-10)')
text(12.5,.85,'Pfa = 3.84e(-13)');
```

Appendix 7.E

```
% df9_5.m BINOPD_P This program calculates the probability of
binomial distribution
% and plot P_dm versus p.
clear

xx = input('enter that starting pt of L (from 1 to 64) = ');
x = [xx:64];

p = input('ent a range of p(i/N) values [.7:.01:.99] = ');
% exp(-12/(1+M));
v = length(p);
b = length(x);
ex = zeros(v,b);
p1 = zeros(v,b);

coff = gamma(65)./(gamma(65-x).*gamma(x+1));
```



```

n = 1;
while n <= v,
    ex(n,1:b) = (p(n) .^x) .* ((1-p(n)) .^(64-x));
    coff(n,1:b) = coff(1,1:b);
    p1(n,1:b) = coff(n,1:b) .* ex(n,1:b);
    n = n+1;
end
if b == 1
    pd = p1';
else
    pd = sum(p1');
end
plot(p, pd)
grid
xlabel('value of p(i/N)')
ylabel('P(L+/N)')

```

Appendix 7.F

% ricianns.m This function calculates the rician pdf given r, snr,
% sigma, and A

```

function y = ricianns(r,snr)
sigma = 1;
A = (10^(snr/20))*sqrt(2);
arg = (r.*A)./(sigma^2);
J = besselj(0, 1i*arg);
I = real(J);
y = (r.*exp(-(r.^2 + A^2) ./ (2*(sigma^2))) .* I) ./(sigma^2);

```

Appendix 7.G

% rayleith.m This function generates the rayleith distribution

```

function y = rayleigh(r);
sigma = 1;
y = r .*exp(-(r.^2) ./ (2*(sigma^2))) ./(sigma^2);

```

Appendix 7.H

% rician.m This function calculates the rician pdf given r, sigma,
% and A

```

function y = rician(r)
global snr
sigma = 1;
A = (10^(snr/20))*sqrt(2);
arg = (r.*A)./(sigma^2);
J = besselj(0, i*arg);
I = real(J);
y = (r.*exp(-(r.^2 + A^2) ./ (2*(sigma^2))) .* I) ./ (sigma^2);

```

Appendix 7.I

```

% binomial.m
function y=binomial(n,l,p)
y=0;
for a=1:n
    pl(a)=prod(linspace(n-a+1,n,a))*((1-p).^(n-a))/
    prod(linspace(1,a,a))*(p.^a);
    y=y+pl(a);
end
end

```

Appendix 7.J

```

function d=inverse(n,l,p)

pfas=1;

%for i=1:2:13,
for i=1:2:9,
    for j=1:10,
        pfas=pfas-(0.1.^(i));
        %pp=pfas/1000000;
        y=binomial(n,l,pfas);
        if (y<=p),break,end
    end
    for j=1:10
        pfas=pfas+(0.1.^(1+i));
        %pp=pfas/1000000;
        y=binomial(n,l,pfas);
        if (y>=p),break,end
    end
end
d=pfas;
end

```

Appendix 7.K

```
%file name: msr1.m
%Find the threshold d for square-law detection

function d=msr1

global ns pfa

if ns==1
    d=-2*1*log(pfa);
else
    x=1000;
    for ii=1:2:5
        for jj=1:10,
            x=x-(10.^(3-ii));
            y=1-gammainc(x,ns);
            if (y>=pfa),break,end
        end
        for jj=1:10
            x=x+(10.^(2-ii));
            y=1-gammainc(x,ns);
            if (y<=pfa),break,end
        end
    end
    d=2*1*x;
end
```

Appendix 7.L

```
%file name: pd1.m

function y=pd1(r)

global snr st_dev var

a=sqrt(2)*st_dev*(10^(snr/20))*sqrt(var);
arg=(r.*a)./var;
I = besselj(0,1i*arg);
j=I(1);
digits=I(2);
b=real(j);
y=(r.*exp(-(r.^2+a^2)./(2*var)).*b)./var;
end
```

Appendix 7.M

```
%file name: pd3.m
%ideal one or multiple samples square_low
%probability density function of detection

function y=pd3(r)

global st_dev snr var ns

a=sqrt(2)*st_dev*(10^(snr/20))*sqrt(var);
arg=a.*sqrt(ns.*r)./var;
I=besselj((ns-1),1i.*arg);
j=I(1);
b=real(j.*((-1i).^(ns-1)));
y=((r./ns./(a.^2)).^((ns-1)./2)).*exp(-(ns.*(a.^2)+r)./var./2)).*b./var./2;
end
```


Phase Measurements and Zero Crossings

8.1 Introduction

In this chapter, a phase measurement method and a zero crossing method are discussed. These methods are relatively simple, but they can operate only under limited conditions. If the input contains only one sinusoidal wave, these methods can provide very accurate frequency measurement. Theoretically these methods can detect the existence of simultaneous signals. The phase measurement approach uses analog in-phase (I) and quadrature (Q) channels as the front end. If the I - Q channels are perfectly balanced, the phase measurement method can measure the frequencies of two input signals. The zero crossing method can measure the frequencies of multiple input signals. However, when multiple signals are present, the frequency accuracy measured by both these two methods will suffer compared with one input signal. Some of these methods need more investigation to determine their performances.

If the input signal is frequency modulated (FM), these methods can find the instantaneous frequency of the signal. In other words, the frequency versus time can be obtained. Such information is difficult to obtain from the Fourier transform. If the signal modulation is phase shift keying (PSK), the phase-sensitive methods can detect phase reversal in the signal, therefore it can detect the chip rate (or the clock frequency) of the PSK. These capabilities are also limited to a single signal. If there are simultaneous signals, these methods may generate erroneous information.

The instantaneous frequency measurement (IFM) receiver, an analog receiver discussed in Chapter 2, measures the phase between an input signal and its delayed version to obtain frequency information. The receiver can process only one signal and is disturbed by simultaneous signals. The phase measurement method is quite similar to this type of receiver. The first element of an IFM receiver is a limiting amplifier, which is a nonlinear device. This nonlinear device makes analysis on multiple signals very difficult, if not impossible. In the phase measurement approach, no limiting amplifier is used in front of the analog-to-digital converter (ADC), therefore it is possible to measure two signals.

Compared with an IFM receiver, the digital phase measurement approach has another advantage, that is, its adaptability. In an analog IFM receiver, the delay time is fixed and is selected based on the minimum anticipated pulse width (PW). As a result, the frequency resolution is fixed. In the digital version, the delay time can be adaptive, that is, units of

sampling time can be changed. In other words, the delay time can be PW dependent. As a result, the frequency resolution can be PW dependent, which is a very desirable property for an EW receiver. Since this method can generate very fine frequency resolution, it may have special EW applications. Some of the possible applications are discussed later in this chapter.

The major disadvantage of the digital phase measurement method is that the bandwidth of the receiver is narrow. An analog IFM receiver can cover an instantaneous bandwidth of 16 GHz (from 2 to 18 GHz). The digital approach is limited by the speed of the ADC and the digital signal processor following it. The positive aspect of the narrow input bandwidth is its low probability of having simultaneous signals. With a single input signal, this method can measure the frequency very accurately.

8.2 Digital Phase Measurement [1–8]

The simplest digital phase measurement is shown in Figure 8.1. This figure is similar to Figure 6.7. The input signal is divided into two paths and downconverted. The two output channels are 90 deg out of phase and are referred to as the *I* and *Q* outputs. In order to keep this discussion simple, let us assume that the two channels are perfectly balanced.

From (6.20), the intermediate frequency (IF) outputs are

$$\begin{aligned} V_{if1} &= A \cos[2\pi(f_i - f_o)t + \phi] \\ V_{if2} &= A \sin[2\pi(f_i - f_o)t + \phi], \end{aligned} \quad (8.1)$$

where A and f_i are the amplitude and the frequency of the input signal, respectively, f_o is the frequency of the local oscillator, and ϕ is the initial phase of the input signal. Let $I(n)$ and $Q(n)$ represent the digitized signal where n is an integer. These can be written as

$$\begin{aligned} I(n) &= A \cos[2\pi(f_i - f_o)nt_s + \phi] \\ Q(n) &= A \sin[2\pi(f_i - f_o)nt_s + \phi], \end{aligned} \quad (8.2)$$

where t_s is the digitization time interval.

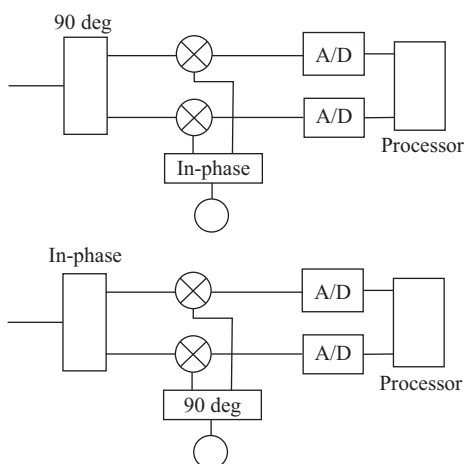


Figure 8.1 Digitizing *I* and *Q* channels.

In order to measure the phase of the signal, a processor is needed to perform the following operations: 1) take the arctangent of $Q(n)/I(n)$ to generate the phase $\theta(n)$, and 2) take the difference of the phase θ at different time intervals. These two steps can be written as

$$\begin{aligned}\theta(n) &= \tan^{-1} \left\{ \frac{Q(n)}{I(n)} \right\} = 2\pi(f_i - f_o)nt_s + \phi \\ \Delta\theta(i) &= \theta(n+i) - \theta(n) = 2\pi(f_i - f_o)it_s,\end{aligned}\tag{8.3}$$

where $\theta(n)$ represents the phase angle at time n and $\Delta\theta(i)$ is the phase difference between two phase angles separated by time i unit of time t_s . If the $\Delta\theta(i)$ can be obtained, the frequency of the input signal can be found as

$$(f_i - f_o) = \frac{\Delta\theta(i)}{2\pi it_s}.\tag{8.4}$$

In this equation, there is one important point: the phase angle is limited from 0 to 2π (or $-\pi$ to π) radians. If the phase change is over 2π (or 360 deg), the phase will be aliased to less than 2π . Under this condition, an ambiguity problem of reporting an erroneous frequency will occur. In order to keep the input signal unambiguous, the angle difference should be kept less than 2π by selecting a small value of i .

Another important issue is the phase discontinuity. The phase difference calculated from (8.3) is obtained from two phase values. If one value is below 2π and the other above, there is phase discontinuity between these two values. The phase discontinuity must be taken into consideration in order to avoid generating an erroneous result. This procedure is often referred to as phase unwrapping.

8.3 Angle Resolution and Quantization Levels

In this section, the relationship between angle data resolution and the number of ADC bits is discussed. Instead of deriving the relation analytically, a few simple examples are shown graphically in Figures 8.2(a) to 8.2(d). In these figures, the amplitude of the signal always matches the full scale of the ADC.

In Figure 8.2(a), the ADCs have only 1 bit. One bit is along the x -axis (voltage output of the I channel) and one bit along the y -axis (voltage output of the Q channel). The circle represents the angle change in an analog sense. When the circle is in a certain square, the center of the square represents the digitized angle. In each quadrant there is only one angle quantization cell. There are a total of four quantization cells and each cell has the same size. In Figure 8.2(b), the ADCs have two bits. Each quadrant has three quantization cells. The cell size is determined by the length of the arc in the square. One can see that the cell sizes are not uniform. There are a total of 12 cells. In Figure 8.2(c), the ADCs have three bits and each quadrant has seven cells. The nonuniformity of the cell size still exists. The total number of cells is 28. In Figure 8.2(d), the ADCs have four bits and each quadrant has 15 nonuniform cells. The total number of cells is 60.

From this illustration, one can conclude that when the signal amplitude matches the full scale of the ADC, the angle quantization cell per quadrant A_q is

$$A_q = 2^b - 1,\tag{8.5}$$

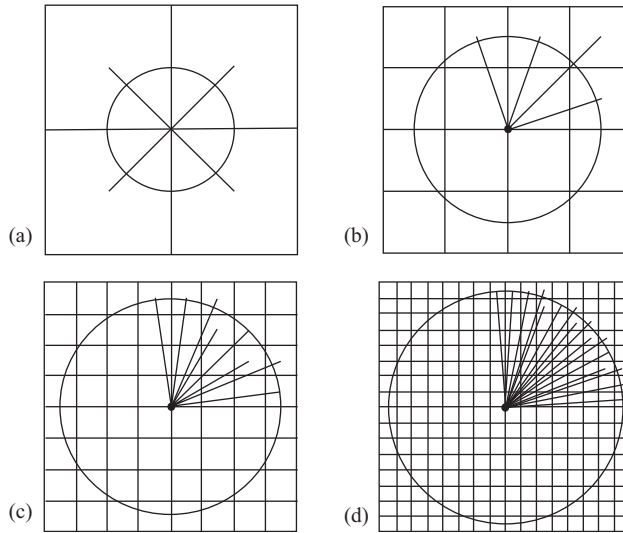


Figure 8.2 Angle resolution versus number of ADC bits: (a) 1 bit, (b) 2 bits, (c) 3 bits, and (d) 4 bits.

where b is the number of bits of the ADC. Since a circle contains four quadrants, the total number of quantization cells A_r is

$$A_r = 4(2^b - 1). \quad (8.6)$$

This quantity represents the maximum number of angle quantization cells. In general, these cell sizes are nonuniform. If the amplitude of the input signal is less than the full scale, the number of angle quantization cells will decrease. For example, if the amplitude of the signal in Figure 8.2(b) reaches only the first bit level, the angle quantization will drop to four cells, as shown in Figure 8.2(a). If the amplitude of the input signal is very strong and drives the ADC into saturation, the number of quantization cells will decrease. In the limit, when the signal is very strong, the sine wave will be digitized more or less like a square wave. The result is similar to the one-bit situation. In order to simplify the discussion, in later discussions the cell size is assumed uniform and the input signal is matched to the maximum of the ADC.

8.4 Comparison of Phase Measurement and Fast Fourier Transform (FFT) Results

In this section, the frequency data resolution generated by phase measurement and by an FFT are compared. Let us use an example in which the ADC has eight bits with a sampling frequency f_s of 250 MHz (or $t_s = 1/f_s = 4 \times 10^{-9}$ s = 4 ns). If a 64-point complex FFT is performed, the total time T is 256 ns (4×64). In this calculation, the period should be used rather than the actual time between samples, as discussed in Section 4.6. The frequency data resolution between adjacent frequency bins is $1/T = 3.906$ MHz, which is independent of the number of bits of the ADC.

If the amplitude of the signal matches the highest level of the ADC, the angle data resolution obtained from (8.6) is 1020 units. If the phase calculation is obtained from the first and the last data points, the actual time difference is $T' = 252$ ns and the corresponding unambiguous bandwidth of the phase measurement method is 3.968 MHz. Another way to look at the unambiguous bandwidth is that whenever the input frequency changes (increase or decrease) by 3.968 MHz, the phase will change by 2π . Thus the phase angle can be measured, theoretically, to one of 1020 cells and each cell has a frequency data resolution of 3.890 kHz (3.968 MHz/1020).

From this example, one can see that from the same input data, if the input is a simple sinusoidal wave, the phase measurement method can provide a frequency data resolution much better than the FFT can provide. In addition, the processing scheme is rather simple in comparison with the FFT processing. However, if there are simultaneous signals, this approach may produce an erroneous frequency, as in an IFM receiver, and this is a major drawback.

8.5 Application of the Phase Measurement Scheme

Equations (8.3) and (8.4) are used to find the frequency of an input signal. In general, using the phase measurement method to obtain fine frequency has two applications: 1) to measure the instantaneous frequency on an FM input signal and 2) to find the highest possible fine-frequency data resolution of a pure sinusoidal wave.

For the first application, the phase difference between every data point is taken. If the data are sampled at a 4-ns (250 MHz) time interval and a total of 64 data points are considered, the phase angle must be calculated at every data point and the phase between each pair of consecutive data points must be obtained. These calculations can be written as

$$\begin{aligned}
 \Delta\theta(1) &= \theta(2) - \theta(1) \\
 \Delta\theta(1) &= \theta(3) - \theta(2) \\
 \Delta\theta(1) &= \theta(4) - \theta(3) \\
 &\dots \\
 \Delta\theta(1) &= \theta(64) - \theta(63),
 \end{aligned} \tag{8.7}$$

where $\Delta\theta(1)$ represents the phase difference of two adjacent phases in time. A total of 63 calculations are required. If the frequency of input signal changes with time, this method can find the frequency change every 4 ns. However, the frequency data resolution is rather poor because of the short delay time. The unambiguous bandwidth is 250 MHz (1/4 ns). If the ADC used has 8 bits, the best frequency data resolution that can be achieved is 245 kHz (250 MHz/1020). If better frequency data resolution is desirable, the phase difference between every other data point can be calculated. In this case, the best frequency data resolution that can be achieved is 122.5 kHz. In general, if a certain delay time is selected, the frequency within the delay period is considered a constant.

If the input signal has a constant frequency, one can use the phase measurement method to obtain very high frequency precision. In this case, it is not necessary to calculate the frequency every 4 ns and a simpler way can be used to obtain the frequency. A binary

approach can save time and calculations. For example, the first, second, third, fifth, and so forth data points can be used to obtain the difference phase angle as

$$\begin{aligned}\Delta\theta(1) &= \theta(2) - \theta(1) \\ \Delta\theta(2) &= \theta(3) - \theta(1) \\ \Delta\theta(4) &= \theta(5) - \theta(1) \\ &\dots \\ \Delta\theta(63) &= \theta(64) - \theta(1).\end{aligned}\tag{8.8}$$

The difference phase angles can be written as $\theta(2^n)$ except for the last data point, where n is an integer and 2^n is the delay time between two data points. The last delay time is $2^n - 1$.

In the above equations, there are 64 (2^6) data points, and the difference phases are calculated only 7 ($6 + 1$) times with delays of 1, 2, 4, 8, 16, 32, and 63. This number of calculations is much less than the number of calculations (63) required in the first approach. In general, if the data have 2^b points, a total of $b + 1$ calculations are needed. In these calculations, the short delay time is used to resolve frequency ambiguity and the long delay is used to provide frequency data resolution. In this example, the shortest delay is 4 ns, which corresponds to an unambiguous bandwidth of 250 MHz. The frequency data resolution generated from the short delay does not need to be very high. It is only required to resolve the ambiguity of the next level (i.e., a minimum of two frequency data resolution cells are needed). For example, $\theta(2)$ has a delay time of 8 ns, which corresponds to an unambiguous bandwidth of 125 MHz. This approach is quite similar to the design of an analog IFM receiver.

8.6 Analysis of Two Simultaneous Signals

As mentioned in previous sections, the major deficiency in the phase measurement approach is that it is susceptible to simultaneous signals. In this section, the theoretical analysis of another phase measurement approach that can measure the frequencies of two simultaneous signals is presented. This approach is limited to two simultaneous signals and is based on McCormick and Lansford [8].

The simultaneous signal detection scheme is rather simple. If there is only one sinusoidal signal, the I and Q channel outputs can be written as in (8.2). The amplitude can be found as

$$\sqrt{I(n)^2 + Q(n)^2} = A,\tag{8.9}$$

which is a constant and independent of time.

If there are two signals of angular frequency f_1 and f_2 , the combined input signal can be written in complex form as

$$s(t) = e^{-j2\pi f_o t} (e^{j2\pi f_1 t} + R e^{j2\pi f_2 t}),\tag{8.10}$$

where f_o is the frequency of the local oscillator and $0 \leq R \leq 1$ represents the amplitude of the second signal. This assumes arbitrarily that the first signal has a unity amplitude and the second signal is weaker than the first one and both signals have zero phase at time $t = 0$. This assumption does not restrict generality. This equation can be written as

$$\begin{aligned}s(t) &= e^{-j2\pi f_o t} e^{j2\pi f_1 t} (1 + R e^{j2\pi \Delta f_2 t}) \\ &= e^{-j2\pi f_o t} e^{j2\pi f_1 t} [p(t) + jq(t)],\end{aligned}\tag{8.11}$$

where $\Delta f = f_2 - f_1$ is the difference frequency and

$$\begin{aligned} p(t) &= 1 + R \cos(2\pi\Delta f t) \\ q(t) &= R \sin(2\pi\Delta f t). \end{aligned} \quad (8.12)$$

This equation can be written in terms of amplitude and phase, and the result is

$$s(t) = e^{-2\pi f_0 t} E(t) e^{j\theta(t)}, \quad (8.13)$$

where $E(t)$ and $\theta(t)$ are the time-dependent envelope and the instantaneous phase of the input signals, which can be written as

$$E(t) = \sqrt{p(t)^2 + q(t)^2} = \sqrt{2 + 2R \cos(2\pi\Delta f t)} \quad (8.14)$$

and

$$\theta(t) = 2\pi f_1 t + \tan^{-1} \frac{q(t)}{p(t)}. \quad (8.15)$$

From (8.14), it is obvious that if there is more than one signal, then the amplitude is time dependent. By measuring the amplitude at different times, one can determine whether there is one or more than one signal.

The instantaneous frequency $f(t)$ can be defined as the derivative of the phase $\theta(t)$ [3], and the result is

$$\begin{aligned} f(t) &= \frac{1}{2\pi} \frac{d[\theta(t)]}{dt} = f_1 + \frac{1}{2\pi} \left(\frac{p\dot{q} - q\dot{p}}{p^2 + q^2} \right) \\ &= f_1 + R\Delta f \left[\frac{R + \cos(2\pi\Delta f t)}{1 + R^2 + 2R \cos(2\pi\Delta f t)} \right] \\ &\equiv f_1 + R\Delta f k_r(t), \end{aligned} \quad (8.16)$$

where $\dot{p}$, $\dot{q}$ represent derivatives with respect to t and

$$k_r(t) = \frac{R + \cos(2\pi\Delta f t)}{1 + R^2 + 2R \cos(2\pi\Delta f t)}. \quad (8.17)$$

These two quantities—the amplitude of the signal and the instantaneous frequency—can be measured to determine the existence of simultaneous signals and to obtain the frequencies of them. Since the frequency of the local oscillator is known, the instantaneous frequency of the signal can be determined.

The amplitude of the envelope, given from (8.14), versus time is shown in Figure 8.3. In this special case, $R = 0.8$.

From (8.17), the maxima of $k_r(t)$ appear at $2\pi f t = 0, 2\pi, 4\pi, 6\pi$, and so forth, where $\cos(2\pi\Delta f t) = 1$ and $k_r(t) = 1/(1 + R)$. The minima of $k_r(t)$ appear at $\pi, 3\pi, 5\pi$, and so forth, where $\cos(2\pi\Delta f t) = -1$ and $k_r(t) = -1/(1 - R)$. The instantaneous frequency depends on the sign of Δf .

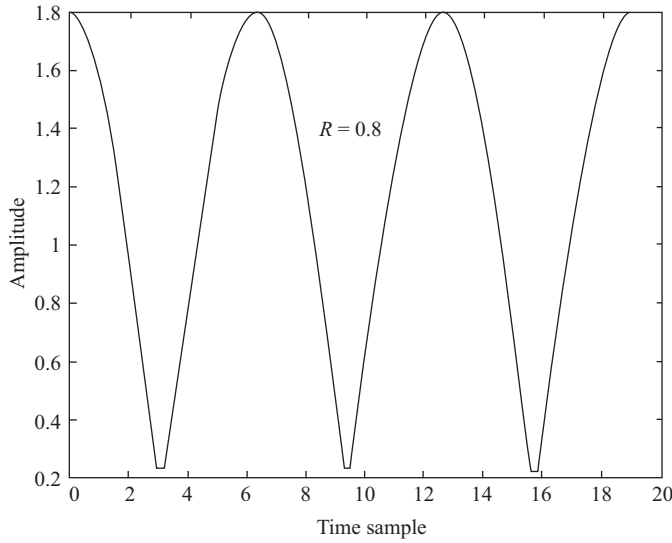


Figure 8.3 Signal envelope versus time.

If Δf is positive (or $f_1 < f_2$), the maximum value of the instantaneous frequency is

$$f_{\max}(t) = f_1 + \frac{R\Delta f}{1 + R} \quad (8.18)$$

and the minimum value is

$$f_{\min}(t) = f_1 - \frac{R\Delta f}{1 - R}. \quad (8.19)$$

The minimum instantaneous frequency can become negative. This condition can be found from (8.19) as

$$f_1(1 - R) < R\Delta f \equiv R(f_2 - f_1)$$

or

$$f_1 < Rf_2. \quad (8.20)$$

Since $0 < R < 1$ as defined in (8.10), in order to obtain a negative frequency, instantaneous frequency f_2 must be higher than f_1 and the condition in the above equation must be fulfilled. Negative instantaneous frequency can be easily understood through the phase angle defined in (8.15). When the phase rotates in the negative direction, the corresponding instantaneous frequency is negative.

If Δf is negative (or $f_1 > f_2$), (8.19) represents the maximum value of the instantaneous frequency and (8.18) represents the minimum value. Under this condition, (8.20) cannot be fulfilled; thus the instantaneous frequency is always positive.

This discussion provides the basic equations to measure the frequencies of two input signals. This discussion is based on continuous functions in the time domain rather than the digitized version. The next section provides an approach to measure these frequencies.

8.7 Frequency Measurement on Two Signals [8]

This section uses the result obtained from the last section to measure the frequencies of two signals. The frequency information is obtained from the instantaneous frequency of (8.16). First, let us take a look at the variable $k_r(t)$ of (8.17). This is a periodic equation with period T such that $\Delta f T = 1$. If the period T is measured, the difference frequency Δf can be determined.

Figure 8.3 shows that the amplitude of the envelope has the same period T as $k_r(t)$. Therefore one can measure either from the amplitude change of the envelope or $k_r(t)$ to obtain the difference frequency Δf . However, $k_r(t)$ is not a term that can be measured explicitly (but the instantaneous frequency $f(t)$ in (8.16) can be measured). The function $f(t)$ is plotted in Figure 8.4 under two different conditions: 1) in Figure 8.4(a), $f_1 = 100$, $f_2 = 150$, and $R = 0.8$, so Δf is positive; 2) in Figure 8.4(b), $f_1 = 100$, $f_2 = 50$, and $R = 0.8$, so Δf is negative. The two frequencies f_1 and f_2 and the amplitude ratio R are arbitrarily chosen since their values bear little significance in this discussion.

In Figure 8.4(a), portions of the instantaneous frequency are negative. The width of the minimum is very sharp in comparison with the maximum values. This shape can indicate Δf is positive ($f_1 < f_2$), while in Figure 8.4(b) the opposite is true and the instantaneous frequency stays in the positive range. This shape can indicate Δf is negative ($f_1 > f_2$). The frequency variation is periodic and this period T can be measured either from the minima in Figure 8.4(a) or from the maxima in Figure 8.4(b). In either case, sharp points in $f(t)$ should be used to determine the period T , as a more accurate result can be obtained. After the value and sign of the difference frequency Δf are determined, both the frequencies of the strong signal and the weak signal can be found.

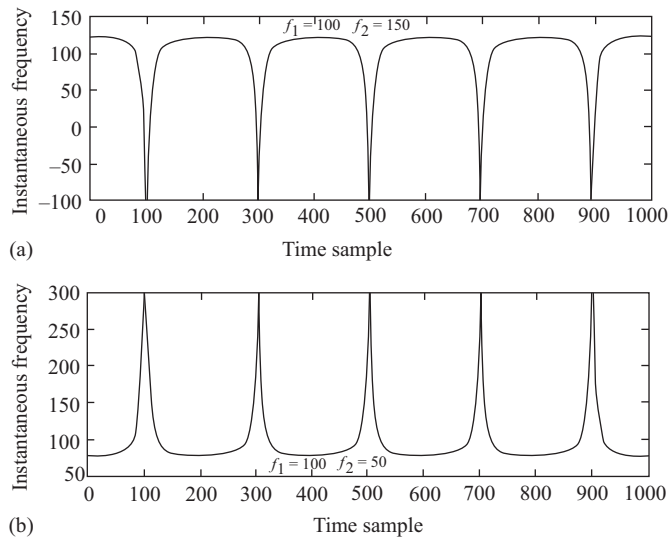


Figure 8.4 Instantaneous frequency versus time: (a) Δf is positive and (b) Δf is negative.

First, let us find the frequency of the strong signal. In order to obtain this frequency, one can take the average of the instantaneous frequency over a period of nT . The result obtained from (8.15) and (8.16) will be

$$\hat{f} = \frac{1}{2\pi T} \int_{t_1}^{t_1+nT} \frac{d[\theta(t)]}{dt} dt = f_1 + \tan^{-1} \left\{ \frac{q(t_1 + nT)}{p(t_1 + nT)} \right\} - \tan^{-1} \left\{ \frac{q(t_1)}{p(t_1)} \right\} = f_1, \quad (8.21)$$

where t_1 is any starting time and n is an integer. Since $p(t)$ and $q(t)$ from (8.12) are periodical of T , the two arctangent terms cancel. This result shows that the average of $f(t)$ is f_1 . In this equation, the period T is not exactly known; however, if one averages many cycles, the frequency error can be made small.

Next, the sign of Δf is determined. The sign can be determined with the help of Figure 8.4. The approach is to measure the time duration of the maximum and minimum of $f(t)$. In Figure 8.4(a), the average of $f(t)$ (or f_1) is close to the maximum value of $f(t)$ and the minima of $f(t)$ occur for a short time, which implies $\Delta f > 0$. If one finds the duration of the maximum is short, then $\Delta f < 0$. Once the sign of Δf is determined, the second frequency can be obtained as

$$f_2 = f_1 \pm \Delta f. \quad (8.22)$$

If the amplitude of the envelope is measured, such as the maximum and minimum, the amplitude of both signals can also be measured from (8.14) as

$$\begin{aligned} E_{\max}^2 &= 2 + 2R \\ E_{\min}^2 &= 2 - 2R. \end{aligned} \quad (8.23)$$

Thus the strong signal A_{st} is 1 and the weak one A_{wk} is R , and they can be found as

$$\begin{aligned} A_{st} &= \frac{E(t)_{\max}^2 + E(t)_{\min}^2}{4} = 1 \\ A_{wk} &= \frac{E(t)_{\max}^2 - E(t)_{\min}^2}{4} = R. \end{aligned} \quad (8.24)$$

This discussion outlines how to obtain two frequencies from the phase measurement scheme. This discussion is based on the assumption that the I and Q channels are perfectly balanced. If the I and Q channels are not perfectly balanced, even a single signal with constant amplitude and frequency can cause the amplitude output to vary as a function of time. Thus, in a real receiver design, the unbalance of the I and Q channels limits the applications of measuring two simultaneous signals.

8.8 Single-Frequency Measurement from Zero Crossing

In the phase measurement method, the input signal must be divided into two parallel outputs: the I and Q channels. If the input is a simple sinusoidal, the frequency can be measured from the period of the signal. The period can be measured from the maxima, the minima, or the zero amplitude points of the digitized data. The zero amplitude points are usually referred to

as the zero crossings because they are usually obtained from two adjacent data points, one positive and one negative.

In this section, the zero crossings are used to find the frequency. This method has the same limitation as the IFM receiver measurement method. Simultaneous signals may cause erroneous results. In the phase measurement system, a minimum of two samples per cycle are required to measure the input frequency. The zero crossing method discussed here uses a real signal, therefore a minimum of four samples per cycle are required.

If the input signal is digitized at sampling frequency $f_s = 1/t_s$, it is not likely that zero amplitude points will be sampled. Thus the first attempt is to find the time of zero crossing points from the digitized data. This can be accomplished from either of the following trigonometric relations [9]:

$$\begin{aligned}\cos[(n+2)2\pi ft_s + \alpha] &= 2 \cos(2\pi ft_s) \cos[(n+1)2\pi ft_s + \alpha] - \cos(n2\pi ft_s + \alpha) \\ \sin[(n+2)2\pi ft_s + \alpha] &= 2 \cos(2\pi ft_s) \sin[(n+1)2\pi ft_s + \alpha] - \sin(n2\pi ft_s + \alpha),\end{aligned}\quad (8.25)$$

where n is an integer, t_s is the sampling period, f is the frequency of the input signal, and α is any initial angle. Now, using the second equation, three consecutive sampled data can be written as

$$\begin{aligned}x(n) &= A \sin(n2\pi ft_s + \alpha) \\ x(n+1) &= A \sin[(n+1)2\pi ft_s + \alpha] \\ x(n+2) &= A \sin[(n+2)2\pi ft_s + \alpha].\end{aligned}\quad (8.26)$$

Substituting these data into (8.25), the result is

$$x(n+2) = 2x(n+1)\cos(2\pi ft_s) - x(n). \quad (8.27)$$

From this equation, the frequency can be obtained as

$$2\pi ft_s = \cos^{-1} \left[\frac{x(n+2) + x(n)}{2x(n+1)} \right]. \quad (8.28)$$

The frequency of the input signal can be found from this equation. This method is further discussed in Chapter 11.

Figure 8.5(a) shows an example of the sampled data. In this figure, three points are chosen and there is a sign change between $x(n+1)$ and $x(n+2)$. The point before the sign change is used as the $t = 0$ reference. In this case, the reference is $x(n+1)$, which will be written as $\sin(\alpha)$. In order to create this condition, it is assumed $n = -1$ in (8.26) and the results are

$$\begin{aligned}x(-1) &\equiv x_1 = A \sin(-2\pi ft_s + \alpha) = A \sin(\alpha) \cos(2\pi ft_s) - A \cos(\alpha) \sin(2\pi ft_s) \\ x(0) &\equiv x_2 = A \sin(\alpha) \\ x(1) &\equiv x_3 = A \sin(2\pi ft_s + \alpha) = A \sin(\alpha) \cos(2\pi ft_s) + A \cos(\alpha) \sin(2\pi ft_s).\end{aligned}\quad (8.29)$$

In this equation, x_1 , x_2 , and x_3 are just new notations used in the following derivations for simplicity of the result. From this equation, it is easy to see that

$$\cos(2\pi ft_s) = \frac{x_1 + x_3}{2x_2}, \quad (8.30)$$

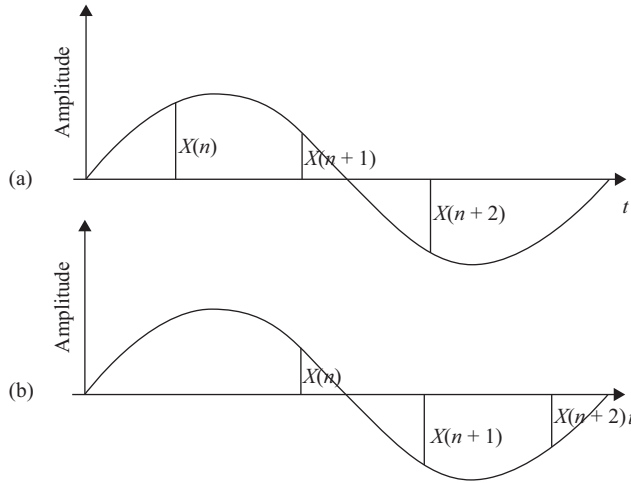


Figure 8.5 Three points selected to calculate zero crossing: (a) between the second and third samples and (b) between the first and second samples.

which is the same result as (8.28). It is also obvious that in order to find the time from x_2 to the zero crossing point, the angle α must be found. Therefore the next step is to obtain $\sin(\alpha)$ from (8.29) and (8.30), which can be written as

$$\begin{aligned} \frac{x_2}{x_3} &= \frac{\sin \alpha}{\sin \alpha \cos(2\pi fT) + \cos \alpha \sin(2\pi fT)} \\ x_3 \sin \alpha - x_2 \sin \alpha \cos(2\pi fT) &= x_2 \sqrt{1 - \sin^2 \alpha} \sin(2\pi fT) \\ \sin^2 \alpha x_3^2 - 2x_2 x_3 \cos(2\pi fT) + x_2^2 \cos^2(2\pi fT) &= x_2^2 (1 - \sin^2 \alpha) \sin^2(2\pi fT) \\ \sin^2 \alpha &= \frac{x_2^2 [1 - \cos^2(2\pi fT)]}{x_3^2 - 2x_2 x_3 \cos(2\pi fT) + x_2^2} \end{aligned} \quad (8.31)$$

From this equation, the time difference between x_2 and the zero crossing can be found as

$$\alpha = \sin^{-1} \left\{ \sqrt{\frac{x_2^2 [1 - \cos^2(2\pi fT)]}{x_3^2 - 2x_2 x_3 \cos(2\pi fT) + x_2^2}} \right\}. \quad (8.32)$$

From the second equation of (8.29), x_2 (or $x(0)$) is $\sin \alpha$ away from zero. The corresponding time δt from x_2 to the zero crossing point can be found as

$$\delta t = \frac{\alpha}{2\pi f} \quad (8.33)$$

since the phase angle changes 2π , covering one cycle with a period of $1/f$, and f is obtained from (8.28). Once this time is determined, the true zero crossing can be obtained by adding δt to $n + 1$, the second data point in (8.26).

8.9 Ill Condition in Zero Crossing for a Single Signal and a Remedy

There is one problem in the above approach, that is, the case where the value of $x(n + 1)$ in (8.26) (or x_2 in (8.30)) is very small. Under this condition, the error in calculating f will be large. To avoid this problem, one can select three different points. For example, if $x(n + 1)$ in Figure 8.5(a) is close to zero, one can choose the next three points, as shown in Figure 8.5(b). Under this condition, the zero crossing is between the first and second samples.

The data point before the zero crossing is still used as $t = 0$. Three values can be written as

$$\begin{aligned} x(0) &\equiv x_1 = A \sin(\alpha) \\ x(1) &\equiv x_2 = A \sin(2\pi f t_s + \alpha) = A \sin(\alpha) \cos(2\pi f t_s) + A \cos(\alpha) \sin(2\pi f t_s) \\ x(2) &\equiv x_3 = A \sin(4\pi f t_s + \alpha) = A \sin(\alpha) \cos(4\pi f t_s) + A \cos(\alpha) \sin(4\pi f t_s). \end{aligned} \quad (8.34)$$

From these relations, it is easy to show that

$$\cos(2\pi f t_s) = \frac{x_1 + x_3}{2x_2} \quad (8.35)$$

and

$$\sin^2 \alpha = \frac{x_1^2 [1 - \cos^2(2\pi f t_s)]}{x_2^2 - 2x_1 x_2 \cos(2\pi f t_s) + x_1^2}. \quad (8.36)$$

The same approach as before can be used to find the time delay from x_1 to the zero crossing points.

It should be noted that in both cases the time calculated is referenced to the data point before the zero crossing. From simulated data it is demonstrated that when the signal-to-noise (S/N) ratio is high, the calculated result is very accurate. If $S/N = 1,000$ dB, the error is less than 10^{-8} Hz. When $S/N = 100$ dB, the error is less than 10 Hz, while for $S/N = 10$ dB, the error is about 100 kHz.

Although the zero crossing time calculated from the above equations is based on the exact solution, the calculation itself is rather tedious. It involves multiplication, square rooting, and an inverse sine calculation. Besides, it may have ill-conditioned cases. Thus a simplified approach is discussed in the next section.

8.10 Simplified Zero Crossing Calculation for a Single Signal

A simpler way to estimate a zero crossing is to connect a straight line between two points on either side of a zero crossing. This is an approximate approach. Figure 8.6 shows such an arrangement. In this approach, two consecutive data points are located, one above zero and one below, or vice versa. A straight line is drawn between the two points and the point where this line crosses the x -axis is considered as the approximate zero crossing point.

The mathematics to calculate this value are as follows. Let these two points be x_1 and x_2 at t_1 and t_2 , respectively. The straight line passing through these two points is

$$\frac{x - x_1}{t - t_1} = \frac{x_1 - x_2}{t_1 - t_2}. \quad (8.37)$$

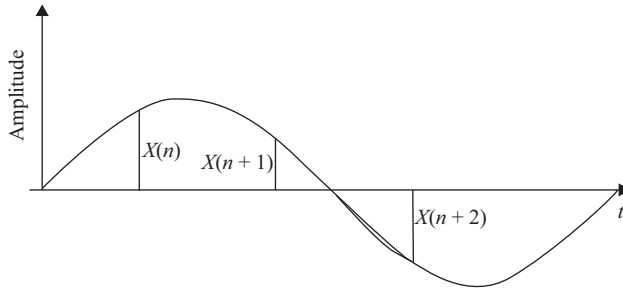


Figure 8.6 A straight line to approximate a zero crossing.

This line intersecting with $x = 0$ provides

$$t = t_1 + \frac{x_1(t_2 - t_1)}{x_1 - x_2} = t_1 + \frac{x_1 t_s}{x_1 - x_2}, \quad (8.38)$$

where the relation of $t_2 - t_1 = t_s$ is used in the above derivation.

Since (8.37) does not generate the true zero crossing, it is desirable to find the error. In this discussion, the maximum error will be derived. The maximum error occurs at the highest input frequency, where four samples are obtained per cycle. Under this condition, $t_s = \pi/2$. Assume that $x_1 = \cos(t)$, then $x_2 = \cos(t + \pi/2)$. The error as a function of t can be found from the above equation by substituting these values for x_1 and x_2 as

$$\begin{aligned} e(t) &= \left\{ t + \frac{t_s \cos(t)}{\cos(t) - \cos(t + \pi/2)} \right\} - t_z \\ &= \left\{ t + \frac{\frac{\pi}{2} \cos(t)}{\cos(t) - \sin(t)} \right\} - t_z, \end{aligned} \quad (8.39)$$

where t_z is the true zero crossing time. To find the worst error, one takes the derivative of $e(t)$ with respect to t . The result is

$$\begin{aligned} \frac{de(t)}{dt} &= 1 + \frac{\pi}{2} \left[\frac{(\sin t + \cos t)(-\sin t) - \cos t(\cos t - \sin t)}{(\sin t + \cos t)^2} \right] \\ &= 1 - \frac{\pi}{2} \left[\frac{(\sin^2 t + \cos^2 t)}{(\sin t + \cos t)^2} \right] \\ &= 1 - \frac{\pi}{2} \left[\frac{1}{(\sin t + \cos t)^2} \right]. \end{aligned} \quad (8.40)$$

Setting this result to zero yields

$$(\sin t + \cos t)^2 = \frac{2}{\pi}, \quad (8.41)$$

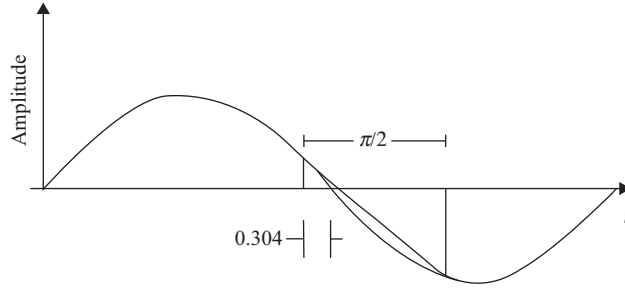


Figure 8.7 Worst error in straight-line approximating of the zero crossing.

which can be solved as

$$\begin{aligned}\sin^2 t + 2 \sin t \cos t + \cos^2 t &= \frac{\pi}{2} \\ 2 \sin t \cos t &= \frac{\pi}{2} - 1 \\ \sin 2t &= \frac{\pi - 2}{2}\end{aligned}$$

or

$$t = \frac{1}{2} \sin^{-1} \left(\frac{\pi - 2}{2} \right) = \begin{cases} 0.304 \\ \pi - 0.304 = 2.838 \end{cases} \quad (8.42)$$

This result is shown in Figure 8.7. The zero crossing produced under this condition can be found through (8.38) as follows:

$$\begin{aligned}t_1 &= 2.838 \\ x_1 &= \sin(t_1) = \sin(2.838) = 0.299 \\ x_2 &= \sin\left(t_1 + \frac{\pi}{2}\right) = \sin(4.409) = -0.954 \\ t_{zs} &= t_1 + \frac{0.299 \times \frac{\pi}{2}}{0.299 + 0.9541} = 3.213 \\ t_z &= \pi \\ e(t_1) &= t_{zs} - t_z = 0.071,\end{aligned} \quad (8.43)$$

where t_1 is the result obtained from (8.42), t_{zs} is the zero crossing calculated from this straight-line approximation, and t_z is the true zero crossing time. Thus the worst error in percentage per sampling period is $e(t_1)/t_z = 4.5\%$.

Besides the simplicity in frequency calculation, this approach does not have ill-conditioned cases. When a data point is close to zero, it does not create an inaccurate result, as discussed in Section 8.9.

8.11 Experimental Results from Single-Frequency Zero Crossing Methods

In this section, the zero crossing is applied to data collected from a sine wave. The data were collected by a Hewlett-Packard (Palo Alto, CA) digital scope. The input signal is 200 MHz. The output of the scope has 8 bits. The sampling time was a nominal 1 ns; however, the actual sampling time was 9.99401×10^{-10} s. The digitized data are shown in Figure 8.8(a). In this demonstration, only 200 samples were used. The frequency was calculated from the n th zero crossing with respect to the first one. The relation can be written as

$$f_n = \frac{n - 1}{2(t_{zn} - t_{z1})}, \quad (8.44)$$

where n represents the n th zero crossing starting from 2, t_{zn} is the n th zero crossing time, t_{z1} is the first zero crossing time, and f_n is the frequency calculated from the n th and first zero crossing time. This approach uses a longer time difference between many zero crossings to find the input frequency, and better resolution can be obtained.

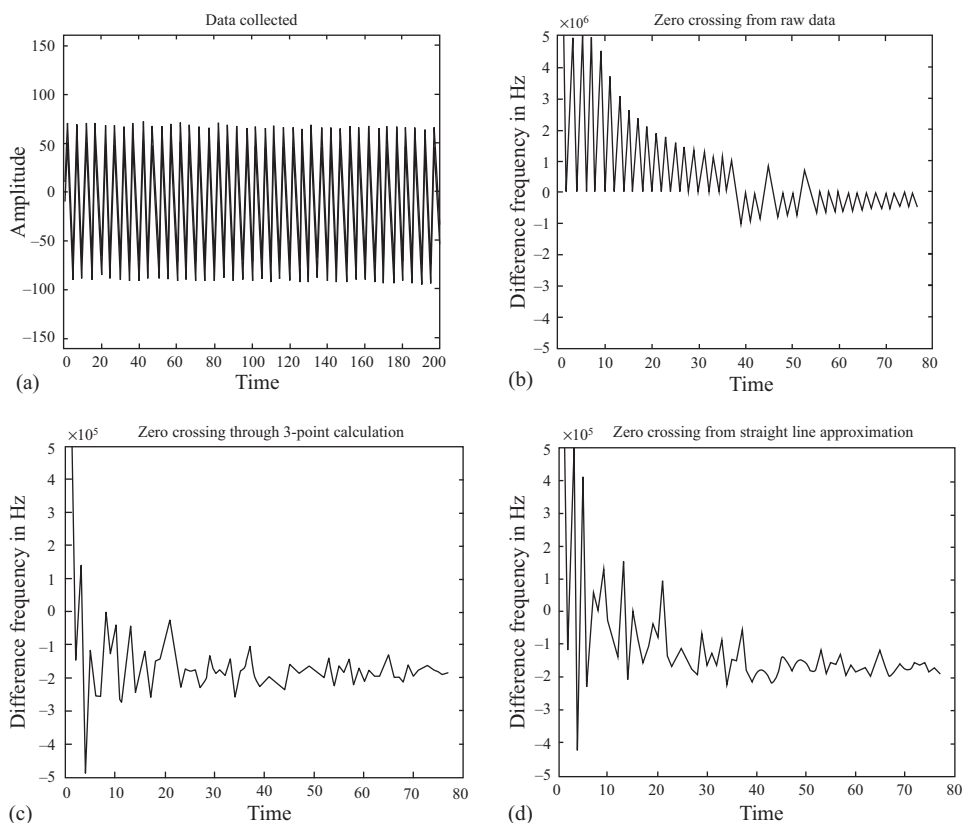


Figure 8.8 Error frequency calculated from the zero crossing: (a) input signal, (b) without fine zero crossing, (c) with three-point fine zero crossing, and (d) with straight-line fine zero crossing.

The results are shown in Figures 8.8(b) through 8.8(d). These figures show the error frequency (measured frequency minus the input frequency) versus time. Let us compare only the variation of the error frequency and neglect the bias error, which may result from experimental setup error. Figure 8.8(b) shows the result when the time resolution is limited to 1 ns. When two data points have opposite signs, the first data point is assumed to be the zero crossing. The variations of the error frequency converge to about ± 200 kHz. Figure 8.8(c) shows the results obtained from the zero crossings calculated from the discussion in Section 8.8. In this calculation, the ill-conditioned cases were detected and corrected as discussed in Section 8.9. If the ill-conditioned cases were not properly corrected, some large errors appeared. However, those results are not presented here. The maximum variation of the error frequency converges to about ± 20 kHz. The result obtained from the straight-line approach presented in Section 8.10 is shown in Figure 8.8(d). The result is similar to Figure 8.8(c).

8.12 Application to Coherent Doppler Radar Frequency Measurement [10]

Both the phase measurement and the zero crossing methods can measure the frequency of one input signal very accurately. The previous discussion is limited to continuous wave (CW) signals. However, these approaches can also be used to measure the frequency of coherent Doppler radars.

A coherent radar is a pulsed radar in which the pulses are obtained from gating on and off a CW signal, as shown in Figure 8.9. In this figure, the CW signal is gated by the rectangular windows and they represent the radar pulses transmitted. There is a certain phase relation between the pulses because they are part of the same CW signal.

A Doppler radar measures the velocity of the target from the Doppler effect. In order to measure the Doppler frequency over a certain range, the spectrum lines must be separated far apart to avoid ambiguity. To accomplish wide-spectrum line separation, the pulses must be close together in time. This effect was discussed in Section 3.5.

The two critical requirements that the phase method and zero crossing approach have are 1) the pulsed signal must be part of the same sine wave (coherent) and 2) the pulses must be close together. Figure 8.10 shows the four pulses with a PW of τ and pulse repetition interval (PRI) of T .

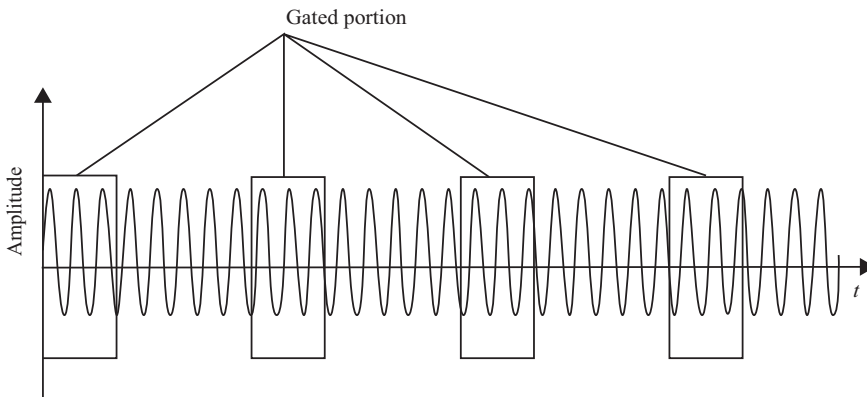


Figure 8.9 Coherent pulse train.

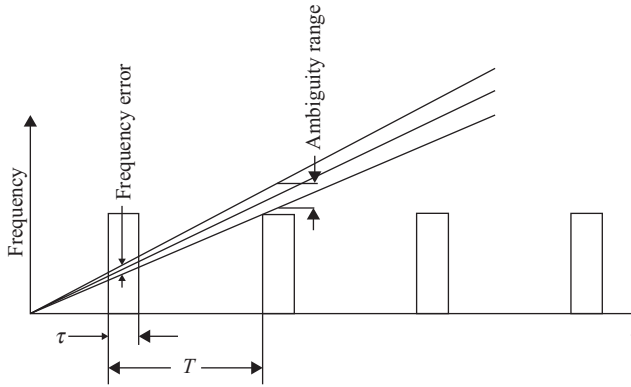


Figure 8.10 Phase error in coherent radar measurement.

If the carrier frequency of the radar is f_i , the corresponding time per cycle is $t_i = 1/f_i$. The frequency accuracy Δf_m measured in time τ can be written as

$$\Delta f_m = \frac{1}{k\tau}, \quad (8.45)$$

where k is a constant. In general, $k > 1$ and it depends on the measurement method and the amount of noise. The percentage error can be written as

$$\frac{\Delta f_m}{f_i} = \frac{1}{k\tau f_i}. \quad (8.46)$$

In order to use this information to continue the measurement on the second pulse, the error extended to the next pulse must be evaluated. The error is amplified by T , as shown in Figure 8.10. However, in order for this extrapolation to work properly, this extended error must be less than one cycle of f_i , thus the relation can be written as

$$\frac{T\Delta f_m}{f_i} = \frac{T}{k\tau f_i} \leq \frac{1}{f_i}$$

or

$$\frac{\tau}{T} \geq \frac{1}{k}. \quad (8.47)$$

The quantity τ/T is usually considered as the duty cycle. This means that the duty cycle must be greater than $1/k$. If the above requirement is fulfilled, the frequency resolution of a Doppler pulse train can be measured as well as in case of a CW signal.

8.13 Zero Crossing Used for General Frequency Determination [11–24]

In the previous sections, zero crossings were used to determine the frequency of the input signal. This approach can provide very accurate frequency information, but only for one input signal. If there is more than one input signal, the previously discussed methods will produce erroneous frequency data. In the following sections, the zero crossing method is

used to generate frequency data that are the same as that obtained from a discrete Fourier transform (DFT). This operation is based on the theory that a band-limited signal can be represented by the real and complex zeros in the function or by a polynomial. Real and complex zeros are discussed in the next section.

The dynamic range of a digital receiver depends on the quantization levels of the ADC used in the system. It was discussed in Chapter 5 that it is difficult to make an ADC with many quantization levels that operate at high speeds. One of the potential advantages of zero crossing spectrum analysis is that an ADC with many quantization levels is no longer needed, provided high clock speed and accurate zero crossing detection can be implemented. However, if an ADC is to be used to measure zero crossing, the number of bits is important because the greater the number of bits, the more accurate the time of zero crossing can be measured. One possible disadvantage of using zero crossing times for multiple signals spectral analysis is that the required signal processing may be relatively complicated. The numerical operations required may actually be comparable to those of the DFT.

The discussion in the next few sections will be kept relatively simple and some of the theoretical analysis is omitted. The discussion is based on Kay and Sudhaker [22] and a thorough theoretical discussion can be found there. This discussion will concentrate on the mechanism used to solve the problem. The presentation is divided into three parts: 1) the basic definition of the problem, 2) the proper generation of zero crossings, and 3) the spectrum estimation.

8.14 Basic Definition of Zero Crossing Spectrum Analysis [11–25]

The input signal $x(t)$ must be bandwidth limited, which means the signal (or all the signals with the narrowband frequency combined) must have a bandwidth $\leq B$, where B is the single-sided bandwidth of the system. The observation time t is from $-T/2 \leq t \leq T/2$. The signal will repeat itself outside the time window T . This assumption comes from the basic idea of digital signal processing, and this phenomenon is explained in Chapter 3. The time bandwidth of the system is BT . The signal $x(t)$ can be represented through the exponential Fourier series as [22]

$$x(t) = \sum_{n=-N}^N C_n e^{\frac{j2\pi nt}{T}}, \quad (8.48)$$

where $N = BT$ and C_n are complex constants, $C_n = C_{-n}^*$, and $*$ represents a complex conjugate. Now let us assume the argument that t has complex values, say $t \rightarrow \xi = t + j\sigma$ and $Z = e^{j2\pi t/T}$. Then, $x(t)$ can also be written in terms of Z as

$$x(Z) = \sum_{n=-N}^N C_n Z^n. \quad (8.49)$$

In this equation, there are $2N$ zeros and the zeros of $x(Z)$ are given by

$$Z_i = e^{\frac{j2\pi \xi_i}{T}} = e^{\frac{j2\pi(t_i + j\sigma_i)}{T}}, \quad (8.50)$$

where $i = 1, 2, \dots, 2N$. The actual axis crossing zeros of $s(t)$ are the real zeros for which $\xi_i = t_i$ (or $Z_i = e^{j2\pi t_i/T}$, $\sigma = 0$). The complex zeros of $x(t)$ are the ones for which $\xi_i = t_i + j\sigma_i$ for $\sigma_i \neq 0$.

Assuming that $x(t)$ has only real zeros, the function $x(t)$ can be written in terms of sine functions as

$$x(t) = 2^{2N} |C_N| \prod_{i=1}^{2N} \sin \left[\frac{\pi}{T} (t - t_i) \right], \quad (8.51)$$

where t_i is the zero crossing time. Whenever $t = t_i$, $x(t) = 0$. It should be noted that the total number of real zeros is $2N$. The function of $\sin[\pi(t - t_i)/T]$ can be written as

$$\sin \left[\frac{\pi(t - t_i)}{T} \right] = \frac{e^{\frac{j\pi(t-t_i)}{T}} - e^{-\frac{j\pi(t-t_i)}{T}}}{2j} = \frac{-j}{2} \left[\sqrt{\frac{Z}{Z_i}} - \sqrt{\frac{Z_i}{Z}} \right] = \frac{-j}{2\sqrt{Z_i}} Z^{-\frac{1}{2}} (Z - Z_i). \quad (8.52)$$

If $|C_N|$ is unknown, this function $x(t)$ may be reconstructed to within a scale factor. Using the relation in this equation, the polynomial $x(Z)$ in (8.51) can be written as

$$\frac{x(Z)}{C_N} = Z^{-N} \prod_{i=1}^{2N} (Z - Z_i) = Z^{-N} \prod_{i=1}^{2N} \left(Z - e^{\frac{j2\pi t_i}{T}} \right), \quad (8.53)$$

where C_N is a constant and $x(Z)$ is given in (8.49). Once $x(Z)$ is written in the form shown in (8.49) with all the coefficients C_n known, the amplitude of each frequency component can be found as

$$X(k) = (2BT + 1) C_k, \quad (8.54)$$

where $k = -N, -N + 1, \dots, 0, \dots, N$. Therefore one can start from (8.52), that is, from the product of $(Z - Z_i)$ to achieve the form of (8.49). The magnitude of C_n represents the amplitude of that frequency component.

In this zero crossing approach, there must be enough real zeros, otherwise the above approach cannot be applied. In the next section, the minimum required zero crossings and one approach to generate them are discussed.

8.15 Generating Real Zero Crossings [22]

In order to obtain information from zero crossings, the number of real zero crossings must equal $2BT$ (or $2N$). If there are not enough real zero crossings, they have to be created to fulfill the required condition.

One way to create all the necessary zeros is by adding a high-frequency signal to the input. The high-frequency signal can be written as

$$x_h(t) = A_h \cos(2\pi f_h t) \quad \frac{-T}{2} \leq t \leq \frac{T}{2}, \quad (8.55)$$

where

$$A_h > \max |x(t)| \quad f_h = B + \frac{1}{T}. \quad (8.56)$$

This frequency f_h is slightly higher than the bandwidth B . The amplitude A_h of the signal must be slightly greater than the sum of all the input signals combined. If the added high-frequency signal is too strong, it will dominate all the zero crossings and the signals to be

measured will have little effect on the zero crossings. As a result, the accuracy of the zero crossing measurement will suffer. In a practical receiver design, a strong high-frequency signal can be used. The relation in (8.56) can be considered as the upper limit of the dynamic range.

The new signal with the high-frequency signal added is

$$y(t) = x(t) + x_h(t). \quad (8.57)$$

After the frequency analysis, this high frequency should be subtracted from the input signal in the frequency domain. With this modification, the total zero crossings are $2(BT + 1)$, which satisfies the requirement.

Figure 8.11 is used to demonstrate the generation of zero crossings. Figure 8.11(a) shows the sum of two sinusoidal waves: the low-frequency one has unit amplitude, while the second

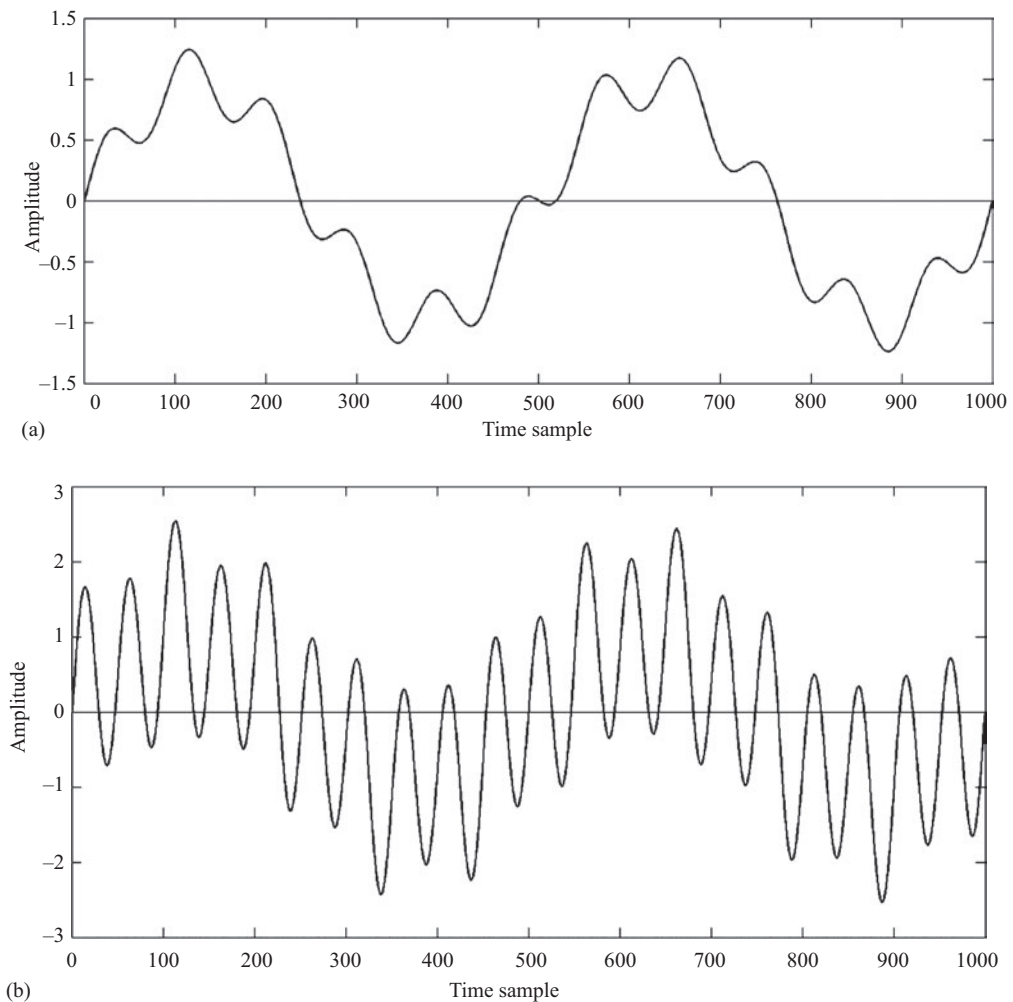


Figure 8.11 Zero crossings of multiple signals: (a) two signals and (b) with added high-frequency signal.

signal has an amplitude of 0.25 with a frequency 5.5 times the first one. Since the first and last points do not cross the zero axis, they are not counted as zero crossings. In this figure, there are five zero crossings. Intuitively one can determine that there are not enough zero crossings since many fine changes do not cross the real axis. Under this condition, one can consider that there are complex zero crossings, which are difficult to realize from this figure. If another signal has an amplitude of 1.3, which is greater than 1.25 (or $1 + 0.25$), with a frequency 10 times the first one, the result is shown in Figure 8.11(b). In this figure one can see that there are many zero crossings and all the detail variations caused by the second signal can be represented by these zero crossings. This processing is referred to as changing complex zeros into real zeros.

If B is 1 GHz and T is 1 μ s, then $2BT = 2000$, which means the system will require 2000 zero crossings. The high-frequency signal required will be $(B + 1/T)$ or 1001 MHz ($1000 + 1$). Therefore, if this signal is added to the input signal, the overall zero crossings should be 2002. Obviously, with this large number of zero crossings, the calculation of the coefficients from (8.53) will not be easy.

8.16 Calculating Coefficients for Zero Crossing Spectrum Analysis [22, 26]

In this section, the approaches to generate the coefficients are discussed. The function of concern is $y(Z)$, which has $2(BT + 1)$ zero crossings rather than $2BT = N$. Thus this equation can be written as

$$y(t) = \prod_{n=-N-1}^{N+1} C_n e^{\frac{j2\pi nt}{T}}. \quad (8.58)$$

In this equation, $C_{N+1} = A_h/2$ since the amplitude and the frequency of the highest frequency are known signals. From (8.53), $y(Z)$ can be written as

$$\frac{y(Z)Z^{N+1}}{C_{N+1}} = \prod_{i=1}^{2(N+1)} (Z - Z_i) = a_0 Z^{2N+2} + a_1 Z^{2N+1} + \cdots + a_{2N+2}$$

and

$$Z_i = e^{\frac{j2\pi t_i}{T}}, \quad (8.59)$$

where t_i is the zero crossing time. In receiver applications, it is important to measure the relative amplitude of all the frequency components, which are equivalent to the amplitudes of the coefficient C_n .

One approach to find the coefficients of (8.59) is referred to as the direct calculation. In this approach, all the terms $(Z - Z_i)$ in (8.59) are multiplied together to obtain the coefficients of Z . When the number of zeros is small this method may be used.

If the number of zeros is large, the direct multiplication method may become cumbersome. Under this condition, a recursive method can be used. The recursive method can be started from direct multiplication as follows:

$$\begin{aligned}
 y^1(Z) &= Z - Z_1 \\
 y^2(Z) &= y^1(Z)(Z - Z_2) = (Z - Z_1)(Z - Z_2) = Z^2 - (Z_1 + Z_2)Z + Z_1Z_2 \\
 y^3(Z) &= y^2(Z)(Z - Z_3) = (Z - Z_1)(Z - Z_2)(Z - Z_3) \\
 &= Z^3 - (Z_1 + Z_2 + Z_3)Z^2 + (Z_1Z_2 + Z_2Z_3 + Z_3Z_1)Z - Z_1Z_2Z_3 \\
 &\dots \\
 y^{k+1}(Z) &= y^k(Z)(Z - Z_{k+1}),
 \end{aligned} \tag{8.60}$$

where $y^k(Z)$ represents the product of k zeros. This relation can be extended to a more general case as

$$\begin{aligned}
 y^k(Z) &= \prod_{i=1}^k (Z - Z_i) = Z^k + \sum_{i=1}^k a_{i,k} Z^{k-1} \\
 &= Z^k + a_{1,k} Z^{k-1} + a_{2,k} Z^{k-2} + \dots + a_{k-1,k} Z + a_{k,k} \\
 y^{k+1}(Z) &= (Z - Z_{k+1})y^k(Z) = Z^{k+1} + (a_{1,k} - Z_{k+1})Z^k + (a_{2,k} - Z_{k+1}a_{1,k})Z^{k-1} + \dots \\
 &\quad + (a_{k,k} - Z_{k+1}a_{k-1,k})Z - a_{k,k}Z_{k+1} \\
 &= Z^{k+1} + \sum_{i=1}^k (a_{i,k} - Z_{k+1}a_{i-1,k})Z^{k+1-i} - a_{k,k}Z_{k+1}.
 \end{aligned} \tag{8.61}$$

In this equation, each coefficient has two subscripts. The first subscript represents the numerical order of the coefficient and the second subscript represents recursion order. This equation reveals that the coefficients a_{k+1} can be obtained from the zero Z_{k+1} and the two coefficients of a_k . The recursion relations are

$$\begin{aligned}
 a_{0,k} &= 1 \\
 a_{i,k+1} &= a_{i,k} - Z_{k+1}a_{i-1,k} \quad i = 1, 2, \dots, k \\
 &= -Z_{k+1}a_{k,k} \quad i = k + 1
 \end{aligned} \tag{8.62}$$

Let us use these relations to obtain the results in (8.62).

$$\begin{aligned}
 a_{0,0} &= 1 \\
 a_{1,1} &= -a_{0,1}Z_1 = -Z_1 \\
 a_{1,2} &= a_{1,1} - Z_2a_{0,1} = -Z_1 - Z_2 \\
 a_{2,2} &= -Z_2a_{1,1} = Z_1Z_2 \\
 a_{1,3} &= a_{1,2} - Z_3a_{0,1} = -Z_1 - Z_2 - Z_3 = -(Z_1 + Z_2 + Z_3) \\
 a_{2,3} &= a_{2,2} - Z_3a_{1,2} = Z_1Z_2 + Z_2Z_3 + Z_3Z_1 \\
 a_{3,3} &= -Z_3a_{2,2} = -Z_1Z_2Z_3.
 \end{aligned} \tag{8.63}$$

The $a_{i,3}$ values are the final result, and they are the same results obtained from $y^3(Z)$ in (8.60).

Obviously, both direct and recursive methods to calculate the coefficients are much more complicated than the calculation in the single-frequency case, but this method can process simultaneous signals.

If these calculated coefficients represent the frequency components of the FFT, only half of them carry information because the frequency components of the FFT have this property. From (8.59), one can see that there are $2N$ ($N = BT$) roots and $2N + 1$ coefficients from a_0 to a_{2N} . Mathematically these coefficients should have the following relations:

$$a_i = a_{2N-i}^*, \quad (8.64)$$

where $i = 0, 1, \dots, N$.

But one can see that these relations do not hold. For example,

$$\begin{aligned} a_0 &= 1 \\ a_{2N} &= \prod_{i=1}^{2N} Z_i. \end{aligned} \quad (8.65)$$

In general, they are not the complex conjugate of each other.

However the following relation does hold [26]:

$$|a_i| = |a_{2N-i}|, \quad (8.66)$$

where $i = 0, 1, \dots, N$. Thus, if the amplitude of the frequency components is of interest, half of the coefficients can provide all the needed information. This result agrees with the FFT result.

The coefficient a_i can be modified by a factor to fulfill the relation in (8.64) [26]. This factor is

$$Z_f = \frac{-1}{2} \sum_{i=1}^{2N} \theta_i, \quad (8.67)$$

where $\theta_i = \tan^{-1} \left(\frac{\text{Im } Z_i}{\text{Re } Z_i} \right)$ and Im and Re represent the imaginary and real parts of a quantity, respectively.

The coefficient a_i can also be modified as

$$b_i = a_i Z_f. \quad (8.68)$$

These b_i values can fulfill the relation

$$b_i = b_{2N-i}^*, \quad (8.69)$$

where $i = 0, 1, \dots, N$ and b_N has real value. Therefore, strictly speaking, the coefficients of (8.59) can be made to have the property shown in (8.64) if they are multiplied by a constant phase term.

8.17 Possible Configuration of a Zero Crossing Spectrum Analyzer

Summarizing the previous discussion on zero crossing spectrum analysis, one can conclude that only two simple steps are needed. First, enough real zeros must be generated. Second, these real zeros are used to calculate the coefficients of the function created by the product of $(Z - Z_i)$.

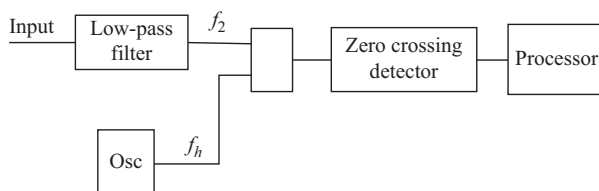


Figure 8.12 Configuration of zero crossing spectrum analysis.

In using the zero crossing for spectrum analysis, a strong signal with a known high frequency must be added to the input signal to create the required real zeros. It might be bothersome for receiver designers that a high frequency must be injected into the input signal. One must perform spectrum analysis on the entire spectrum and at the same time the injected signal with the highest frequency must be neglected. If there is an input signal with a frequency close to the injected one, the results could be confusing.

One way to separate the injected signal is shown in Figure 8.12. In this figure, the input signal bandwidth is limited to f_2 , which can be much less than f_h . Thus the frequencies of the input signal will be far away from the frequency of the injected signal. Since the input signal is limited to f_2 , only the coefficients related to frequencies up to f_2 need to be calculated.

Suppose that the signal bandwidth is B and the bandwidth including f_h is B_h and $B_h > B$. There will be $2B_hT$ zeros. Equation (8.59) can be written as

$$\frac{y(Z)Z^{B_hT}}{C_{B_hT}} = \prod_{i=1}^{2B_hT} (Z - Z_i). \quad (8.70)$$

In this equation, only BT coefficients need to be evaluated. A processor to calculate these coefficients in real time must be built.

These outputs are the same as the frequency components of an FFT operation. Further signal processing is required to determine the number of input signals and their center frequencies.

References

- [1] Earp CW. *Frequency Indicating Cathode Ray Oscilloscope*. U.S. Patent 2434914A, January 27, 1948.
- [2] Wilkens MW, Kincheloe WR Jr. *Microwave Realization of Broadband Phase and Frequency Discriminators*. Technical Report no. 1962/1966-2 SU-SEL-68-057. Stanford, CA: Stanford Electronics Laboratories; November 1968.
- [3] Myers GA, Cumming RC. 'Theoretical response of a polar-display instantaneous-frequency meter'. *IEEE Transactions on Instrumentation and Measurement* 1971;**IM-20**(1):38–48.
- [4] Lang Stephen W, Musicus BR. 'Frequency estimation from phase differences'. *Proceedings of the 1989 International Conference on Acoustics, Speech, and Signal Processing*, vol. 4. New York: IEEE; 1989:2140–2143.
- [5] Kay S. 'A fast and accurate single frequency estimator'. *IEEE Transactions on Acoustics, Speech and Signal Processing* 1989;**37**(12):1987–1990.

- [6] Panter PF. *Modulation Noise and Spectral Analysis Applied to Information Transmission*. New York: McGraw-Hill; 1965.
- [7] Chu D. 'Phase digitizing sharpens timing measurements'. *IEEE Spectrum* 1988;**25**(7):28–32.
- [8] McCormick WS, Lansford JL. 'Time domain algorithm for the estimation of two sinusoidal frequencies'. *IEE Proceedings—Vision, Image and Signal Processing* 1994;**141**(1):33–38.
- [9] Kay SM, Marple SL Jr. 'Spectrum analysis—a modern perspective'. *Proceedings of the IEEE* 1981;**69**(11):1380–1419.
- [10] Paciorek LJ. Anaren Microwave Inc. Private communication.
- [11] Bond FE, Cahn CR. 'On the sampling the zeros of bandwidth limited signals'. *IRE Transactions on Information Theory* 1958;**4**(3):110–113.
- [12] Voelcker HB. 'Toward a unified theory of modulation part I: phase-envelope relationships'. *Proceedings of the IEEE* 1966;**54**(3):340–353.
- [13] Voelcker HB. 'Toward a unified theory of modulation—part II: zero manipulation'. *Proceedings of the IEEE* 1966;**54**(5):735–755.
- [14] Sekey A. 'A computer simulation study of real-zero interpolation'. *IEEE Transactions on Audio and Electroacoustics* 1970;**18**(1):43–54.
- [15] Voelcker HB. 'Zero-crossing properties of angle-modulated signals'. *IEEE Transactions on Communications* 1972;**20**(3):307–315.
- [16] Voelcker HB, Requicha AAG. 'Clipping and signal determinism: two algorithms requiring validation'. *IEEE Transactions on Communications* 1973;**21**(6):738–744.
- [17] Voelcker HB, Requicha AAG. 'Band-limited random-real-zero signals'. *IEEE Transactions on Communications* 1973;**21**(8):933–936.
- [18] Logan BF Jr. 'Information in the zero crossings of bandpass signals'. *Bell System Technical Journal* 1977;**56**(4):487–510.
- [19] Papoulis A. *Signal Analysis*. New York: McGraw-Hill; 1977.
- [20] Requicha AAG. 'The zeros of entire functions: theory and engineering applications'. *Proceedings of the IEEE* 1980;**68**(3):308–328.
- [21] Higgins RC. 'The utilization of zero-crossing statistics for signal detection'. *Journal of the Acoustical Society of America* 1980;**67**(5):1818–1820.
- [22] Kay SM, Sudhaker R. 'A zero crossing-based spectrum analyzer'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1986;**34**(1):96–104.
- [23] Kedem B. 'Spectral analysis and discrimination by zero-crossings'. *Proceedings of the IEEE* 1986;**74**(11):1477–1493.
- [24] Marvasti FA. *A Unified Approach to Zero-Crossings and Nonuniform Sampling of Single and Multidimensional Signals and Systems*. Chicago: Department of Electrical Engineering, Illinois Institute of Technology; 1987.
- [25] Kedem B. *Time Series Analysis by Higher Order Crossings*. New York: IEEE; 1994.
- [26] Marden M. *Geometry of Polynomials*, 2nd ed. Providence, RI: American Mathematical Society; 1985.

Monobit Receiver

9.1 Introduction

In this chapter, the concept of the monobit receiver is introduced. This technique can be considered as a digital channelized approach. The fast Fourier transform (FFT) is very simple and can be built on one chip. A simple frequency encoder is used after the FFT outputs to determine the number of input signals and their frequencies. The design of a candidate encoder will be presented. The encoder and the FFT can be built on one chip. The chip has been fabricated and the monobit receiver concept has been validated successfully in the laboratory.

The monobit receiver concept is inspired by commercial Global Positioning System (GPS) receiver designs. Usually the analog-to-digital converter (ADC) in a commercial GPS receiver has only 1 or 2 bits and the GPS signal is more complicated than a pulsed radio frequency (RF). This idea is adopted in wideband receiver applications and the name monobit receiver is used. This technique can be used to reduce to a minimum the hardware required for a given receiver function, with only a slight reduction in performance.

Because the ADC used in the monobit receiver has very few bits, the system is basically nonlinear. A nonlinear system is difficult to analyze theoretically, thus in designing the receiver, one uses data collected from a data collection system. The collected data are processed in a computer to evaluate the performance. In order to determine the number of input signals, thresholds must be incorporated in the chip design. These threshold values are based on computer simulation results.

The major advantage of the monobit receiver is its simplicity. The price for simplicity is reduced performance in certain configurations. The monobit concept should not be considered as a direct replacement of or compared with the digital channelized approach discussed in the next chapter. Monobit receivers may be used for special applications or to enhance or supplement other receivers. The current demonstration chip only implements the FFT and the frequency encoder. Because the receiver is nonlinear, the RF front-end design can be very simple. In the future, the RF chain and ADC may be included on the same chip. In other words, the entire receiver could be fabricated on one chip. For electronic warfare (EW) applications, size is a very important factor, especially for an airborne system. The ability to easily reconfigure and minimize costs of integration is very important to fielded system applications.

9.2 Original Concept of the Monobit Receiver

The original idea of the monobit receiver is to reduce the FFT complexity by eliminating the multiplication in the FFT operation. One simple way to eliminate multiplication is to use a 1-bit ADC. One bit generates ± 1 as output, and this is the input to the FFT operation.

The discrete Fourier transform (DFT) can be written as

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}, \quad (9.1)$$

where $x(n)$ is the input data and $e^{-j2\pi kn/N}$ is the kernel function. If the input $x(n)$ is ± 1 , no multiplication is needed with the kernel function. Because the FFT only requires addition and subtraction, the chip design can be very simple.

The next step is to manipulate the kernel function. In a computer, a large number of bits are used to represent the value of the kernel function. Because the operation will be built into hardware, the number of bits is important and should be minimized. Because the input has only 1 bit, a low number of bits in the kernel function does not significantly degrade the output in the frequency domain. An optimum combination of signal bits and kernel function bits can be determined experimentally by observing the FFT outputs in the frequency domain.

9.3 Monobit Receiver Idea [1]

Another way to avoid multiplication in the FFT operation is to reduce the number of bits of the kernel function to 1 bit. This idea can be illustrated by using (9.1). The kernel function, however, is a complex function and therefore it cannot be represented by a 1-bit real number. The minimum to represent the kernel function by 1 bit is with 1 bit in the real and 1 bit in the imaginary domain. Mathematically this idea can be written as

$$e^{-j2\pi kn/N} \Rightarrow +1, -1, +j, -j. \quad (9.2)$$

The kernel function can be equal to one of these four values. Under this condition, no multiplication is required in performing the FFT.

Graphically the values of the kernel function are equally spaced around the unit circle in the complex plane as shown in Figure 9.1. The value starts from the real value of 1 and is spaced by angle $2\pi/N$, where N is the total number of FFT points. In this figure $N = 8$. In Figure 9.2 the values of the kernel function are digitized by 1 bit on the real axis and 1 bit on the imaginary axis. One can consider that all the kernel function values in the range of $7\pi/4 \leq \theta < \pi/4$ are quantized to 1, $\pi/4 \leq \theta < 3\pi/4$ are quantized to j , $3\pi/4 \leq \theta < 5\pi/4$ are quantized to -1 , and $5\pi/4 \leq \theta < 7\pi/4$ are quantized to $-j$ as illustrated in Figure 9.2.

Once the kernel function is digitized into 1 bit, the input signal can be multiple bits and there is still no multiplication needed. Using simulation, the number of input data bits is increased to find the effect on the output. There is improvement from 1 to 2 bits, but from 2 to 3 bits the improvement is very small. Increasing beyond 3 bits shows no noticeable improvement. In order to keep the chip design simple, a 2-bit ADC is used.

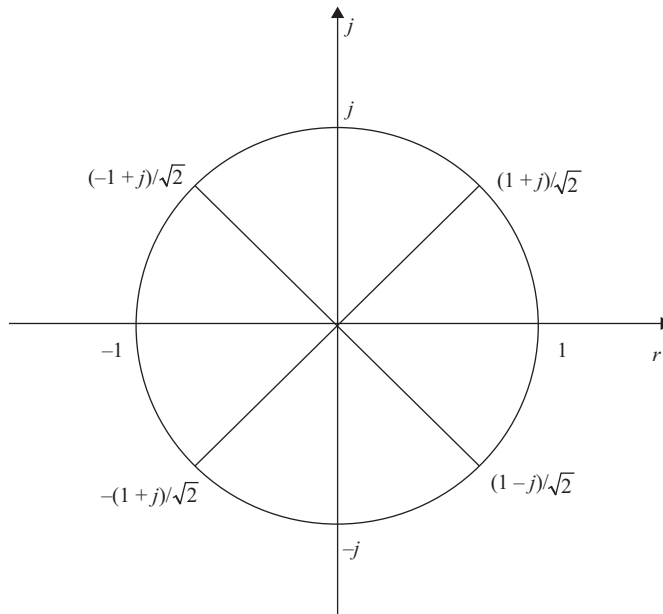


Figure 9.1 Kernel function of DFT with $N = 8$.

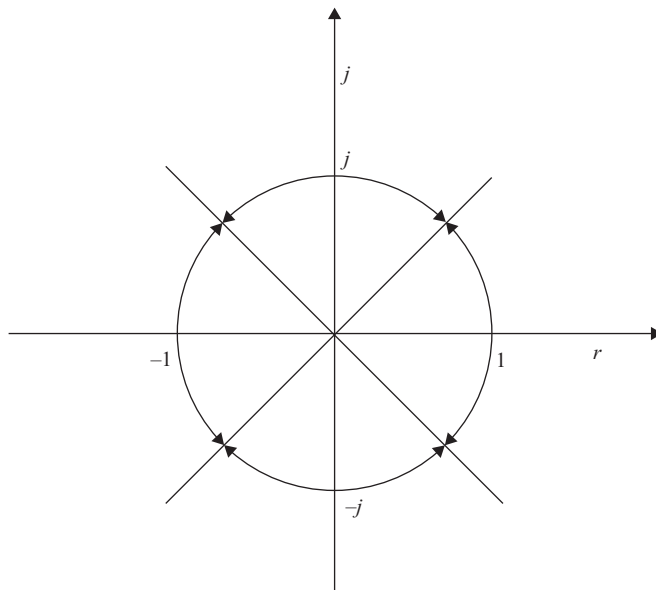


Figure 9.2 Kernel function is digitized 1 bit on the real and imaginary axes.

9.4 Design Criteria

Once the basic approach is determined, some design criteria can be chosen. In order to achieve a 1-GHz input bandwidth, the Nyquist sampling frequency required is 2 GHz. To take the finite slope of the input filter into consideration, 2.5 times the input bandwidth is often used. Thus the ADC should operate at 2.5 GHz with only 2 bits. The corresponding sampling time is 0.4 ns. As stated in Section 5.14, the ADC should be able to take an input of 2.5 GHz. This kind of ADC is readily available since it only requires 2 bits.

If 256 points of data are used in the FFT operation, the equivalent time is 102.4 ns (256×0.4), which is approximately equal to the desired minimum pulse of 100 ns. Thus a 256-point FFT is selected for the design. A rectangular window is used for the FFT operation, which means the input data are not attenuated. In order to simplify the chip design, there is no data overlapping between adjacent FFTs. This arrangement limits the time resolution to 102.4 ns. If a faster FFT chip can be designed, the time resolution will be improved. The bandwidth of each channel is approximately 9.77 MHz ($1/102.4$ ns). Thus the receiver has good sensitivity and can separate signals close in frequency, such as 10 MHz. With this design, the output sampling time is 102.4 ns, which corresponds to an output sampling rate of 9.77 MHz.

The rectangular window has high sidelobes and limits the receiver dynamic to less than 10 dB. Because the input data to the FFT have only 2 bits, the ADC is equivalent to a hard limiter. The amplitude information will be lost anyway. A hard limiter, such as a 1-bit digitizer, exhibits the capture effect under simultaneous signal conditions. The capture effect means that the strong signal suppresses the weak one. This effect also generates many harmonics in the frequency domain.

Figure 9.3 shows the capture effect of a 1-bit digitizer. Figure 9.3(a) shows the spectrum of two sinusoidal waves separated by 3 dB in amplitude. Figure 9.3(b) shows the result after it is digitized by the 1-bit ADC. The amplitudes of the two signals are separated by about 7 dB and many harmonics are generated. From this figure one can see that the receiver is not expected to receive two signals separated very far in amplitude. In other words, the instantaneous dynamic range of the receiver is lower. This is the major deficiency of the monobit receiver.

9.5 Receiver Components

The monobit receiver can be divided into five major parts: the RF chain, the ADC, the demultiplexer, the FFT, and the frequency encoder. Figure 9.4 shows these five components. The ADC operates at 2.5 GHz with only 2 bits. The demultiplexer is 1 to 16, which connects the input to 16 parallel outputs. Each data point contains 2 bits and each bit needs a demultiplexer. In order to simplify the discussion, only the data points rather than the bit number will be used in later discussions.

The FFT operation uses 256 data points and it takes 16 parallel input data every 6.4 ns (16×0.4). It takes a total of 16 cycles or 102.4 ns (16×6.4) to collect all the data. The FFT operation is performed every 102.4 ns. The encoder determines the number of input signals and their frequencies, thus it is referred to as a frequency encoder. A general encoder usually provides frequency, pulse amplitude, pulse width (PW), and time of arrival (TOA) information, as discussed in Section 2.6. To generate a time mark from the encoder is relatively easy, however, it is not included in the present chip design. If a time mark is included in the

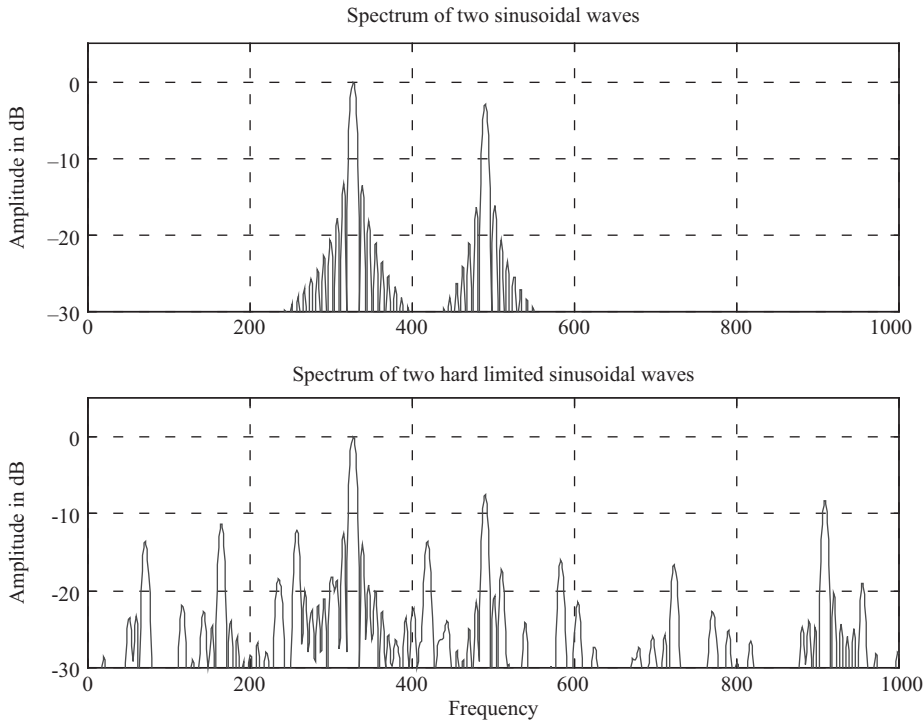


Figure 9.3 Effect of a hard limiter (1-bit ADC): (a) input signal and (b) digitized signal.

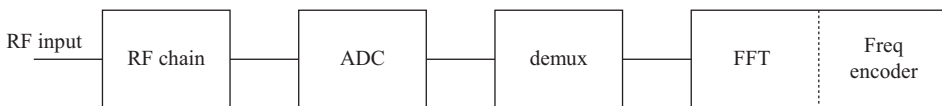


Figure 9.4 The five major portions of the receiver.

design, the resolution will be 102.4 ns, which will be used to generate the PW and TOA. The amplitude information is lost through the 2-bit ADC. Thus this receiver cannot provide pulse amplitude information. This information, if desired, must be obtained from some other circuit, such as a log video amplifier placed in parallel with the monobit circuit.

In the present design only the FFT operation and the frequency encoder are integrated on one chip. In this chapter the discussion will be concentrated on this chip design. Since the ADC is very simple, it appears that the ADC can be fabricated on the same chip in later designs.

9.6 RF Chain, ADC, and Demultiplexer

The RF chain design in a wideband receiver is very important. As discussed in Chapter 5, in order to obtain the desired sensitivity and dynamic range, the gain must be equal to a certain

value and the third-order intercept point must be above a certain value. In the monobit receiver, the instantaneous dynamic range is rather low, at about 5 dB, which means that when two input signals are separated by more than 5 dB, the receiver will only process the strong signal.

With this low dynamic range, the RF chain can be very simple. Instead of using linear amplifiers, a limiting amplifier can be used. The input versus output of a limiting amplifier is shown in Figure 9.5. The performance of a limiting amplifier is similar to an amplifier operating at saturation. The difference between a conventional amplifier operating at saturation and a limiting amplifier is that the output from a limiting amplifier is constant and the output from a conventional amplifier may vary. In many designs the limiting amplifier and conventional amplifier can be interchangeable. It is a common practice to use limiting amplifiers in front of an instantaneous frequency measurement (IFM) receiver.

The RF chain in front of the ADC is shown in Figure 9.6. The center frequency of the two filters is at the center of the second alias zone, as shown in Figure 9.7. The first filter is used to reject out-of-band interference. The second filter is used to limit the noise generated by the amplifier. Without the second filter, the noise from 0 to 2.5 GHz will appear at the input of the ADC. Figure 9.8 shows a picture of the RF chain, in which two limiting amplifiers are used. Each amplifier has a gain of about 30 dB, thus the overall gain is about 60 dB.

The FFT chip cannot accept the input signal at 2.5 GHz. If the data are decimated by 16 and fill a buffer 16-bits wide, the equivalent sampling rate is 156.25 MHz ($2500/16$) and the chip can accept this transfer rate. The ADC has two bits (outputs) and each output is connected to a demultiplexer, as shown in Figure 9.9. The chip will accept 16 bits in parallel at 156.25 MHz. Figure 9.10 shows a picture of the actual receiver on a board along with the

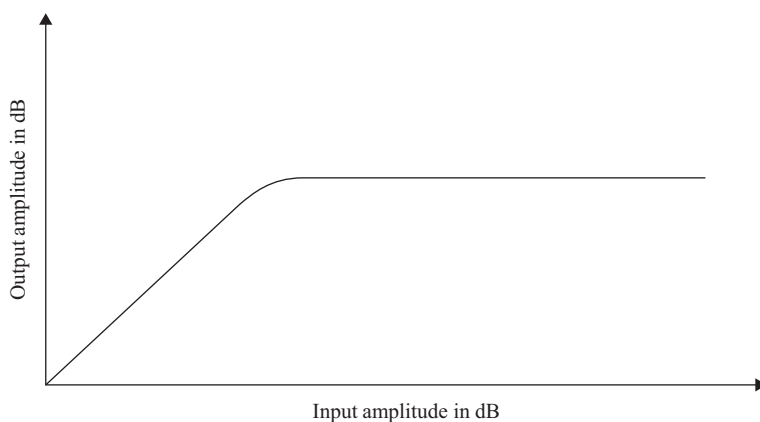


Figure 9.5 Input versus output of a limiting amplifier.

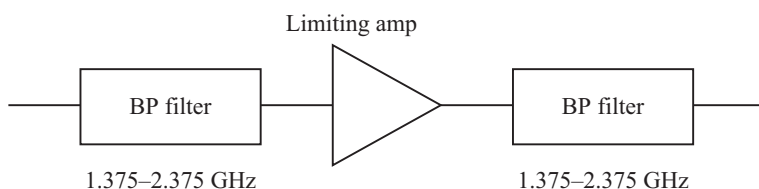


Figure 9.6 RF chain arrangement.

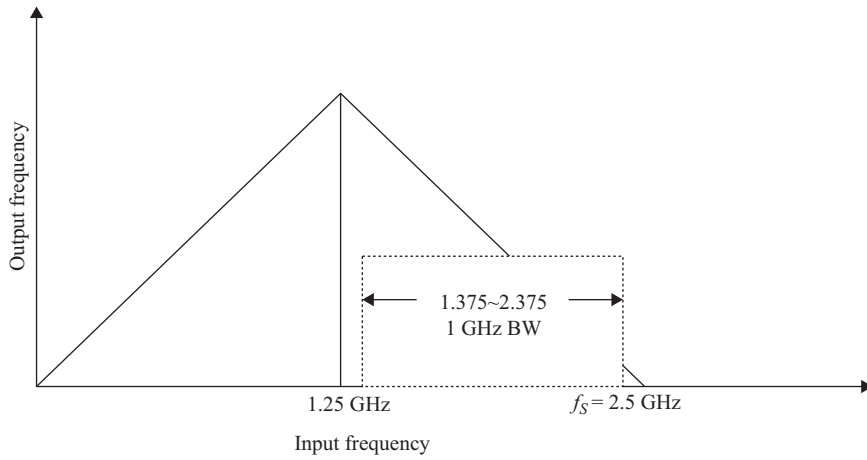


Figure 9.7 Frequency plan of the RF chain.

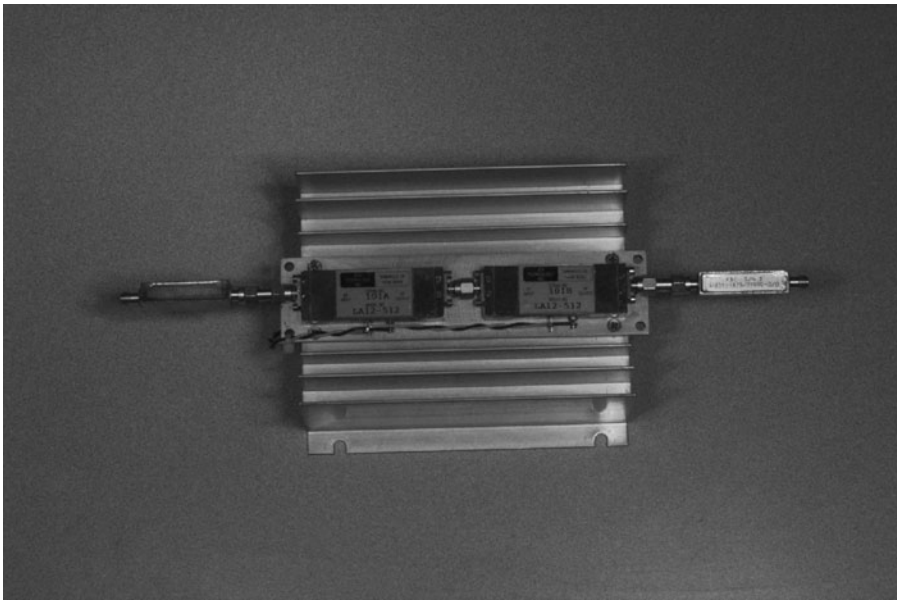


Figure 9.8 The RF chain.

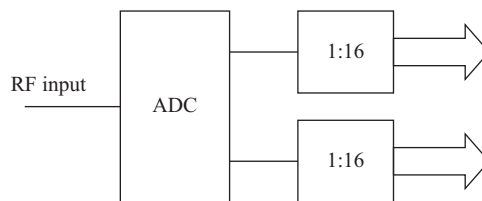


Figure 9.9 ADC and demultiplexers.

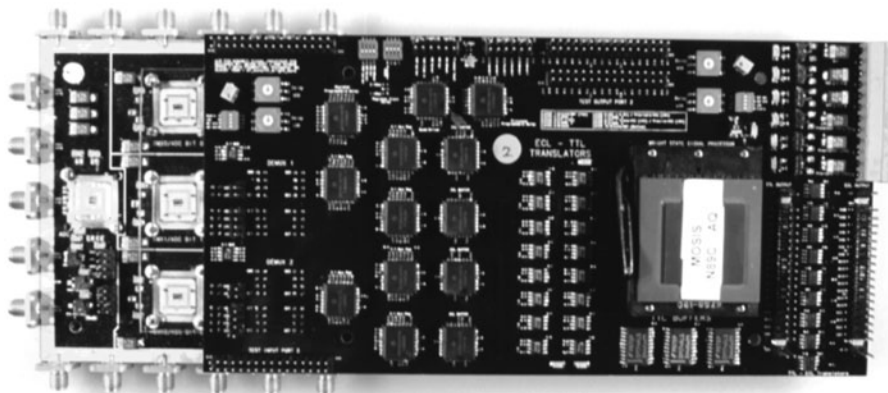


Figure 9.10 An experimental monobit receiver.

ADC, the demultiplexers, and the FFT/frequency encoder chip. This board was built using existing components that are specifically designed for the monobit receiver. On the left of the board is the ADC, which has 3 bits. Following the ADC are three demultiplexers. Only two of the three demultiplexers are used. The large chip on the right lower corner of the board is the FFT/frequency encoder chip. The components in the middle of the board are the level translators, which change the voltage level of the emitter coupled logic (ECL) to the voltage level of the complementary metal oxide semiconductor (CMOS) logic. The ADC and the demultiplexers are ECL and the FFT/frequency encoder chip is CMOS. Because the voltage level used in these two technologies is different, level translators are used to match the voltage levels. In the future the same logic types will be used to build the ADC, the demultiplexers, and FFT chip, thus the level translators will no longer be needed. Many RF inputs are shown on the edge of the board and they are used to supply clocks to the different components, such as the demultiplexers. Because all the clocks used on the chip are phase locked, only one clock input is necessary. This experimental model is rather complicated. With advances in technology, future versions of the board will be simplified tremendously.

The required gain of the RF chain can be determined experimentally. In the experimental setup, an 8-bit ADC is used to collect data and the data are converted into 2 bits through a software program. There are several possible ways to convert the outputs of the 8-bit ADC to 2 bits. It appears that when the outputs of the four levels of the 2-bit ADC generate approximately the same number of outputs, the receiver produces a better result. This experiment can be used to adjust the RF gain to match the 2-bit ADC.

9.7 Basic FFT Chip Design [2, 3]

The basic chip design follows the FFT operation. As discussed in Section 3.11, the FFT uses the butterfly technique to pass the input data from one layer to another. The operation can be shown symbolically as in Figure 9.11. Since the input has 256 data points, there are 8 layers of processors ($2^8 = 256$). The operations between the layers are the values obtained from the kernel function. Because the kernel function has only four values— $+1$, -1 , $+j$, $-j$ —only additions are needed between layers. A computer program is developed to trace the signal flow.

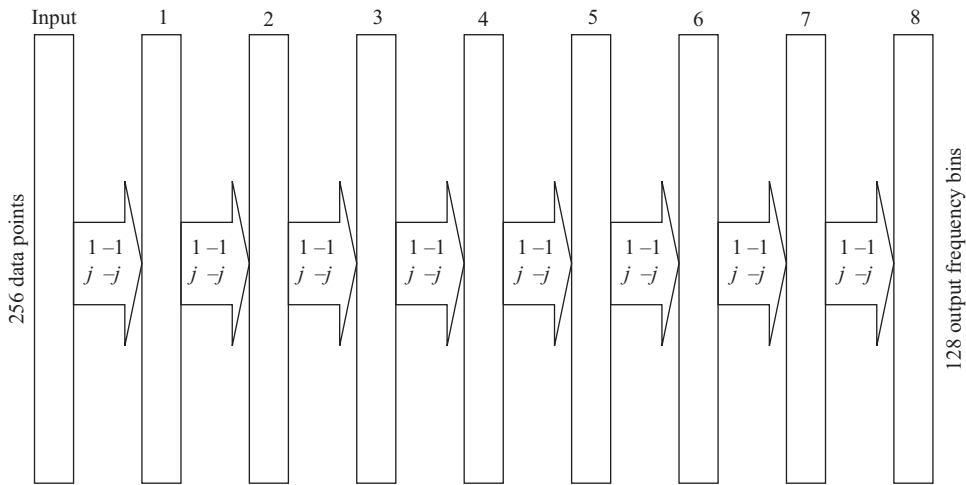


Figure 9.11 Basic FFT layout.

From the signal flow, a table is generated to determine all the adders and inverters used. The final results obtained from inverters and adders are compared with the results calculated from (9.1) and (9.2) to ensure that the design is correct.

A 256-point FFT generates 256 outputs. Because 128 outputs carry redundant information, only 128 outputs are kept as outputs. These 128 outputs cover a 1.25 GHz bandwidth. In order to cover 1 GHz, only 104 outputs ($128/1.25 = 102.4$) are needed. In the demonstration chip, however, all 128 outputs are available.

The input data to the FFT chip have only 2 bits. Because the operation uses addition and subtraction, the input values can be either positive or negative. In order to accommodate these operations, the 1 sign bit is added to the 2 bits of the ADC at the input of the FFT operation. Thus it can be considered as 3 bits to the input of the FFT. After the first butterfly operation, the outputs of the first layer become 4 bits, including 3 amplitude bits and 1 sign bit. The number of bits increases by one from one layer to the next. The final outputs should have 11 bits including the sign bit. In order to simplify the chip design, variations are tested experimentally to reduce the number of bits in the higher layers. After the fifth layer, the output has 8 bits. It has been found that if, after layers 6, 7, and 8, only 8 bits are kept, the results show insignificant change. To truncate the output from 9 to 8 bits, the least significant bit is ignored. This operation is applied to the outputs of layers 6, 7, and 8.

9.8 Frequency Encoder Design [2, 3]

The main purpose of the frequency encoder design is to determine the number of input signals and their frequency. The 128 complex outputs of the FFT are input to the frequency encoder. In the following discussion the values used are either from simulated results or from processing actual digitized data.

From simulated results it is observed that the monobit receiver can process several simultaneous signals. This means that if multiple input signals of comparable amplitude are

present, the FFT outputs will have peaks of the correct frequencies. However, in order to simplify the chip design, only two input signals will be processed by the receiver. With this design goal in mind, the receiver output should be one of three possibilities: no signal, one signal, and two signals. This limited output possibility makes the frequency encoder design relatively simple in comparison with a conventional frequency encoder with a large number of unknown signals.

One obvious approach of the frequency encoder is to find the amplitudes of all 128 frequency components. Because the outputs from the FFT are complex, it is necessary to find their amplitudes through the relation

$$|X(k)| = \sqrt{X_r^2(k) + X_i^2(k)}, \quad (9.3)$$

where $X_r(k)$ and $X_i(k)$ are the real and imaginary parts of the k th frequency components. This operation, however, is rather complicated in a chip design. It is difficult to perform on every one of the 128 outputs.

To avoid this complicated operation, a threshold is set at the FFT outputs. Because the FFT outputs are complex, the thresholds are set for both the real and imaginary parts. The detail of the threshold setting is discussed in the next section. If an FFT output crosses the threshold, it might be an eligible signal. Experimental results show that with proper thresholds, the maximum number of outputs that can cross the thresholds is four. The amplitudes of these outputs are calculated from (9.3). Because the total number is equal to or less than four, the calculation is manageable in the chip design. Another threshold is used to compare with the amplitude of the outputs. If a frequency component crosses the threshold, it will be declared as an input signal.

Figure 9.12 shows the basic functions of the frequency encoder chip. The final output from the frequency encoder can be zero, which means that no input signal is detected; the output can be one specific number, which represents the frequency of the input signal; or the output can be two numbers, which represents the frequencies of two input signals.

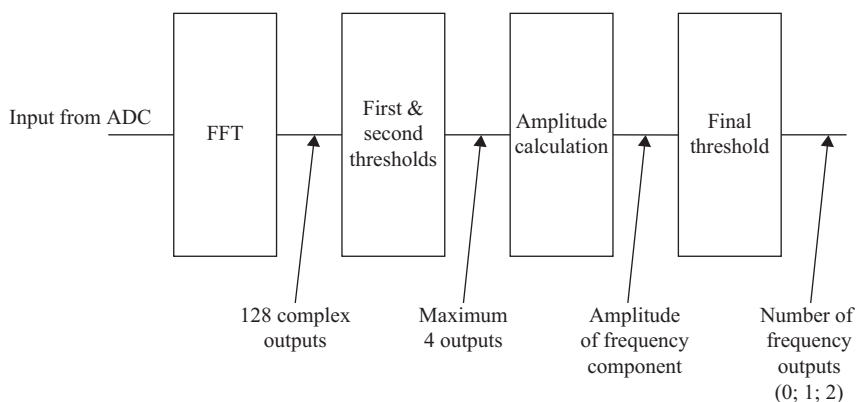


Figure 9.12 The functional block of the frequency encoder.

9.9 Selection of Thresholds [1–3]

Determination of the threshold can be considered one of the most difficult issues in the monobit receiver design. Because the system is nonlinear, it is difficult to analyze it. One way to determine the threshold is through a large number of trials of experimental data.

Before discussing details of threshold selection, let us present the basic requirements. The basic philosophy is as follows: 1) The receiver should only rarely produce a false frequency report, when only noise is present. This is often referred to as the probability of false alarm of the receiver. 2) The receiver should report only one frequency when the input is a single signal. If the receiver reports more than one frequency, the additional signals are often referred to as spurious signals. 3) The receiver should report two frequencies when the input has two signals.

Consider the first requirement. It is usually unacceptable for a receiver to continue reporting false signals. If the threshold is set high enough, the false alarm rate is low; however, the receiver sensitivity is also reduced, which is undesirable. Usually one false report every few tens of seconds is acceptable. Consider requirements 2 and 3. There is usually a compromise. If one wants to reduce the chance of a spurious response, this increases the chance of missing the second signal. The general consensus is that one would rather miss a signal than report a false one. A false signal may cause the signal-sorting processor following the receiver to identify the false signal as a real one. This operation wastes valuable resources and time in the processor.

In determining the threshold of false alarms, the FFT outputs must be measured without input signal. A large amount of data must be processed to estimate the noise spectrum outputs. Each FFT operation utilizes 256 digitized data sampled at 2.5 GHz. The maximum amplitude of the frequency components on each trial is stored. A total of 350,000 sets are processed in about 72 hours. Each 256 data points represent about 100 ns, thus 350,000 records represent 35 ms of data. This process stores 350,000 maximum values. A threshold higher than the highest maximum value must be used to ensure that the receiver does not report a false alarm. Let us refer to this threshold as the preliminary threshold. This threshold is not used as an actual threshold in the encoder design but it will be used to determine the actual threshold. In this test, it only ensures that the receiver will not generate a false alarm during this 35-ms trial. This is a big problem with design by simulation, because so much time is spent to obtain a few results. If the actual hardware receiver is used to monitor the false alarms, it takes only 35 ms to process 35 ms of data.

Once the preliminary threshold is selected, a single signal is used as input to test the receiver response. The desired result is to receive one output for one input signal. However, the test result shows that the receiver may generate more than one output signal report for one input signal. This means that the preliminary threshold is too low. In order to reduce the spurious response, the actual threshold must be higher than the preliminary threshold. This level is referred to as the first threshold level. In determining this threshold, experimental data are used. The experimental data are generated with various input power levels. In general, strong input signals are used for this testing. This threshold is used in the encoder design. This first threshold is chosen so that most of the time one input signal generates only one output signal report.

Once the first threshold is determined in the encoder design, two input signals are applied and tested. Because the front end is highly nonlinear, caused by the limiting amplifier and the

2-bit ADC, the two signals can interfere with each other, as discussed in Section 9.6. As a result, sometimes the first threshold is not crossed, even if two strong input signals are applied to the input of the receiver. This condition is obviously not acceptable. To remedy this problem a second threshold is selected. This threshold is lower than the first one but higher than the preliminary. Figure 9.13 shows the two-threshold arrangement. Figure 9.13(a) shows the single signal condition. In this figure, one signal crosses the first threshold and a spur crosses the second threshold. If the second threshold is used alone, the receiver will report one false signal.

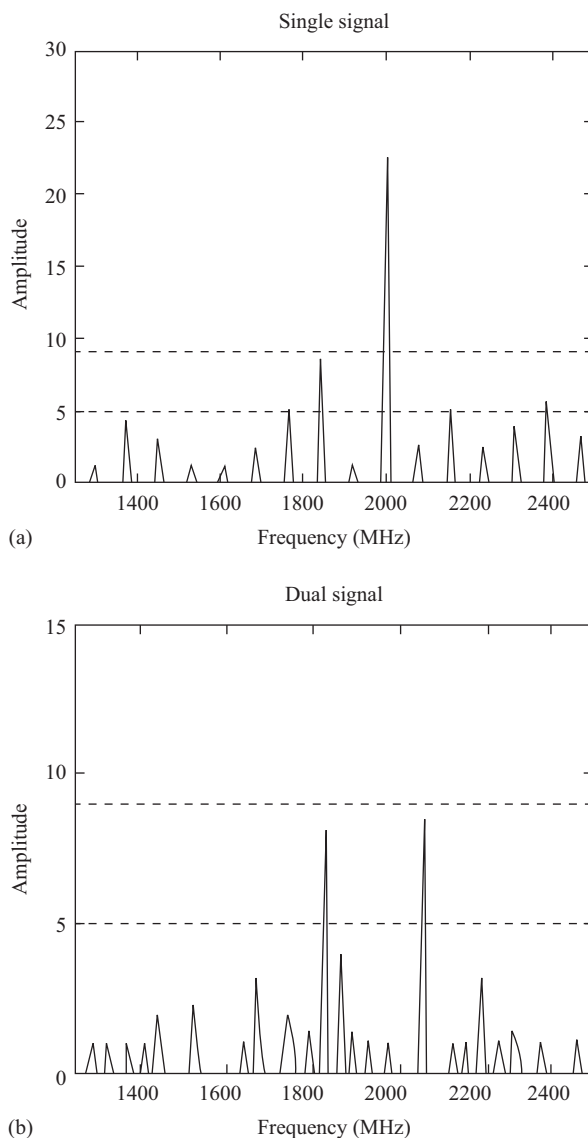


Figure 9.13 Two-threshold arrangement: (a) single input signal and (b) two input signals.

Figure 9.13(b) shows the two-signal condition. In this figure, both signals are under the first threshold but cross the second one. If the first threshold is used alone, the receiver will miss both signals. Because the FFT outputs are complex, both the first and the second thresholds have real and imaginary parts. Figure 9.13 shows only the real portion of the threshold. The spectrum is the real part also. The imaginary part displays similar results.

The operation of the two-threshold arrangement is as follows. Initially the first threshold is tested. If the FFT outputs cross this level, these outputs are kept and the second threshold will not be tested. If the first threshold is not crossed, the second threshold is tested. The FFT outputs crossing this threshold are kept. Limited experimental results show that most of the time four or less than four FFT outputs can cross the thresholds. When two input signals are exactly the same amplitude, a very small percentage of time more than four outputs will cross the threshold. If this situation occurs, the first four outputs from the output order (not the largest four) are selected. Under this condition a real input signal might be missed.

The amplitudes of the FFT outputs crossing the thresholds are calculated from (9.3). These amplitudes will be compared with a threshold, which is referred to as the final threshold. If no signal is present at the input of the receiver, the first and second thresholds will not be crossed. If one or two FFT outputs cross the final threshold, the receiver will report the number and the frequency of the signals. If more than two FFT outputs cross the final threshold, only the two largest values are kept as the measured signals. This is the way to limit the receiver to process only two signals. Once the number of signals is determined, their corresponding frequencies can be obtained from the numerical FFT frequency bins.

9.10 Preliminary Performance of a Monobit Receiver

In order to test a receiver, a computer must be connected to the outputs of the encoder. The results from the encoder are compared with the input signal to determine the errors measured. The outputs are usually measured in a statistical manner by repeating the same input many times, say 100 to 1000 times. A percentage of erroneous reports can be obtained using this testing method. However, the monobit receiver outputs cannot yet be read by a computer. Only a limited number of input conditions can be evaluated, therefore the performance of this is referred to as preliminary performance data.

First, the RF chain is not included in the test. The purpose of this test is to test the ADC, the FFT, and frequency encoder chip. The ADC can digitize very high frequency input signals such as 10 GHz. Because the ADC will downconvert the input to the baseband as part of the ADC process, the FFT chip can process the signal. It has been demonstrated that the ADC and FFT chip can process input signals up to 10 GHz. This might be a very important factor in designing the wideband RF chain. The term “wideband RF chain” is used to distinguish from the RF chain shown in Figure 9.6. As mentioned in Section 2.4 for EW applications, the frequency of interest is from 2 to 18 GHz and the baseband receiver has only a 1-GHz bandwidth. The conventional approach is to divide the 2 to 18 GHz frequency range into 1-GHz bands. The frequency of each band is shifted to the input of the baseband receiver through frequency conversion. In accomplishing this, mixers and local oscillators are needed. Because the ADC can take inputs from 2 to 10 GHz directly, filters and wideband amplifiers might be enough to build the wideband RF chain. This approach can eliminate the mixers and local oscillators.

Table 9.1 Preliminary performance of the monobit receiver.

Input frequency (GHz)	1.375–2.375
Two-frequency resolution (MHz)	10
Single-frequency resolution (MHz)	10
Sensitivity (dBm)	–70
Dynamic range (dB)	
Single-signal	75
Two-signal spur-free	70
Instantaneous	5
Minimum pulse (ns)	200
Time of arrival (ns)	102.4
Number of signal capability	2

The monobit receiver completed with the RF chain and the FFT chip has been tested. In this case the RF chain limits the input bandwidth to 1 GHz. The performance is listed in Table 9.1.

The input frequency range is equal to the design goal. When two signals are separated by about 10 MHz, the receiver is able to measure them. The single-signal frequency resolution is 9.77 MHz, which is obtained from $1250/128$. The sensitivity of the receiver is usually measured across the input frequency range of 1 GHz. For this receiver, however, the sensitivity is measured over a few frequency values and the approximate value is about -70 dBm. There is no probability of false alarm nor probability of detection associated with this sensitivity. The single-signal dynamic range is 80 dB since the receiver can process a strong input signal at 10 dBm. The two-signal spur-free (or third-order intermodulation) dynamic range is rather high compared with a conventional receiver. The reason is that the receiver can process only two simultaneous signals, thus it cannot detect the third signal, which determines the lower limit of the spur-free dynamic range.

The instantaneous dynamic range is about 5 dB; when two signals are within 5 dB in amplitude, it usually means that the receiver can measure both of them. This definition does not apply to the monobit receiver. When two signals are of the same amplitude, the receiver does not report them all the time. The receiver reports both signals only about 24% of the time. About 76% of the time the receiver only reports one signal. When two signals are present at the input of the receiver, it does not report an erroneous frequency as does an IFM receiver. The receiver either reports one frequency correctly or reports both frequencies correctly. When two input signals are separated by more than 5 dB, the receiver will report only the frequency of the strong one. The above definition is referred to as the instantaneous dynamic range, which is different from the conventional definition. In the conventional definition, the receiver must measure both signals, which is more stringent than the definition used here. When three signals of equal amplitude are presented at the input of the receiver, the receiver often reports one or two frequencies correctly. If the third signal is weak, it usually will not affect the frequency measurement.

Very limited data are collected from the above test. Some of the results are obtained from the design stage, and the input data are collected from an 8-bit ADC and converted to 2 bits through a software program. Two input frequencies are randomly selected and their amplitude difference is kept constant. For each power difference, 1000 sets of data are collected on

Table 9.2 Performance on two simultaneous signals.

Amplitude of second signal vs. first (dB)	Found first signal (%)	Found second signal (%)	Found both signals (%)	Found neither signal (%)	Found erroneous signals (%)
0	65.1	59.3	24.4	0	0
−1	78.9	45.0	23.9	0	0.45
−2	89.2	29.9	20.9	0	0.38
−3	93.9	18.0	12.0	0.13	0.38
−4	97.9	9.5	7.6	0.13	0.25
−5	99.8	3.3	3.0	0	0.13

the signals with random frequencies. These data are listed in Table 9.2. The purpose is to provide a rough idea of the receiver performance.

The receiver occasionally misses pulses and sometimes generates erroneous signals. An erroneous signal is defined as a signal whose measured frequency is off by 6 MHz, which is slightly larger than half the frequency resolution bin of 9.77 MHz. A rigorous test on a hardware monobit is needed to generate quantitative results.

The minimum PW is about twice the FFT frame time (204.8 ns) because this PW guarantees to fill one FFT frame (102.4 ns). Test results indicate that a 100-ns pulse can be detected by the receiver, but not 100% of the time. A 150-ns pulse can be detected most of the time. Because only limited trials can be accomplished without an automatic testing setup, the minimum pulse capability is given as 200 ns. The time resolution is 102.4 ns.

The two major deficiencies of this receiver are the limited instantaneous dynamic range and the two-signal capability. It is desirable to have a higher instantaneous dynamic range and the ability to process more than two simultaneous signals.

9.11 Possible Improvements

It is desirable to eliminate the two deficiencies of the receiver. Possible approaches are to increase the bit number of the ADC or change the kernel function of the receiver. Tests have indicated that increasing the input number of bits with 1 bit kernel function of $\pm 1, \pm j$, results in little change to the FFT outputs. Changing the kernel function should improve the performance. However, increasing the number of bits in the kernel function requires multiplication, which defeats the concept of minimal hardware and processing used in the monobit receiver.

One possible approach is to increase the number of bits in the kernel function and still keep the FFT operation limited to additions. Increasing the kernel function from four points to eight points will increase four points at $(1 + j)/\sqrt{2}$, $(1 - j)/\sqrt{2}$, $(-1 - j)/\sqrt{2}$, and $(-1 + j)/\sqrt{2}$. These four points are on the unit circle. Because the amplitude has the factor $1/\sqrt{2}$, multiplication is needed to produce the FFT outputs. If these four points are moved toward the corner of a square with unity sides, the additional four points become $1 + j$, $1 - j$, $-1 - j$, and $-1 + j$. This operation is illustrated in Figure 9.14. Because the factor $1/\sqrt{2}$ is eliminated, multiplication is no longer needed in producing the FFT outputs.

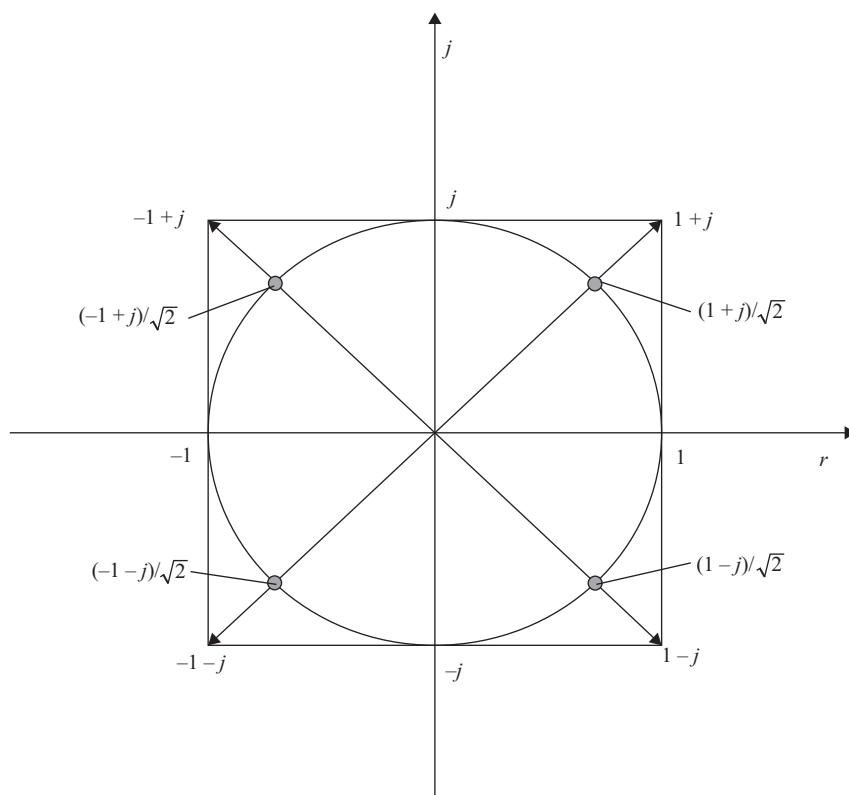


Figure 9.14 Moving the kernel function values.

Simulated data are used to test this concept. These eight points are obtained from Figure 9.15. The values of the 256 points of the kernel function can be divided into eight regions. There are 32 points in one region. All the values in one region can be represented by one point. After this modification the input data are increased to 3 bits. Limited trials are then tested and the results indicate that this arrangement can improve the dynamic range slightly.

From observing the FFT outputs, it appears that the peaks of the spurious signals are slightly lower (about 3 dB) than that of the monobit receiver. Because of the sidelobe limit of the rectangular window and the low number of bits of the ADC, the dynamic range cannot be expected to improve drastically. As mentioned in Section 5.7, every bit can provide about 6 dB in dynamic range. It also appears that three signals of the same amplitude can produce three peaks at the correct frequency bins. This phenomenon indicates that the modified kernel function can still process three signals.

If this idea is implemented in a receiver design, the FFT will be more complicated because of the additional terms in the kernel function. The frequency encoder might also be more complicated if the receiver is designed to process three signals, because the final results could be 0, 1, 2, and 3. However, one can also design a receiver to process two signals with

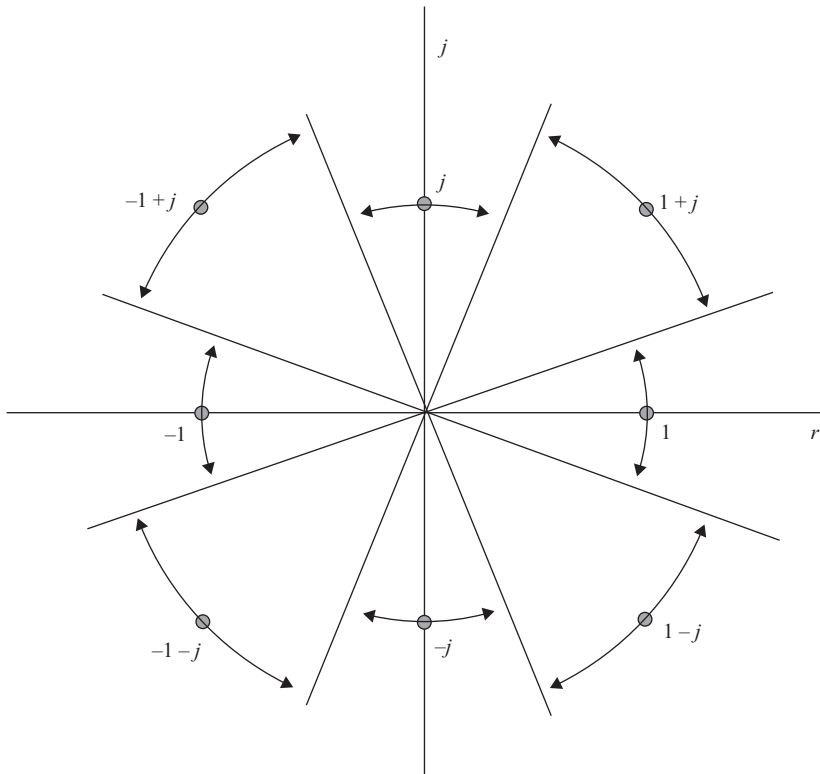


Figure 9.15 Digitizing the Kernel function values.

the idea to improve instantaneous dynamic range. Because of the potentially limited performance improvement and the complication of the chip design, this idea has not been investigated thoroughly.

9.12 Chip Layout [2, 3]

This last section provides some information on the chip layout. Although it is possible to build the chip on a field programmable gate array (FPGA), the first chip is built with application-specific integrated circuit (ASIC) technology. ASIC uses double metal 0.5- μm scalable CMOS technology and builds in an 84-pin package. It operates at 156.25 MHz (2.5 GHz/16) since the demultiplexer is 1 to 16. The chip contains 812,931 transistors and has a die size of approximately 15 mm $\times$ 15 mm. The chip is broken down into five subsystems, as discussed in previous sections. These are the input stage, FFT block, initial sorting, squaring and addition, and final sorting. The process in each subsystem must be completed within 102.4 ns (0.4×256), which is the time it takes to accumulate 256 data points. The timing of the subsystems is simulated and the results are listed in Table 9.3. The timing analysis includes the delay of each pipelined flip-flop.

Table 9.3 Timing analysis of each subsystem.

Subsystem	Critical path (ns)
Input stage	99.5
FFT block	48.02
Initial sorting	90.11
Squaring and addition	28.95
Final sorting	34.42

Although the input stage shows the longest time, it is not the time of concern since this is the time for the chip to take all the input data. It takes about 102.4 ns to collect the 256 points of data. The longest time required is in the initial sorting. After the chip is fabricated, it performs satisfactorily. From this brief discussion it can be seen that the chip is rather simple.

References

- [1] Tsui JBY, Schamus JJ, Kaneshiro DH. 'Monobit receiver'. *Proceedings of the IEEE MTT-S International Microwave Symposium*, vol. 2. New York: IEEE; 1997:469–471.
- [2] Pok D, Chen C-IH, Montgomery C, Tsui BY, Schamus J. 'ASIC design for monobit receiver'. *Proceedings of Tenth Annual IEEE International ASIC Conference and Exhibit*. New York: IEEE; 1997:142–146.
- [3] Pok DSK, Chen C-IH, Schamus JJ, Montgomery CT, Tsui JBY. 'Chip design for monobit receiver'. *IEEE Transactions on Microwave Theory and Techniques* 1997;**45**(12):2283–2295.
- [4] Grajal J, Blazquez R, Lopez-Risueno G, Sanz JM, Burgos M, Asensio A. 'Analysis and characterization of a monobit receiver for electronic warfare'. *IEEE Transactions on Aerospace and Electronic Systems* 2003;**39**(1):244–258.

Frequency Channelization and Afterwards Processing

10.1 Introduction

Channelization is one of the most important operations in building digital electronic warfare (EW) receivers. The equivalent analog operation is the filter bank. Therefore digital channelization can be considered a digital filter bank. It can also be considered as an N -port network with one input and $N - 1$ outputs. An input signal will appear at a certain output according to its frequency. By measuring the outputs from the filter bank, the frequency of the input signal can be determined.

The only practical approach to building a wideband digital EW receiver with today's technology is through channelization. A common method of performing channelization is by employing the fast Fourier transform (FFT). To build a receiver using FFT, the length and the overlap of the FFT are very important parameters. These parameters are related to the minimum pulse width (PW) and the frequency resolution, which determines the sensitivity of the receiver. The frequency information can be obtained from the outputs of the digital filters. In order to obtain the input frequency, the filter outputs must be further processed. The main objectives of a receiver are to determine the number of input signals and their frequencies. The circuit used to accomplish these goals is referred to as the encoder.

The encoding circuit is the most difficult subsystem to design in an EW receiver and most research effort is spent on the encoder design. This is true for both digital and analog receivers. The main problems are to avoid the generation of false alarms and the detection of weak signals. In an analog filter bank, the shape of the filter is difficult to control, and it is difficult to build filters with uniform performance, such as bandwidth and ripple factor, therefore the encoder must accommodate this problem. The shape of each individual filter in a digital filter bank can be better controlled. As a result, the encoder should be slightly easier to design because it does not need to compensate for the filter differences. Because of the complexity of the encoder, its design will not be discussed in detail.

The design of a specific digital filter bank will also be presented. This specific example is used to illustrate the design procedure while avoiding the unnecessary mathematical complexity of a general design. In this example, the concepts of polyphase filters and multirate operation are introduced. In order to understand these concepts, decimation and interpolation are discussed first.

10.2 Filter Banks and FFT [1, 2]

The straightforward approach for building a filter bank is to build individual filters, each one with a specific center frequency and bandwidth. Figure 10.1 shows such an arrangement. Each digital filter can be either a finite impulse response (FIR) or infinite impulse response (IIR) type. Theoretically, each filter can be designed independently with a different bandwidth or shape. In this arrangement, if the input data are real (as opposed to being complex) the output data are also real. The output is obtained through convolving the input signal $x(n)$ and the impulse response of the filter $h(n)$. One of the disadvantages of this approach is that the operation of the filter bank is computationally complex.

It is desirable to build a receiver with uniform frequency resolution; that is, the filters have the same shape and bandwidth. Since FFT converts time domain data to frequency domain data and each element of the FFT results represents the signal's component at a certain frequency, we can consider FFT as a kind of filter bank. It is easier to build such a filter bank through FFT techniques than by using individual filter design since there is less computation.

The similarity between the FFT and convolution operation is illustrated in the following paragraphs. In the FFT operation, one set of data in the time domain can be used to find one set of data in the frequency domain. In order to process the input data in a continuous manner, the FFT must also operate continuously. This subject is further discussed in Sections 10.3 and 10.4.

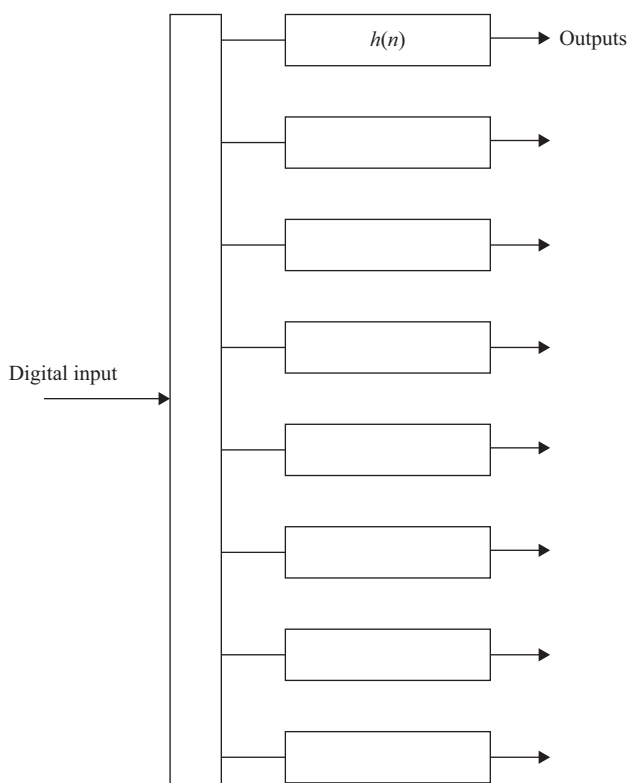


Figure 10.1 A filter bank.

This discussion is similar to that in Kay [1]. Let us assume that one frequency component from the FFT output is equivalent to one filter output at a specific point in time from a filter bank. The output of the k component $X(k)$ from an N -point FFT can be written as

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}. \quad (10.1)$$

In order to relate this output to convolution, let us define an impulse function as

$$h(k) = e^{j2\pi kk_0/N}, \quad -(N-1) \leq k \leq 0, \quad (10.2)$$

thus

$$X(k_0) = \sum_{n=0}^{N-1} x(n) h(k-n)|_{k=0}. \quad (10.3)$$

One can see that this expression is a discrete convolution as shown in (3.107). It represents the input signal $x(n)$ convolving with $h(n)$.

This operation illustrates that a certain frequency bin from the FFT operation can be treated as an input signal convolved with a certain impulse function. Therefore one can consider that each individual FFT output can be represented by a filter impulse function convolved with the input signal. Because the FFT operation is rather simple compared with an individual filter design, the FFT is used for filter bank design in the rest of this chapter. A similar discussion showing that one frequency bin of the FFT outputs is equivalent to a convolution of the input signal with a filter impulse can be found in Harris [2].

10.3 Overlapping Input Data in the FFT Operation [2–4]

In the previous section it was demonstrated that each FFT output can be considered as a filter output. In order to operate on a continuous input signal, the FFT must operate on different intervals of data at different times. Usually the initial data point is labeled $n = 0$ and the data interval can slide M points and be represented by $n = M$. The corresponding FFT can be written as

$$X(k) = \sum_{n=M}^{N+M-1} x(n) e^{-j2\pi kn/N}. \quad (10.4)$$

The M value must be changed continuously with the input signal. This operation is sometimes referred to as the short time Fourier transform (STFT).

Figure 10.2 illustrates the input data overlapping condition. In this figure the FFT uses only 8 data points. When $M = 0, 1, 2, \dots$, as shown in Figure 10.2(a), the input data slide one point every time, which is referred to as the sliding discrete Fourier transform (DFT) and is discussed in Section 3.12. For this case the data can be considered 100% overlapping. If the minimum PW is 8 data points long, this approach can always fill one of the FFT windows with the shortest pulse. The TOA resolution mentioned in Section 2.6 is equal to the t_s , where $t_s = 1/f_s$ is the sampling time and f_s is the sampling frequency. With 100% overlapping, the FFT must be performed every t_s , thus the computation load is very high.

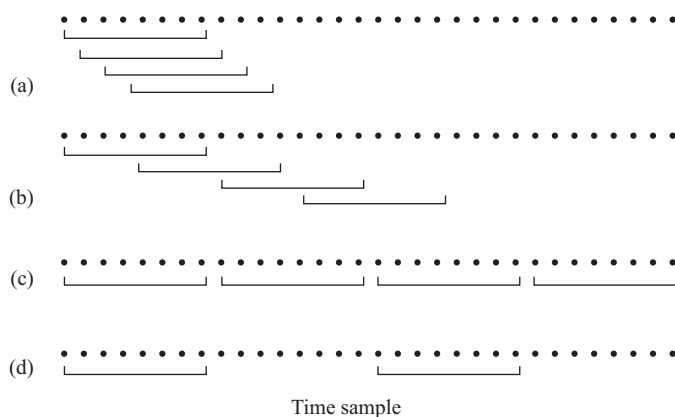


Figure 10.2 Data string in the time domain: (a) 100% overlap, (b) 50% overlap, (c) zero overlap, and (d) missing 50% of data.

Figure 10.2(b) shows 50% data overlapping, which corresponds to $M = 0, 4, 8, \dots$. The shortest pulse under the worst-case condition can fill 75% of the window. Under this condition the sensitivity of the receiver will degrade. The TOA resolution will be reduced to $4t_s$. The FFT, however, only operates every $4t_s$, which means less computation is required.

Figure 10.2(c) shows zero overlapping, but no missing data either. The corresponding $M = 0, 8, 16, \dots$. The shortest pulse under the worst-case condition can fill 50% of the window. The TOA resolution and the FFT operation rate are both decreased to $8t_s$. This situation usually provides the lowest acceptable FFT operation rate. This phenomenon will be explained later in this section.

Figure 10.2(d) shows that some data are missing. For this illustration, $M = 0, 16, 32, \dots$. In general this is not an acceptable choice because the receiver will miss pulses.

The degradation of filling half a window is illustrated in Figure 10.3 with 64 input data points. Figure 10.3(a) shows that the data fill a rectangular window in the time domain and Figure 10.3(b) shows that the data fill only half of the window. The corresponding FFT outputs are shown in Figures 10.3(c) and 10.3(d). In order to smooth the outputs in the frequency domain, zero padding is used in performing the FFT. The spectrum in Figure 10.3(d) not only decreases in amplitude, but also spreads in frequency. This frequency spreading causes difficulty in designing the parameter encoder following the FFT outputs. Even for a long pulse, the signal usually fills the window only partially at the leading and trailing edges. These partially filled windows will spread the energy in the frequency domain to adjacent channels and may cause detection problems.

The selection of the FFT operation rate depends on the technology development of the FFT chip. With today's technology, the FFT operation speed is much less than the sampling rate of the analog-to-digital converter (ADC). This point is one key factor in determining the receiver design approach.

10.4 Output Data Rate from FFT Operation [2–6]

If the FFT operation is performed on N points of data, the resulting N frequency outputs will occur at the same time. If the sampling frequency is f_s , which is the input data rate, the

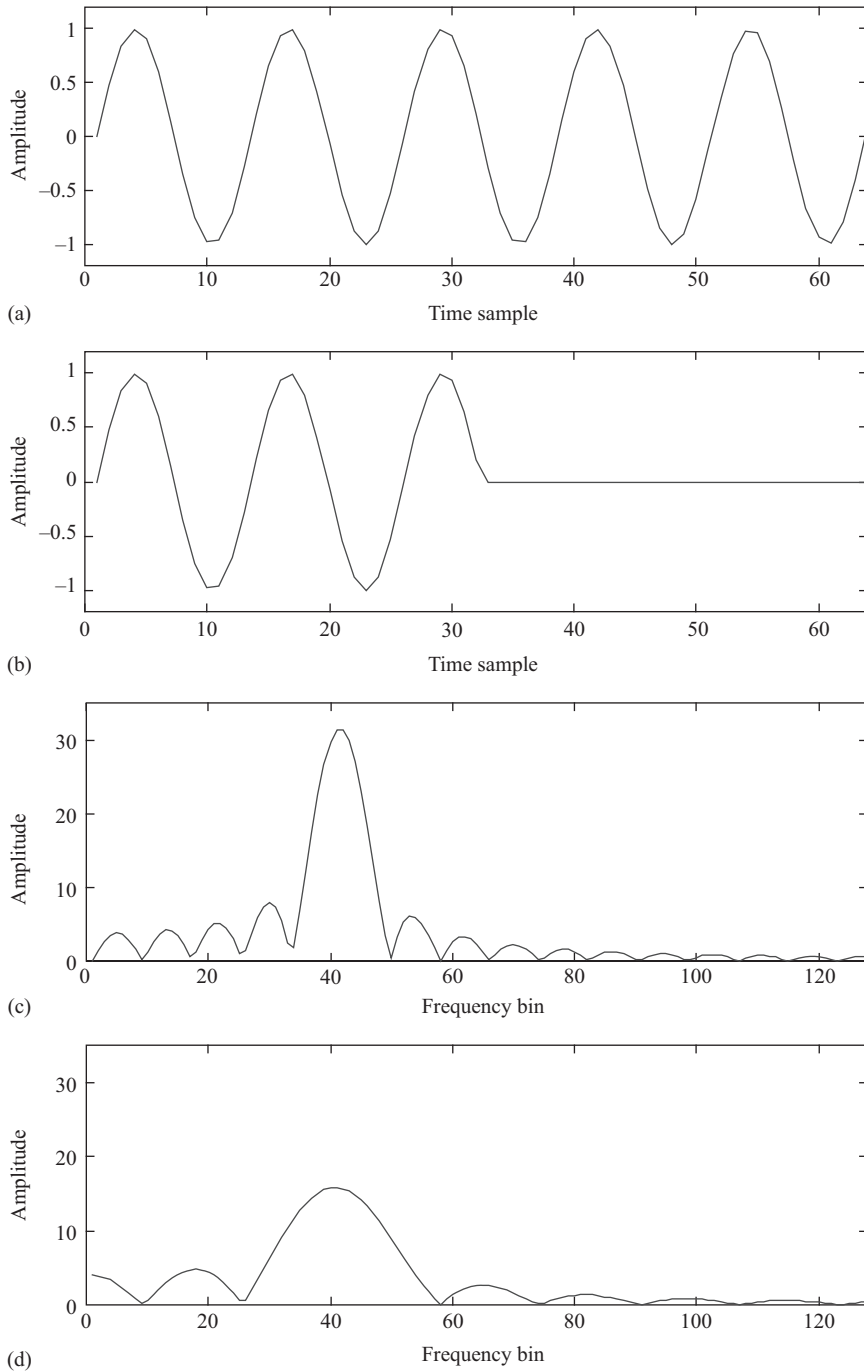


Figure 10.3 FFT on filled and partially filled windows: (a) signal filled window (time domain), (b) partially filled window (time domain), (c) signal filled window (frequency domain), and (d) partially filled window (frequency domain).

corresponding input sampling time is $t_s = 1/f_s$. The output rate depends on the data overlapping rate discussed in the previous section. For example, if the input data are 100% overlapped as shown in Figure 10.2(a), the output rate is also f_s . If the input data are shifted by M samples between every FFT operation, the output sampling time is Mt_s , which corresponds to an output sampling rate of f_s/M .

The output sampling rate is very important in building a receiver because the outputs from the FFT are usually further processed to obtain finer frequency resolution. For this processor, the input rate is the output sampling rate from the FFT. This output sampling rate determines the bandwidth of this processor. If a rectangular window containing N points is used to process the input signal with zero overlapping, the output sampling rate is f_s/N . The corresponding output bandwidth is about f_s/N . For a rectangular window in the time domain the corresponding output in the frequency domain is a sinc function and the response is shown in Figure 10.4. This shape represents the response of one filter output. Figure 10.4(a) shows the detailed shape of this filter and Figure 10.4(b) shows three adjacent filters. This filter shape is not desirable because the sidelobes are very high and the first sidelobe is only 13 dB down. If a signal of frequency f_i falls in channels B and C, it will also enter channel A through its first sidelobe. This phenomenon limits the instantaneous dynamic range of the receiver to less than 13 dB.

Because the output bandwidth is limited to f_s/N , channel A cannot properly process a signal at f_i because it is outside of its bandwidth. If channel A processes this signal, it may assign an erroneous frequency. A higher output sampling rate will help this situation.

To lower the sidelobes of the filter, a weighting function can be applied in the time domain. A weighting function widens the main lobe but suppresses the sidelobes. Because the main lobe is wide, fewer channels are needed to cover the desired input bandwidth. In order to further process the signal, the output sampling rate must be increased to match the bandwidth. Figure 10.5 shows the FFT output of a Hamming window (Section 3.10). The need for increasing the output sampling rate can also be explained in the time domain. Figure 10.6 shows the time domain of a Hamming function. One can see that only the data near the center of the weighting function are given weights near unity. The data close to the edges of the window are heavily attenuated. If a zero overlapping approach is used, the contribution from these data in the FFT operation will be nonuniform, which is undesirable. If the windows are overlapped in the time domain, which is equivalent to increasing the output sampling rate, this deficiency can be remedied.

10.5 Decimation and Interpolation [7–10]

The concepts of decimation and interpolation are discussed briefly in this section since they will be used to design filter banks. Decimation of data means only using one data point in a group of data. For example, if the data points are $x(n)$, where $n = 0, 1, 2, 3, \dots$, the decimated result x_d by M is

$$x_c = x(Mn), \quad (10.5)$$

where M is an integer. In other words, one data point is selected every M points of data. If $M = 2$, $x_d(0) = x(0)$, $x_d(1) = x(2)$, $x_d(2) = x(4)$, $\dots$. If $M = 3$, $x_d(0) = x(0)$, $x_d(1) = x(3)$, $x_d(2) = x(6)$, $\dots$. These results are shown in Figure 10.7. Figure 10.7(a) shows the original data and Figures 10.7(b) and 10.7(c) show the results decimated by 2 and 3, respectively. Decimation by M can be represented by $M\downarrow$.

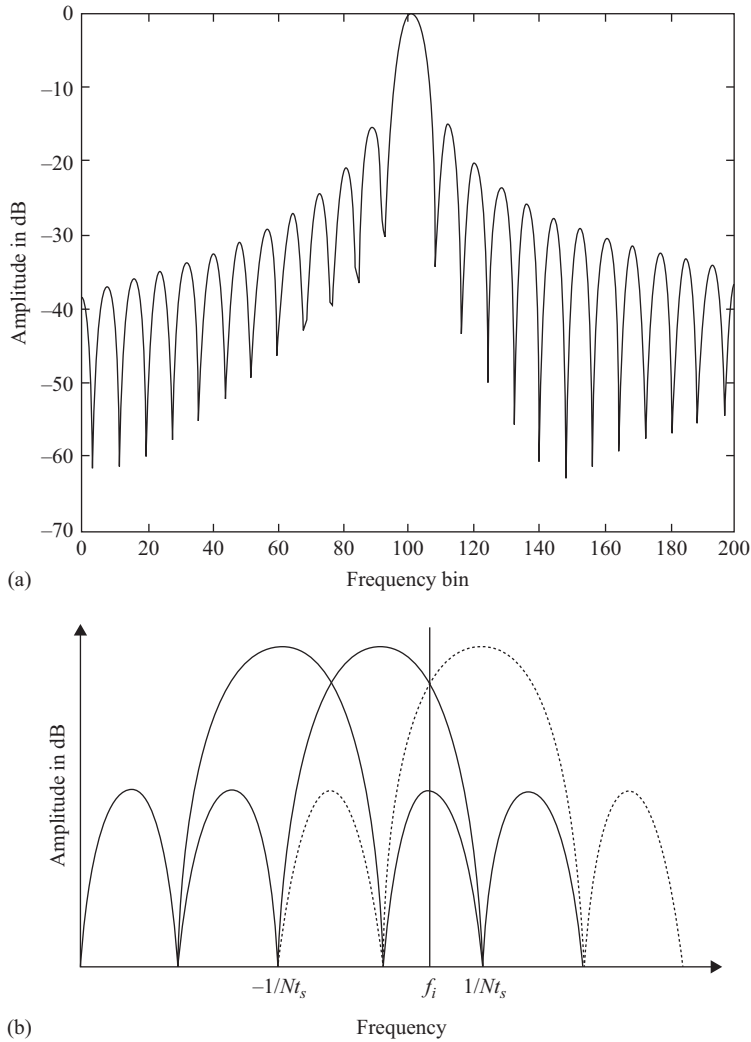


Figure 10.4 FFT of a rectangular window: (a) detailed filter response and (b) three adjacent filters.

It is obvious that decimation loses information. If a signal is sampled at 3 GHz, decimation by 2 is equivalent to sampling the data at 1.5 GHz. Decimation by 3 is equivalent to sampling the data at 1 GHz.

In interpolation, additional data points are added to the original input data. Because it is difficult to add information to the data, only zeros will be added. Mathematically, if the input data are interpolated by L , the result $x_l(n)$ can be related to the input signal as

$$x_l(n) = \begin{cases} x\left(\frac{n}{L}\right) & \text{if } n/L = \text{integer} \\ 0 & \text{if } n/L \neq \text{integer} \end{cases}, \quad (10.6)$$

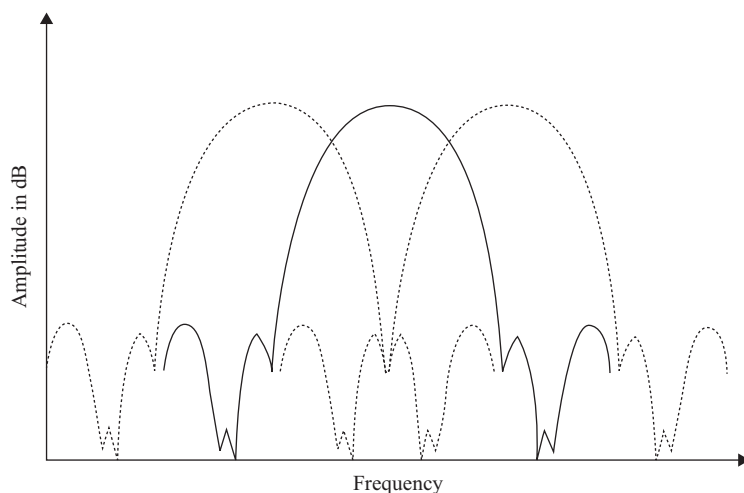


Figure 10.5 Outputs from three adjacent filters with Hamming window.

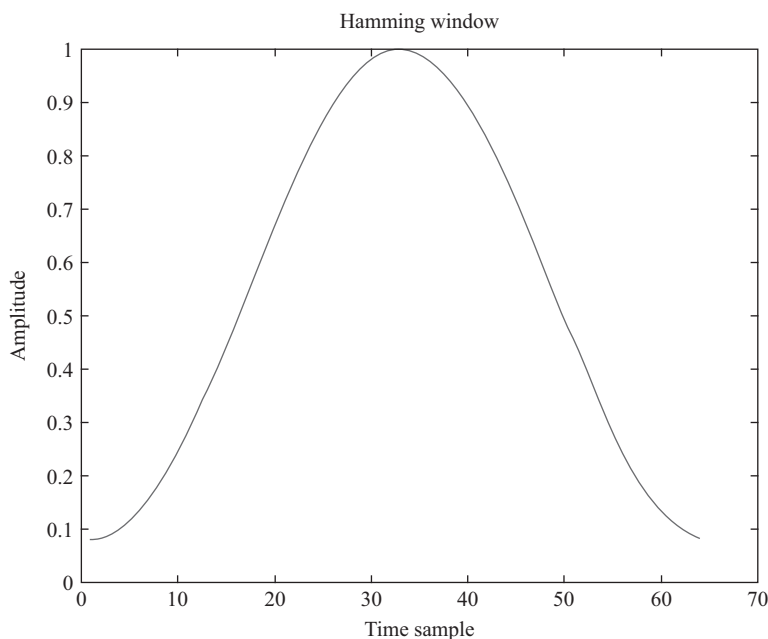


Figure 10.6 Hamming window in the time domain.

where L is an integer. If $L = 3$, $x_l(0) = x(0)$, $x_l(1) = 0$, $x_l(2) = 0$, $x_l(3) = x(1)$, $x_l(4) = 0, \dots$. Figure 10.8 shows the results of interpolation by 2 and 3. As expected, interpolation does not add information to the input data. Interpolation is represented by $L\uparrow$.

If the input data are interpolated by L and decimated by L , the original data will be obtained. This is obvious because interpolation by L adds $L - 1$ zeros per data point and decimation by L takes the added zeros away. If, however, the data are decimated by M first

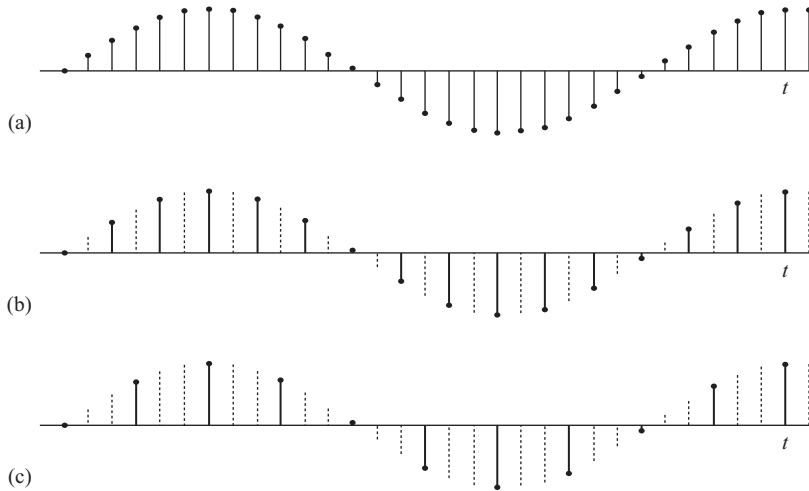


Figure 10.7 Example of decimation: (a) input data, (b) decimated by 2, and (c) decimated by 3.

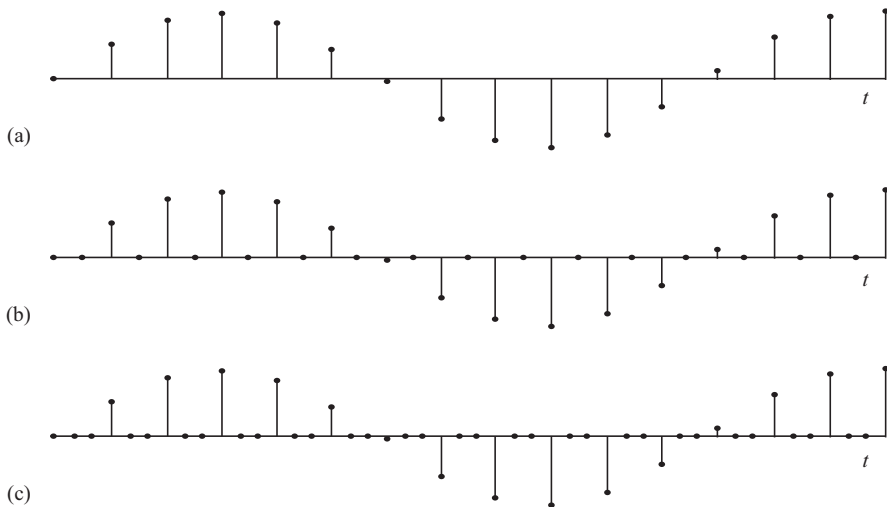


Figure 10.8 Example of interpolation: (a) input data, (b) interpolation by 2, and (c) interpolation by 3.

and interpolated by M , the process will not produce the original data. The decimation process loses data; the interpolation process cannot recover lost data.

10.6 Decimation and Interpolation Effects on the DFT [7–10]

In this section the decimation and interpolation effects on the DFT are presented. As mentioned in the previous section, decimation slows down the sampling rate by a factor of M . If the input data are real (as opposed to being complex) and sampled at f_s , the input bandwidth is $f_s/2$. If an N -point FFT is performed on the input data, the frequency resolution is f_s/N .

If the N -point FFT is performed on the data decimated by M , the input bandwidth is reduced to $f_s/(2M)$ and the resolution is $f_s/(MN)$. Thus the input bandwidth is reduced by M , but the width of the frequency resolution cell is increased by M .

The effect of the FFT with interpolated data is slightly more complicated. The effect of the FFT can be written as follows. If the input data $x(n)$ are interpolated by L , the resulting data $x_I(n)$ have NL points. The FFT of $x_I(n)$ can be written as

$$X_I(k) = \sum_{n=0}^{NL-1} x_I(n) e^{-j2\pi nk/NL}. \quad (10.7)$$

From the previous equation, whenever $n/L \neq \text{integer}$, $x_I(n) = 0$ and $n/L = \text{integer}$, $x_I(n) = x(n)$. Therefore this equation can be written as

$$X_I(k) = \sum_{n=0}^{NL-1} x_I(n) e^{-j2\pi nk/NL} = \sum_{n=0, \text{step } L}^{NL-1} x\left(\frac{n}{L}\right) e^{-j2\pi nk/NL}. \quad (10.8)$$

For the frequency component of $k < N$ the above function can be written as

$$X_I(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi nk/N}, \quad (10.9)$$

which is the same result obtained from a conventional FFT with the original data. Because there are NL data points in the interpolated data, the FFT will have NL frequency components. The value of $k > N$ can be found by replacing $k = mN + k'$ in the above equation, where m is an integer. The result is

$$X(k) = X(mN + k') = \sum_{n=0}^{N-1} x(n) e^{-j2\pi n(mN+k')/N} = \sum_{n=0}^{N-1} x(n) e^{-j2\pi nk'/N}. \quad (10.10)$$

This equation shows that the output data are periodic with a period of N .

Figure 10.9 shows an example of this operation. Figure 10.9(a) shows the amplitude of the FFT outputs of a sine wave; the output has 16 frequency components. Figure 10.9(b) shows the same data interpolated by 3. The output has 48 frequency components consisting of three cycles. In general, if the input data are interpolated by L , the FFT outputs will have L cycles.

From this example one can see that an interpolator does not introduce any additional information in the frequency domain. The additional data points result in many repeated cycles. This effect is different from the zero padding operation discussed in Section 4.3. Although zero padding does not add more information to the input data, it interpolates the FFT outputs and a better frequency estimate can be obtained.

10.7 Filter Bank Design Methodology

In the next five sections, a filter bank design is presented. Let us assume that the ADC has 8 bits and operates at 3 GHz. Under this sampling rate, a PW of 100 ns contains 300 data points. A base-2 number such as 256 can be chosen as the FFT length, which is slightly less than 100 ns. This value is slightly less than the minimum PW required in Section 2.19.

Performing a 256-point FFT with no overlapping data, the FFT must operate at about 11.72 MHz (3000 MHz/256). From Section 10.2, one can see that using a rectangular

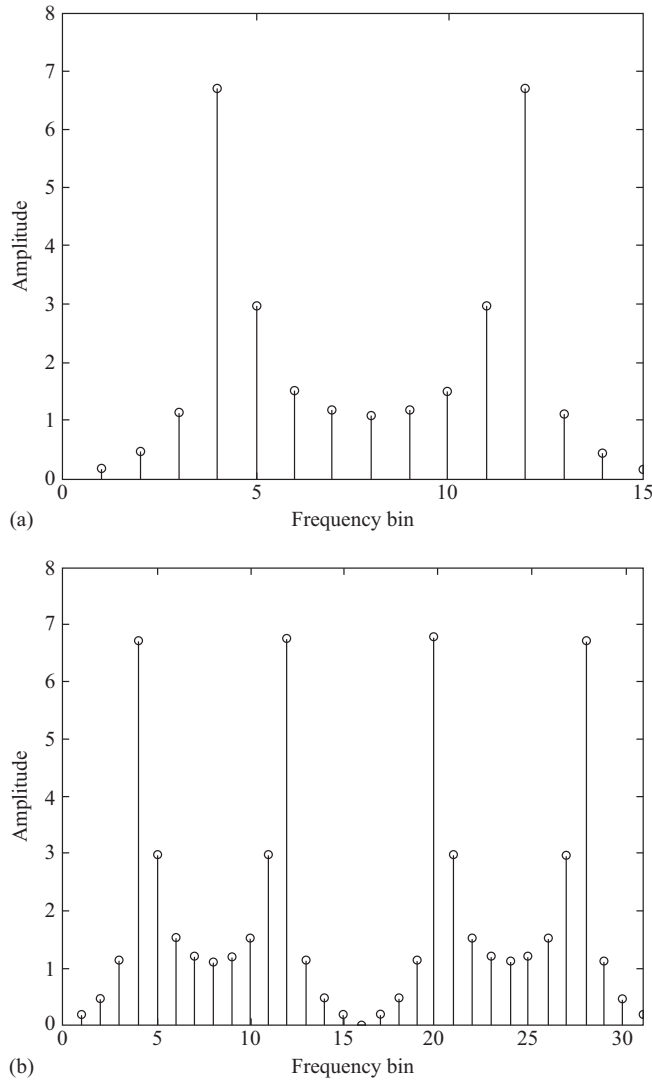


Figure 10.9 FFT of interpolated outputs: (a) 16-point FFT and (b) interpolated by 2.

window results in a receiver with low instantaneous dynamic range due to the window's high sidelobes. In order to suppress the sidelobes, a weighting function (or window) must be applied. As mentioned in Section 10.3, a window in the time domain increases the main lobe width in the frequency domain. This main lobe width is the bandwidth of each individual filter. Increasing the filter bandwidth reduces the number of filters required to cover the desired input bandwidth. A window function, however, necessitates an increase in the processing rate of the FFT operation. No data overlapping operation with a window can cause different contributions from data in the time domain. This is undesirable, as discussed in Section 10.4. As the result, some overlapping is necessary.

10.8 Decimation in the Frequency Domain

Decimation can also be used in frequency domain processing. In this section the FFT outputs are decimated. This operation can decrease the complexity of the FFT operation. Instead of presenting a general case, a special case is presented because the notation is simpler. Let us assume that the outputs of the 256-point FFT are decimated by 8. A 256-point FFT can be written as

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi nk/N}, \quad (10.11)$$

where $N = 256$. There are 256 outputs in the frequency domain. If every eighth output is kept and the other outputs are discarded, the resulting outputs are $k = 0, 8, 16, \dots, 248$. There are a total of 32 ($256/8$) outputs. These outputs can be written as

$$\begin{aligned} X(0) &= \sum_{n=0}^{225} x(n) \\ X(8) &= \sum_{n=0}^{225} x(n) e^{-j2\pi 8n/256} \\ &\dots \\ X(248) &= \sum_{n=0}^{225} x(n) e^{-j2\pi 248n/256}. \end{aligned} \quad (10.12)$$

First, let us arbitrarily choose two frequency components $k = 16$ and $k = 248$ and rewrite in a slightly different form. The results are

$$\begin{aligned} X(16) &= \sum_{n=0}^{225} x(n) e^{-j2\pi 16n/256} = \sum_{n=0}^{225} x(n) e^{-j2\pi 2n/32} \\ &= [x(0) + x(32) + x(64) + \dots + x(224)] \\ &\quad + [x(1) + x(33) + x(65) + \dots + x(225)] e^{-j2\pi 2/32} + \dots \\ &\quad + [x(31) + x(63) + x(95) + \dots + x(255)] e^{-j2\pi 2 \times 31/32} \end{aligned} \quad (10.13)$$

and

$$\begin{aligned} X(248) &= \sum_{n=0}^{225} x(n) e^{-j2\pi 248n/256} = \sum_{n=0}^{225} x(n) e^{-j2\pi 8n/32} \\ &= [x(0) + x(32) + x(64) + \dots + x(224)] \\ &\quad + [x(1) + x(33) + x(65) + \dots + x(225)] e^{-j2\pi 8/32} + \dots \\ &\quad + [x(31) + x(63) + x(95) + \dots + x(255)] e^{-j2\pi 8 \times 31/32}. \end{aligned} \quad (10.14)$$

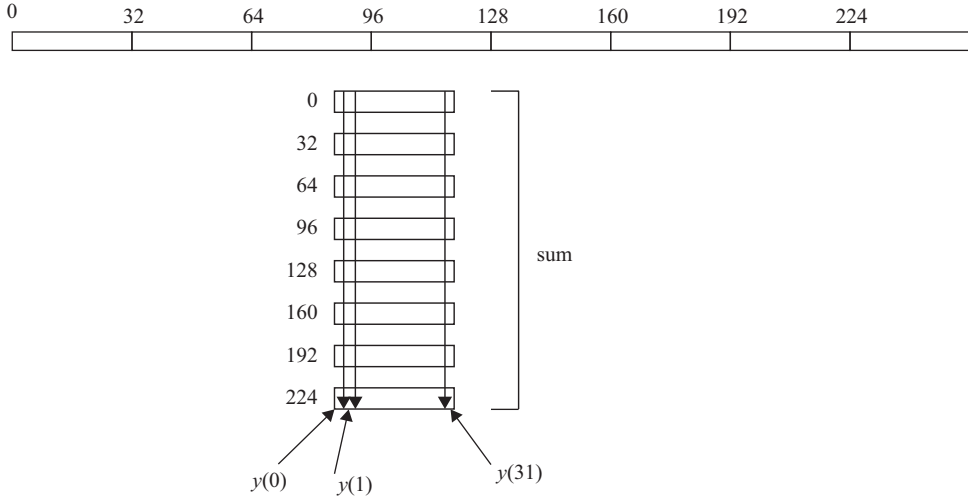


Figure 10.10 Graphic representation of obtaining the y values.

In the above equations, the relation of $e^{-j2\pi n} = 1$ when $n = \text{an integer}$ is used. Now let us define a new quantity $y(n)$ as

$$y(n) = x(n) + x(n + 32) + x(n + 64) + \cdots + x(n + 224) = \sum_{m=0}^7 x(n + 32m), \quad (10.15)$$

where $n = 0$ to 31 . This $y(n)$ represents the values in the bracket of (10.13) and (10.14). Each $y(n)$ value contains a total of 8 data points. This operation can be graphically represented in Figure 10.10. In this figure, the 256 input data points are divided into eight 32-point sections. The beginning data point of each section is shown. These eight sections are stacked and summed vertically, as shown in the figure. The results are the 32 $y(n)$ values.

Using these $y(n)$ values, the FFT results from (10.12) can be rewritten as

$$\begin{aligned} X(0) &= \sum_{n=0}^{31} y(n) \\ X(8) &= y(0) + y(1)e^{\frac{-j2\pi}{32}} + y(2)e^{\frac{-j2\pi^2}{32}} + \cdots + y(31)e^{\frac{-j2\pi 31}{32}} = \sum_{n=0}^{31} y(n)e^{\frac{-j2\pi n}{32}} \\ &\vdots \\ X(248) &= y(0) + y(1)e^{\frac{-j2\pi 31}{32}} + y(2)e^{\frac{-j2\pi 31 \times 2}{32}} + \cdots + y(31)e^{\frac{-j2\pi 31 \times 31}{32}} = \sum_{n=0}^{31} y(n)e^{\frac{-j2\pi 31 n}{32}}. \end{aligned} \quad (10.16)$$

All these equations can be written into one as

$$X(8k) = \sum_{n=0}^{31} y(n)e^{\frac{-j2\pi kn}{32}}, \quad (10.17)$$

where $k = 0, 1, 2, \dots, 31$ and $n = 0, 1, 2, \dots, 31$.

The output $X(8k)$ can be relabeled as $Y(k)$, thus the above equation can be written as

$$Y(k) = \sum_{n=0}^{31} y(n) e^{\frac{-j2\pi kn}{32}}. \quad (10.18)$$

This equation represents a 32-point FFT. In order to obtain the outputs of a 256-point FFT decimated by 8, a 32-point FFT can achieve the goal. Thus the design of the FFT can be simplified. The input signal must be manipulated, however, in order to obtain the desired result.

A general statement without further proof is presented here. If one wants to perform an N -point FFT and the outputs in the frequency domain are decimated by M , one can achieve the goal by performing an N/M -point FFT. A new input format $y(n)$ must be built first. The generalization of the $y(n)$ can be written as

$$y(n) = \sum_{m=0}^{M-1} x(n + mN/M), \quad (10.19)$$

where $n = 0, 1, 2, \dots, (N/M) - 1$. The outputs in the frequency domain can be obtained as

$$X(n) = \sum_{n=0}^{(N/M)-1} y(n) e^{\frac{-j2\pi kn}{N/M}}. \quad (10.20)$$

This section illustrates that when the FFT outputs are decimated by M , the outputs can be obtained from an N/M -point FFT.

10.9 Output Filter Shape from a Decimated FFT Widened by a Weighting Function

In this section the outputs from the FFT are discussed. If 256 data points are used for the FFT and every output is kept, there are 128 independent outputs in the frequency domain. The filters are overlapped at about -3.9 dB $\{20 \times \log[\sin(\pi/2)/(\pi/2)]\}$ down, as shown in Figure 10.4. If 32 points are used for the FFT operation and every output is kept, there are 16 independent outputs in the frequency domain. The filter shape is independent of the length of the FFT, but the bandwidth is inversely proportional to the data length used in the FFT.

If the FFT uses 256 data points but only one out of eight of the outputs are kept, there are a total of 32 outputs. Among these outputs, 16 of them carry redundant information. Therefore only 16 outputs are displayed in Figure 10.11. Each filter output is represented by a sinc function. Only a few sidelobes are shown in this figure and the highest two are only 13 dB below the mainlobe. This filter bank has many holes (high insertion loss region). If an input signal falls in one of the holes, the receiver will miss it entirely. The shape of this filter is definitely unacceptable.

To widen the individual filters and suppress the sidelobes simultaneously, a window (or weighting) function can be applied to the input data. There are many different window functions. The one used here is the Parks–McClellan window since it can provide the desired frequency response. The coefficients of the window can be obtained from the “remez” function

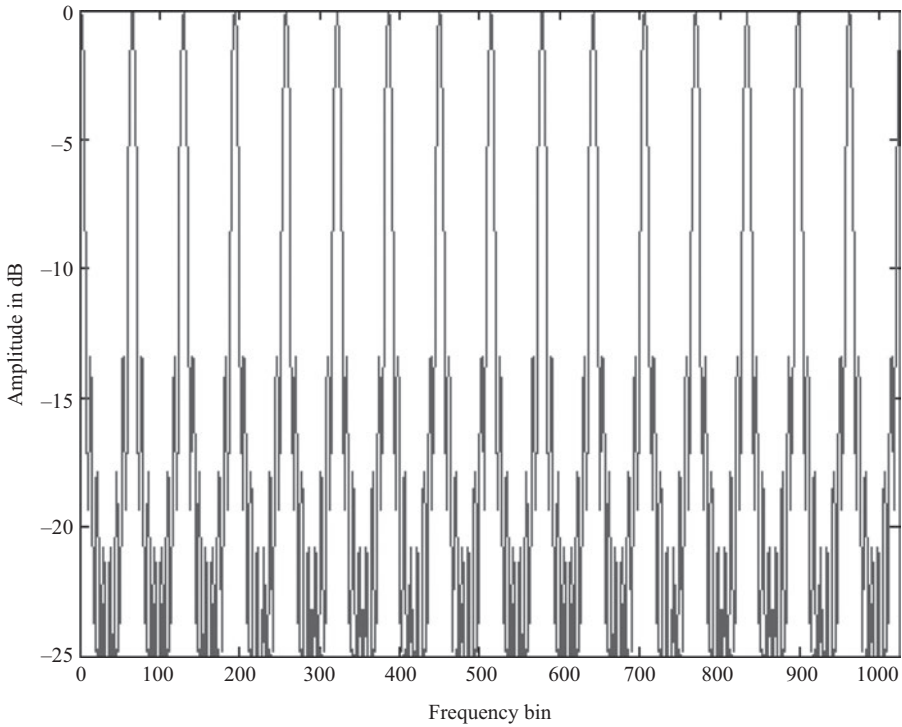


Figure 10.11 Filter outputs from decimated outputs with a rectangular window.

of MATLAB® (MathWorks, Natick, MA). The window function is shown in Figure 10.12. Figure 10.12(a) shows the time domain response as obtained from the MATLAB® “remez” program. Only the relative amplitude of the window function is of interest. Figure 10.12(b) shows the corresponding frequency domain response with very low passband ripples and sidelobes lower than 70 dB. This is a desirable filter shape. The frequency response is obtained from the “freqz” function of MATLAB®. One can see from the time domain response that the window function passes with moderate attenuation on fewer than 50 of the 256 samples. The rest of the input data are highly attenuated. The corresponding effect in the frequency domain is the wide bandwidth of each individual filter shape in the uniform filter bank.

The input data $x(n)$ are modified by the window function $h(n)$. Here, $h(n)$ instead of $w(n)$ is used for the window function because $h(n)$ is used to represent the impulse function of a filter. The resulting data $x_m(n)$ used as the input of the FFT can be written as

$$x_m(n) = x(n)h(n), \quad (10.21)$$

where $n = 0, 1, 2, \dots, 255$. As stated previously, the outputs are decimated by 8. Under this condition the modified data format in (10.15) can be used to find the $y(n)$ as

$$y(n) = \sum_{m=0}^7 x_m(n + 32m) = \sum_{m=0}^7 x(n + 32m)h(n + 32m), \quad (10.22)$$

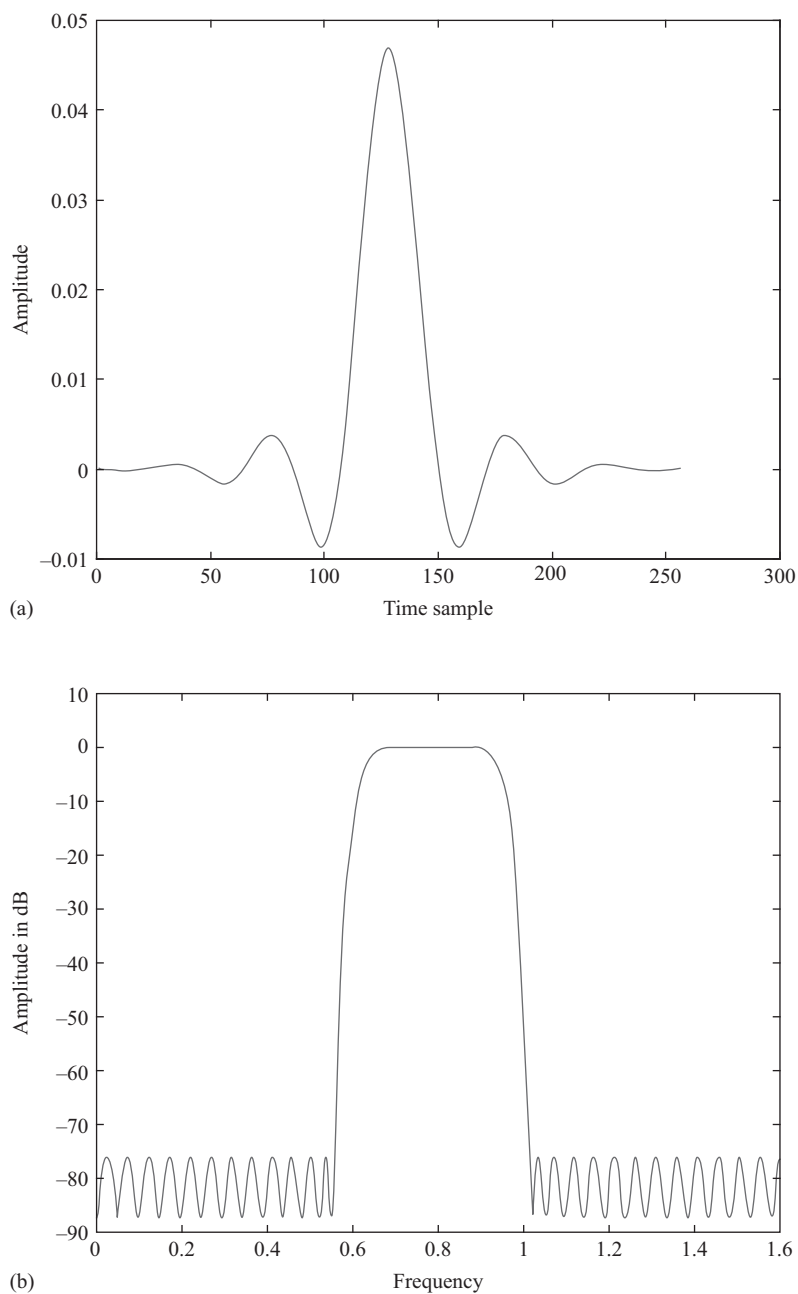


Figure 10.12 Response of a Parks–McClellan window: (a) time domain and (b) frequency domain.

where $n = 0, 1, 2, \dots, (N/M) - 1$. A few $y(n)$ terms are written as

$$\begin{aligned} y(0) &= x(0)h(0) + x(32)h(32) + \dots + x(224)h(224) \\ y(1) &= x(1)h(1) + x(33)h(33) + \dots + x(225)h(225) \\ &\dots \\ y(31) &= x(31)h(31) + x(63)h(63) + \dots + x(225)h(225). \end{aligned} \quad (10.23)$$

If a 32-point FFT is performed on these $y(n)$ values, 16 individual filters will be generated. Each filter shape is as shown in Figure 10.12(b).

The operation presented in this section can be considered as a software approach because the value in (10.22) can be calculated. With this approach it is easy to change the output sampling rate. If one would like to shift the input data by M points, all that is required is to calculate the results from (10.22) as

$$\begin{aligned} y(0) &= x(M)h(0) + x(M + 32)h(32) + \dots + x(M + 224)h(224) \\ y(1) &= x(M + 1)h(1) + x(M + 33)h(33) + \dots + x(M + 225)h(225) \\ &\dots \\ y(31) &= x(M + 31)h(31) + x(M + 63)h(63) + \dots + x(M + 225)h(225). \end{aligned} \quad (10.24)$$

In this equation the only change is the input data points, which determines the output sampling rate. If $M = 1$, the output sampling rate equals the input sampling rate, which corresponds to the 100% data overlapping case in Figure 10.2. This software approach is very flexible.

10.10 Channelization Through a Polyphase Filter and Its Operation [7–10]

Although the software approach discussed in the previous sections is very flexible, it is not suitable for high-speed operation because of the limitation of the calculation speed. However, the same operation can be accomplished in hardware with much higher operation speeds. Now let us consider in more detail the process to generate the $y(n)$ values. The $y(n)$ values listed in (10.23) must be generated from input data shifting with time. One can see that each of these values can be generated from the convolution output of a filter with the input signal. The 256-point window function in the time domain can be written as

$$h(n) = h(255)\delta(n) + h(254)\delta(n - 1) + h(253)\delta(n - 2) + \dots + h(0)\delta(n - 255), \quad (10.25)$$

where the δ function indicates the $h(n)$ value occurs at time n . The impulse sequence of the filter is written in an inverse way. This impulse function can generate the results from (10.23) through convolution with the input signal. Because the window function shown in Figure 10.12(a) is symmetric in the time domain, this inverse is just a subscript change. This function can be decimated by 32, which results in 32 individual filters, each having eight

taps. This filter decimation is often referred to as the polyphase filter. Each of the 32 filters has the response as indicated below:

$$\begin{aligned}
 h_0(n) &= h(224)\delta(n) + h(192)\delta(n-1) + h(160)\delta(n-2) + \cdots + h(0)\delta(n-7) \\
 h_1(n) &= h(225)\delta(n) + h(193)\delta(n-1) + h(161)\delta(n-2) + \cdots + h(1)\delta(n-7) \\
 &\vdots \\
 h_{31}(n) &= h(255)\delta(n) + h(223)\delta(n-1) + h(191)\delta(n-2) + \cdots + h(31)\delta(n-7).
 \end{aligned} \tag{10.26}$$

These filters must be convolved with the proper input data to obtain the result in (10.23). In order to obtain the correct data format, the input data must also be decimated by 32. When the decimated input signal and the decimated filter are convolved and reach steady state, the output is equal to the result of (10.23).

The next time the 32-point FFT is performed, the input $y(n)$ values to the FFT are

$$\begin{aligned}
 y(0) &= x(32)h(0) + x(64)h(32) + \cdots + x(256)h(224) \\
 y(1) &= x(33)h(1) + x(65)h(33) + \cdots + x(257)h(225) \\
 &\vdots \\
 y(31) &= x(63)h(31) + x(95)h(63) + \cdots + x(287)h(255).
 \end{aligned} \tag{10.27}$$

In this equation, the first data point is $x(32)$, thus the input is shifted 32 points. The hardware to accomplish this goal is shown in Figure 10.13. In this figure there are 32 filters and each filter has eight taps. Two cycles of input data are shown and each cycle contains 32 data points. The outputs are shown as $y(n)$ and they are used as the input of the FFT. The final results in the frequency domain are represented by $Y(k)$. In this case the inputs are decimated by 32, and the final frequency domain also has 32 outputs. The input data are shifted 32 points, which is also the output frequency bin number. This case is referred to as the critically sampled case. A critically sampled case is one where the number of output frequency bins equals

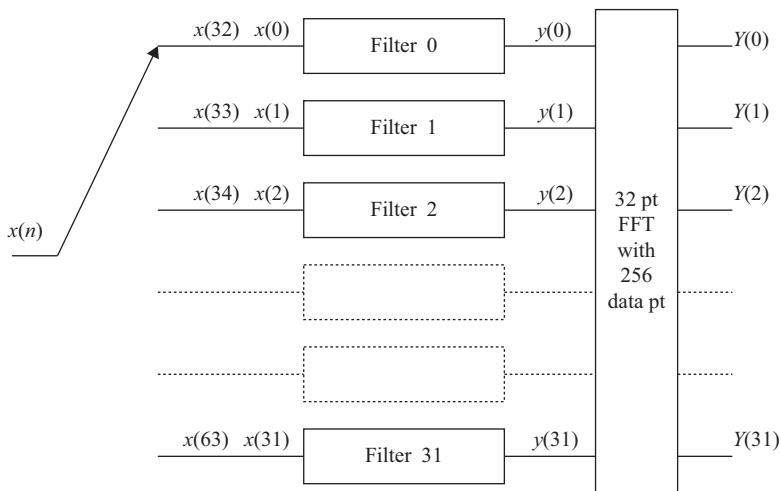


Figure 10.13 Channelization approach.

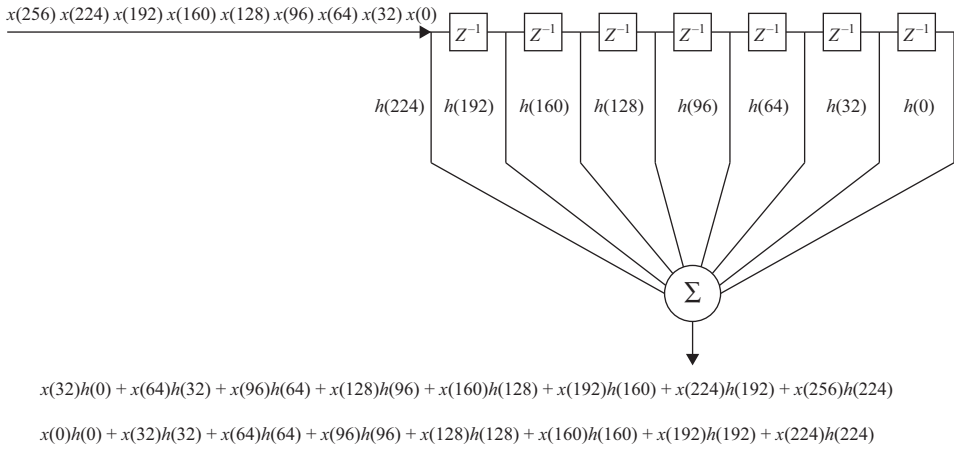


Figure 10.14 An individual polyphase filter.

to the input data shift. This means that the output sampling rate is $1/M$ times the input sampling rate, where M is the number of input data points shifted. If one wants to increase the output sampling rate, the hardware must be modified, which is not as flexible as the software approach discussed in the previous section. The detailed approach of doubling the output sampling rate with the same number of output channels is discussed in Section 10.19.

A FIR filter design is used in this discussion. Filter number 0 with $y(0)$ as output is shown in Figure 10.14. In this figure the decimated input data points are shown. When the input signal reaches steady state, the output of the filter contains eight terms. The first and second consecutive filter outputs after steady state are also listed. The lower line represents the first output, which matches the $y(0)$ output of (10.23). The upper line represents the second time output where the input data are shifted by 32 points. This result matches the $y(0)$ output from (10.27). The rest of the filter outputs can be obtained in a similar manner.

Now we will discuss the detailed operation of a polyphase filter. First, the speed of operation is considered. The input data are sampled at 3000 MHz, which is the input data coming out of the ADC at about 0.33 ns per sample. If the input data to the polyphase filter are decimated by 32, the input rate will be 93.75 MHz (3000/32) and the filter operates at this speed. In order to process all the data, there are 32 parallel channels. It is relatively easy to operate at this lower rate. Because this system has two operation rates, it is often referred to as a multirate system. The 32-point FFT following the filters also operates at this low rate.

From Figure 10.13 one can see that the input data to the filter move 32 points each time. The progression of the data processing is illustrated in Figure 10.15.

As shown in Figure 10.12(a), the input data under the mainlobe of the window function are less than 50 points. This channelization approach shifts the 256-point window only 32 points. There should be enough data overlapping that all the data contribute to the output in a near uniform manner.

Another advantage of shifting the window function 32 points is the fine time resolution. In an EW receiver, time is required to generate two parameters, as discussed in Section 2.6. One is the TOA and the other is the PW. In modern signal sorting algorithms, it is desirable to have fine time resolution. The time resolution provided by the polyphase filter is about

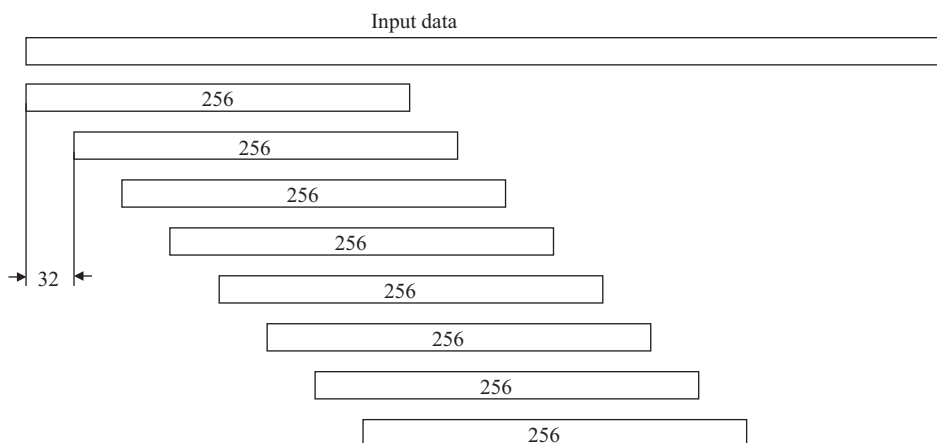


Figure 10.15 Input data rate.

10.7 ns, which is suitable for most of these operations. This output rate is also influenced by the signal processing following the filter, which will be further discussed in later sections.

10.11 Processing After Channelization

We described how the input signals are separated in the frequency domain through channelization. After the frequency channelization, further processing is performed to determine the number of frequencies and the frequencies themselves. The frequency data resolution (or frequency bin width) calculated through an FFT operation determines the frequency precision measured on the input signal. It is often desirable to obtain better frequency precision than the FFT operation can provide. This is especially true for the 32-point FFT described in our filter bank example, which produces a frequency data resolution of 93.75 MHz with a 3-GHz sampling frequency. Another problem with such a wide frequency bin is that when two signals fall into one frequency output channel, the receiver cannot effectively separate them. In other words, the two-signal frequency resolution is only about 93.75 MHz.

The traditional way to separate signals by frequency is through an analog filter bank. Although the main emphasis of this book is digital, one still can consider the analog filtering approach. In an analog channelized receiver, after the filter bank and amplifier, crystal video detectors are used to convert the radio frequency (RF) into video signals. The video signals are digitized by ADCs and are further processed to determine the number of signals and their frequencies. In converting RF into video signals, some information is lost. If two signals fall into one channel, it is difficult to separate them.

In a digital receiver, ADCs can be used after an analog filter bank to obtain digitized RF information. This information can be further processed; for example, the monobit receiver idea can be used to separate two signals in one channel.

Finer frequency estimation and the capability of processing two signals close in frequency are the two main goals of processing after channelization. To solve these two problems, the concepts of the instantaneous frequency measurement (IFM) receiver and the

monobit receiver can be used after channelization. The IFM receiver concept is discussed in Chapter 8 as the phase measurement method. Both the analog filter bank and digital channelization are considered.

10.12 Basic Considerations of the Channelized Approach

A receiver is often designed to match the minimum PW the receiver is expected to process. If the minimum desirable PW is 100 ns, the general rule is to select a filter bandwidth of 10 MHz (1/100 ns), which can be referred to as the minimum required filter bandwidth. A narrower filter will degrade the signal-to-noise (S/N) ratio and disturb the PW measurement because the transient effect of the filter may last longer than the minimum PW. Under this condition, a pulse with the minimum PW is actually extended by the transient effect of the filter. However, PW is not a very reliable parameter in EW receivers because of the multipath problem. Multipath means that a signal reaches the receiver from many different paths. The direct path is the signal reaching the receiver directly. The signal can also reach the receiver by reflecting from an object. The direct signal and reflected signals may interfere and change the PW.

Another more serious problem with the transient effect is that during the transient period the output frequency will shift toward the center of the filter. The transient effect is the filter response to a step function, which causes a damped oscillation at the center frequency of the filter. If the transient period is longer than the PW, the output from the filter will not contain the input frequency, especially when the signal is near the edge of the filter. Once this phenomenon occurs, the frequency encoding circuit following the filter may generate erroneous frequency information.

A short pulse passing through a filter bank can generate outputs in many adjacent filters because the leading and trailing edges appear as a step function to every filter. It is not only difficult to determine the center frequency of the signal, but it is also difficult to determine the number of input signals. From previous receiver design experience, the filter bandwidth is usually chosen to be much wider than the value obtained from the reciprocal of the minimum PW. Sometimes the selected bandwidth may be five or more times wider than this value.

There is a significant difference between designing a single-channel narrowband receiver and a wideband receiver with narrowband channels. In the narrowband receiver, the signal can be tuned to the center of the filter by changing the frequency of the local oscillator. Once the signal is shifted to the center of the filter, the transient effect will be minimized. In a wideband channelized receiver, the filters are fixed in frequency, as are the frequencies of the local oscillators. A signal can fall in the center of a filter as well as in between two channels. When there is more than one signal in one channel, the problem becomes more complicated. Figure 10.16 shows such a problem. In this figure, three adjacent filters A, B, and C are shown with two input signals. Both signals are at the edge of filter B. Because of the finite slope of the filters, these two signals will be processed by all three channels. Theoretically one would like to determine the frequency of signal 1 by channels A and B and signal 2 by channels B and C. However, determining the number of signals under this condition can be difficult. That is why in many receiver designs the minimum separation in frequency is specified wider than the bandwidth of the filters in order to avoid the need to separate signals in the same channel. It is highly desirable to design a receiver that can measure two signals falling into one channel, especially for wideband channels.

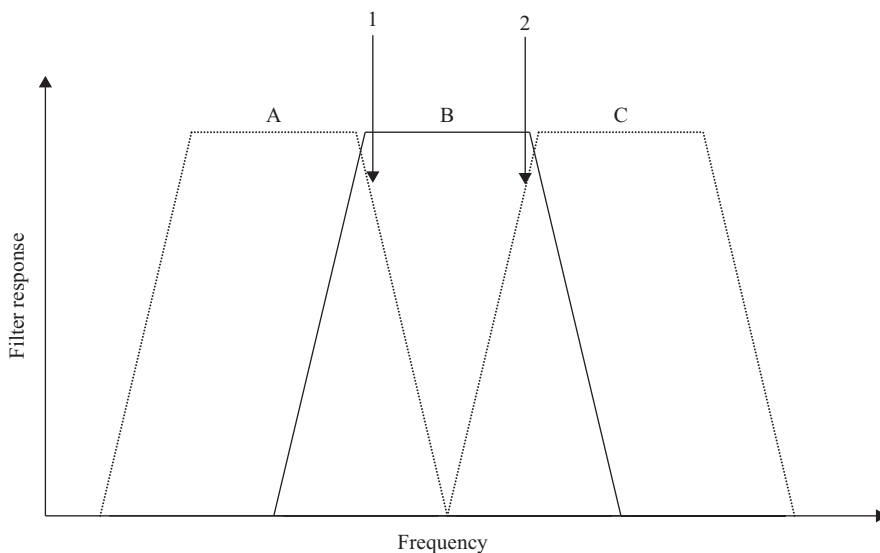


Figure 10.16 Two-signal conditions of a filter bank.

10.13 Filter Shape Selection [11, 12]

Selection of the filter shape is one of the major tasks in designing a channelized receiver. First, let us consider the bandwidth of the filters. Figure 10.17 shows the shapes of three different filters. In these figures, two bandwidths are shown. One is the 3-dB bandwidth, which equals the total bandwidth of the receiver divided by the number of channels; the other is the 60-dB bandwidth. The name of the 60-dB bandwidth is arbitrarily chosen because using a numerical value is easier to refer to in later discussions. Using this definition, the maximum instantaneous dynamic of the receiver is limited to 60 dB, which is a very large value for a wideband receiver. In actual receiver design, this number depends on the filter selected.

In Figure 10.17(a) the 60-dB bandwidth is double the 3-dB bandwidth. With this arrangement a signal will fall into two filters most of the time, except right at the center of a channel, where it falls into only one filter. This can be referred to as the maximum 60-dB bandwidth allowed. If the skirt of the filter is wider than this value, as shown in Figure 10.17(b), one signal can fall into three channels most of the time. This is highly undesirable because one needs to compare the outputs of three channels to determine a signal.

It is desirable to keep the probability of one signal falling into two adjacent channels small, as shown in Figure 10.17(c). If a signal falls into one channel, the logic circuit of that channel will process the signal. If a signal falls into two adjacent channels, both circuits of the adjacent channels will process the signal. It is easier to measure one signal with one encoder circuit. If two channels measure one signal, the results must be compared to determine whether there are one or two signals.

This last requirement in Figure 10.17(c) needs filters with a very sharp skirt. A sharp filter will have a relatively long transient time when the leading and trailing edges of the

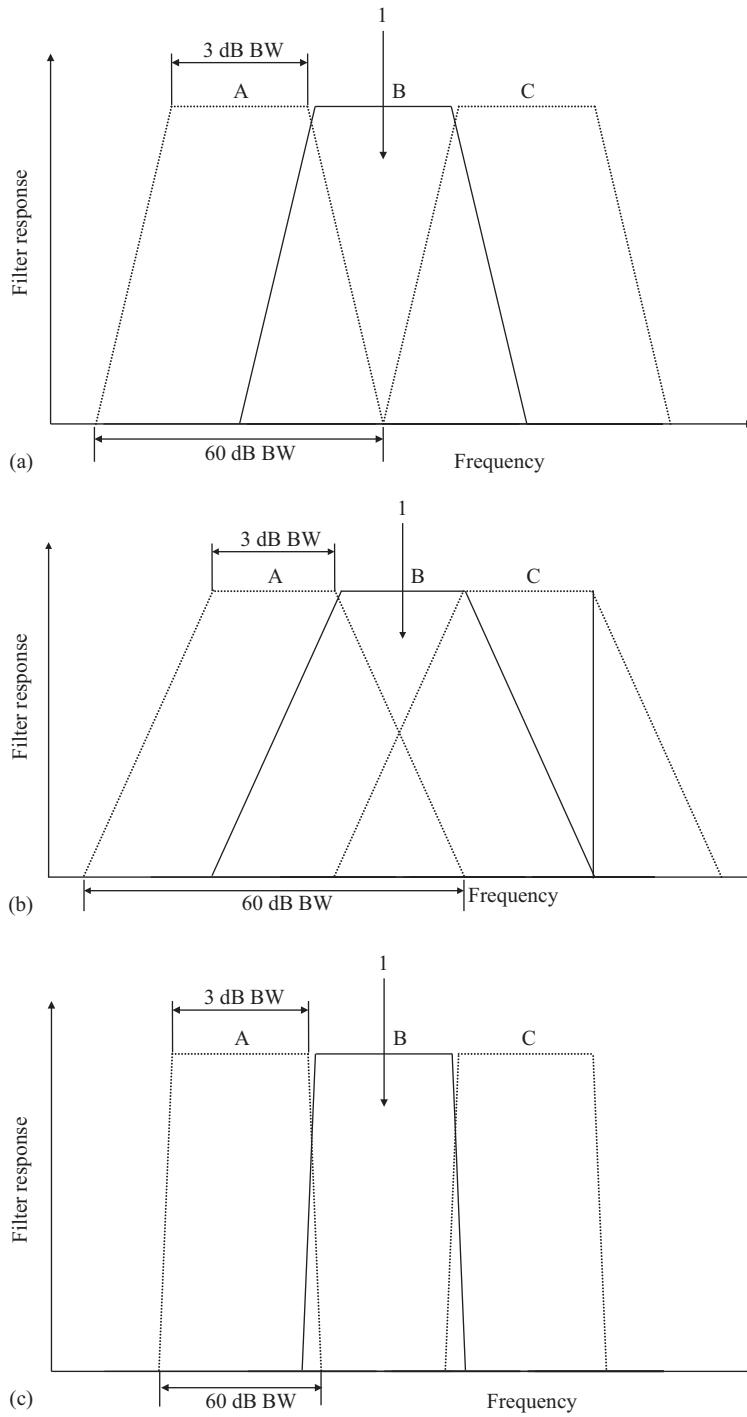


Figure 10.17 Filter shapes: (a) filter with limiting skirt, (b) filter with wider skirt, and (c) filter with narrower skirt.

signal pass through it. During the transient period both the amplitude and the frequency of the output signal change, as mentioned before. If the duration of the transient time is comparable to the minimum PW, the steady state of the pulse might be too short to provide an estimation of the signal frequency. The long transient time eliminates the selection of very sharp filters. The transient effect must be evaluated against the desired minimum PW in selecting filters.

Another issue to be considered is the bandwidth of the processing circuit following the filter. Usually the frequency response of the processing circuit is repetitive, such as in an FFT operation where the unique frequency is from 0 to $f_s/2$. As discussed in Section 3.9, when the signal is near the edge of the band, it can be assigned to a wrong frequency. If the 3-dB band is used for further processing, there is usually an ambiguity at the edge of the circuit because of the presence of noise in the signal. This effect can put a signal on the wrong side of the filter and cause a frequency error equal to the bandwidth of the filter. In order to avoid this problem, the minimum bandwidth of the processing circuit should be equal to the 60-dB bandwidth rather than the 3-dB one. With this bandwidth, if a signal falls in between channels, the processing bandwidth is wide enough to avoid the ambiguity.

The channelized receiver considered earlier is used as an example to demonstrate filter selection. Assume that the window function is the same as the one displayed in Figure 10.12(a). The approach to obtain this filter is as follows.

The sampling frequency is 3000 MHz, thus the unambiguous bandwidth is 1500 MHz. Because only 16 independent channels can be obtained from the 32-point FFT, the equivalent filter bank has 16 outputs. The bandwidth of each channel is 93.75 MHz ($1500/16$), which can be considered as the 3-dB bandwidth. It is desirable to have at least 60-dB attenuation at a bandwidth of 187.5 MHz (93.75×2), which is double the 3-dB bandwidth. This required filter shape is shown in Figure 10.18. Only three adjacent filters are shown.

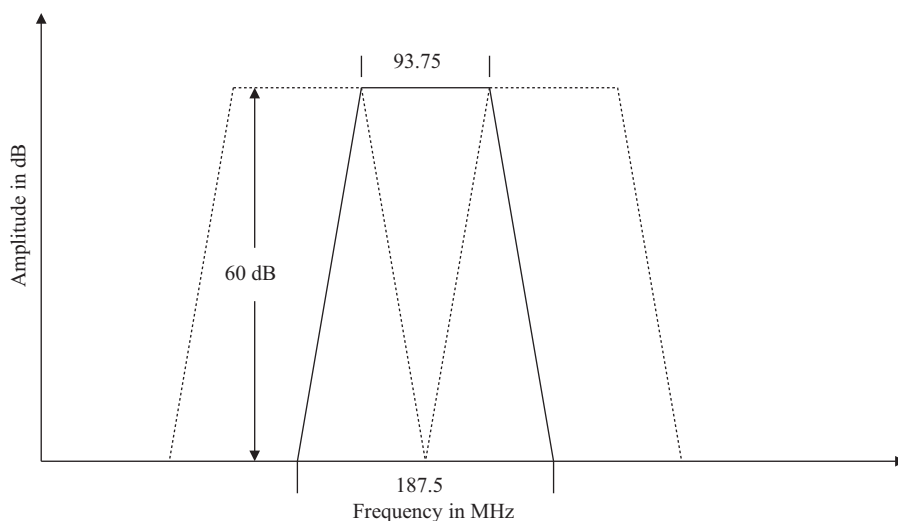


Figure 10.18 Desired filter response.

In order to realize this filter response, one must determine how long the window function (or filter taps) should be. This can be determined from [11]

$$T_p = \frac{-10 \log(R_p R_s) - 13}{2.324 B_{tr}} + 1, \quad (10.28)$$

where R_p and R_s are related to the passband ripple factor and the insertion loss of the stopband, respectively, and B_{tr} represents the transition period in radians, which is $2\pi/64$. The passband ripple in decibels is related to R_p as $20 \log(1 + R_p)$ and the stopband insertion loss is related to R_s as $20 \log(R_s)$. With $R_p = 0.01$ [R_p (dB) = 0.086 dB] and $R_s = 0.001$ [R_s (dB) = -60 dB], $T_p = 163$. Because the total window function contains 256 points, the required filter response should be achieved or exceeded.

Using the MATLAB[®] program and adjusting some of the parameters, such as the 3-dB bandwidth and the 60-dB bandwidth, one can select a window function to result in the filter bank shown in Figure 10.19. This filter bank has a dynamic range of about 75 dB, which is higher than the design goal of 60 dB because of the high number of taps. The filter shape is quite uniform. This is superior to the performance of an analog filter bank. Because the 3-dB bandwidth of the filter is about 93.75 MHz, its capability to separate two signals close in frequency is quite limited. If two signals fall into one filter, the receiver cannot separate them without further signal processing. The performance of the MATLAB[®] example shown here is quite close to what can be done in an actual filter implementation in the laboratory.

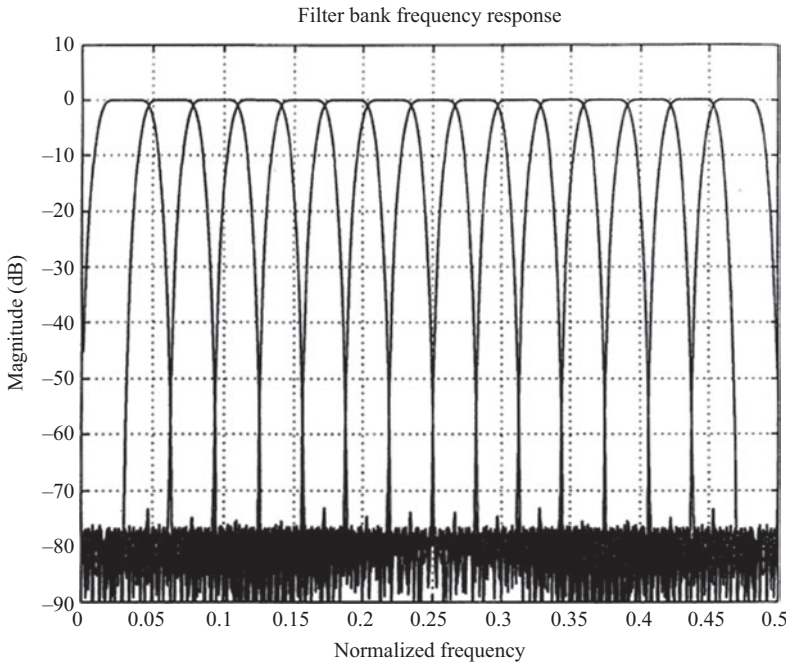


Figure 10.19 Designed filter bank.

10.14 Analog Filters Followed by Phase Comparators

The performance of the phase comparison scheme (or the IFM receiver in the analog scheme) is discussed in Chapter 8. It can improve frequency measurement accuracy. It cannot, however, separate two signals easily, although theoretically the two signals can be resolved as discussed in Sections 8.6 and 8.7. It appears that an analog filter bank followed by narrowband IFM receivers can produce satisfactory results. This idea is often referred to as the channelized IFM receiver, which has not yet been successfully demonstrated because of problems in the frequency encoder design. The main problem is to determine the frequency of one signal falling between two channels as well as the frequencies of two signals in adjacent channels. This is an analog approach and will not be discussed further. An analog filter bank followed by narrowband phase comparators should produce similar results.

Figure 10.20 shows such an arrangement. The input signal passes through a filter bank. Each output is divided into two paths through a 90-deg hybrid since the complex outputs with in-phase and quadrant (I and Q) outputs are needed to obtain the phase of the input signal.

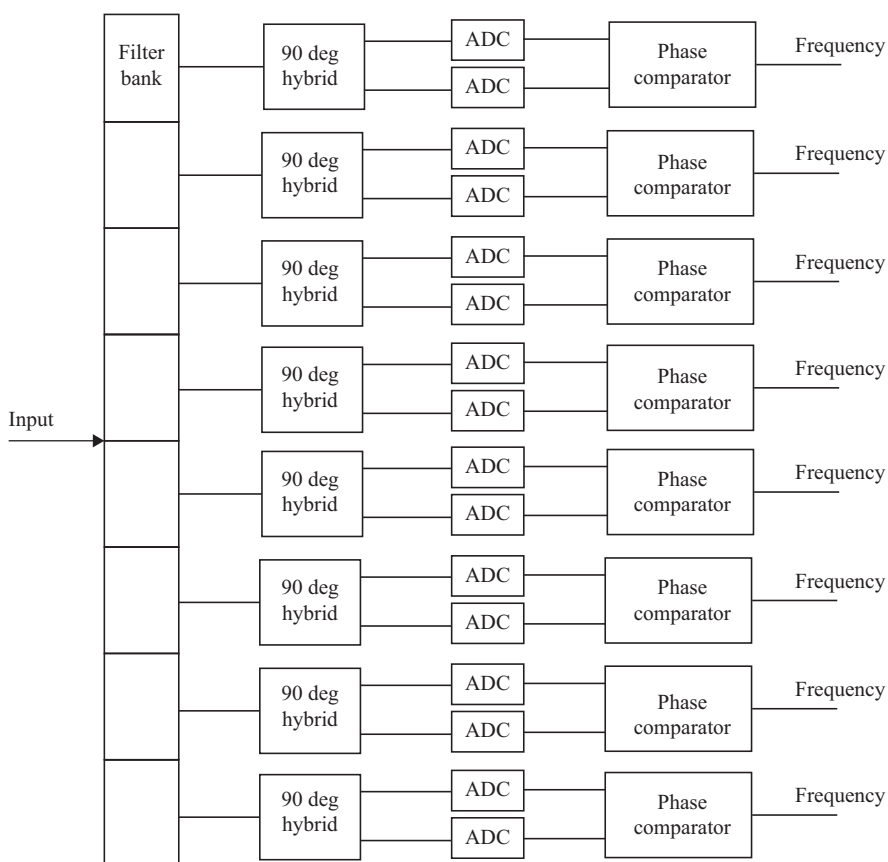


Figure 10.20 A filter bank followed by phase comparators.

Let us use an example to illustrate the phase comparison configuration. Assume that the input bandwidth is 1 GHz and is divided into eight parallel channels. Each channel has a 3-dB bandwidth of 125 MHz (1000/8). Let us also assume that the 60-dB bandwidth is 250 MHz, which is the maximum allowed bandwidth as discussed in the previous section. Because the inputs to the phase comparator are complex (I and Q channels), the sampling frequency can be 250 MHz to cover the 250-MHz bandwidth. Theoretically this approach can process two input signals per channel, as mentioned in Section 8.7. If the I and Q channels are not perfectly balanced, the measured response of a single signal can appear as two simultaneous signals. This is one of the fundamental limitations of the phase measurement method.

The concept of using phase comparators after analog filters has not been attempted experimentally, perhaps because of the hardware complexity of the implementation.

10.15 Monobit Receivers Followed by Phase Comparators

The wide monobit receiver discussed in Chapter 9 has a relatively narrow output band, which is about 9.77 MHz (2500/256). A signal in a certain channel will have the same frequency reading whether it is at the center or at the edge of the channel. Thus a signal at the center of the channel can get a good frequency reading but a signal on the edge of the channel can have a frequency error that is close to half a channel of 4.89 MHz (9.77/2). This error can be considered as the frequency digitization error. It is desirable to have a more accurate frequency reading than this value. As discussed in Chapter 8, the phase comparison method can provide a much better single-frequency precision than the FFT outputs. Thus it is desirable to have phase comparators after the monobit receiver.

The phase comparator can be achieved through the following approach. If the highest amplitude from the FFT with a 4-value kernel function is $|X(k)|$, the phase of the output can be calculated as

$$\theta_n(k) = \tan^{-1} \left(\frac{\text{Re}[X_n(k)]}{\text{Im}[X_n(k)]} \right), \quad (10.29)$$

where $\theta_n(k)$ represents the phase of frequency component k at time n , and Re and Im represent the real and imaginary parts of $X_n(k)$. If the phase measured at time $n + 1$ is

$$\theta_{n+1}(k) = \tan^{-1} \left(\frac{\text{Re}[X_{n+1}(k)]}{\text{Im}[X_{n+1}(k)]} \right), \quad (10.30)$$

the fine frequency can be found as

$$f = \frac{\theta_{n+1}(k) - \theta_n(k)}{2\pi t_o}, \quad (10.31)$$

where t_o is the output sampling time. The single frequency precision improvement is discussed in Section 8.4.

The filter bandwidth of 9.77 MHz can be considered as the 3-dB bandwidth, which is also the frequency data resolution of the monobit receiver. Two signals separated by less than

9.77 MHz in frequency are difficult to separate by the phase comparators. Because this bandwidth is relatively narrow, the probability of this situation occurring is low and will not be considered here.

The outputs from the monobit receiver are complex, which is equivalent to the I and Q channels, therefore the phase can be obtained directly from these outputs. The output sampling time is 102.4 ns, as discussed in Section 9.4, and the corresponding sampling rate is 9.77 MHz ($1/102.4$ ns). Because the outputs are complex, the equivalent bandwidth is also 9.77 MHz, which is the same as for the 3-dB bandwidth. Using (10.29), (10.30), and (10.31) to calculate fine frequency, the phase calculated will have 2π ambiguity. The 2π phase range covers the frequency bandwidth of 9.77 MHz. If a signal falls between two adjacent channels, it also falls on the edge of the phase comparator. The noise in the receiver may determine that the signal is 9.77 MHz away because of the 2π ambiguity. This effect may cause large error in the frequency reading.

In order to eliminate the large error, two approaches can be used. The first approach is to compare the amplitudes of two neighboring channels of the output channel. Figure 10.21 shows such an arrangement. In this figure, one signal is between two channels, B and C. Let us assume that the output from channel B is higher than that from channel C. Therefore the output from channel B is used to find the fine frequency. Because of the noise in the input data, it is possible to put the fine frequency at position 2, which is the wrong frequency. In this approach, the amplitudes of channels A and C (the neighboring channels of B) are compared. Because the signal is between B and C, the output from channel C should be higher than from channel A. This condition can be used to determine that the frequency of the input signal should be between B and C rather than A and B.

The phase comparators are used after the monobit receiver to improve the frequency accuracy of the input signal. Although the monobit receiver has only 2 bits as input, the FFT

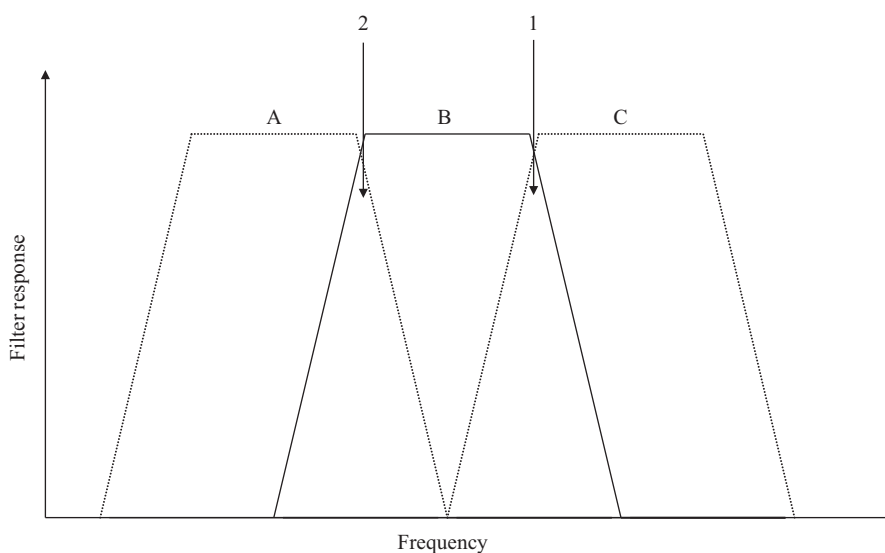


Figure 10.21 One signal falls between two channels.

outputs have 8 bits (including one sign bit). Thus the phase value can be calculated rather accurately. A computer is used to calculate the results of (10.29) through (10.31). If the phase calculated is near the edge of a phase comparator, the amplitudes of the two neighboring channels are compared to determine the direction of the frequency shift. A frequency error of 0.5 MHz can be achieved without large error. Thus this method improves the single-frequency precision by about 10 times. In order to use this method, two consecutive outputs from the monobit receiver must be used, therefore the minimum PW must be more than 204.8 ns.

The second approach to eliminate the 2π ambiguity is to increase the output sampling rate. One can double the output sampling rate of the monobit receiver to 19.53 MHz, which corresponds to a sampling time of 51.2 ns. With this arrangement, a signal falling between two adjacent channels will not fall on the edge of the phase comparator. Thus the 2π ambiguity problem at the edge of a phase comparator will not occur. This implies that the monobit receiver output rate will be increased to 19.53 MHz, which will require doubling the monobit receiver clock rate. This approach will increase the complexity of the design significantly. One advantage of this approach is that the minimum PW is still at 102.4 ns since this method uses two cycles of 51.2-ns outputs.

10.16 Digital Filters Followed by Phase Comparators [13]

The digital filter obtained from the FFT operation as discussed in Section 10.13 is similar to the wideband monobit receiver (e.g., the outputs are also complex). The major difference is the bandwidth of the filter. If the sampling rate is 3 GHz and a 32-point FFT is performed with a window function, the resulting filter bandwidth is about 93.75 MHz ($3000/32$). With this wideband filter the frequency data resolution is 93.75 MHz, which is too coarse for signal sorting applications. The minimum frequency precision allowed is about 10 MHz. A better frequency precision is always desirable as long as the minimum PW stays the same, therefore it is essential to improve this frequency precision. The approach is identical to the method discussed in the previous section, which can improve the frequency precision many times. In these wideband filters, simultaneous signals should also be considered since the probability of two signals falling into one channel is high.

Let us use the digital filter bank discussed in Section 10.13 as an example. The filter bank shape is shown in Figure 10.19. If there are two signals and both are in channel 5, signal 1 is close to channel 4 and signal 2 is close to channel 6. The signals are separated by $1/4$ of the channel width. Signal 1 is 20 dB stronger than signal 2. In this case the outputs from all three channels—4, 5, and 6—are used to calculate the phase. The corresponding frequencies from the three channels are shown in Figure 10.22. Channel 4 shows the frequency of signal 1 and channel 6 shows the frequency of signal 2. Channel 5 has both signals and the output has ripple on it. Because signal 1 is stronger than signal 2, the average frequency measured in this channel is the frequency of signal 1, as discussed in Section 8.7. Comparing frequencies in channels 4 and 5, one can see this effect. The ripple indicates that two simultaneous signals are present in the channel. The frequency of the ripple represents the difference frequency between the two signals. In this design, channels 4 and 6 can measure the two frequencies, making it unnecessary to obtain the frequency from channel 5.

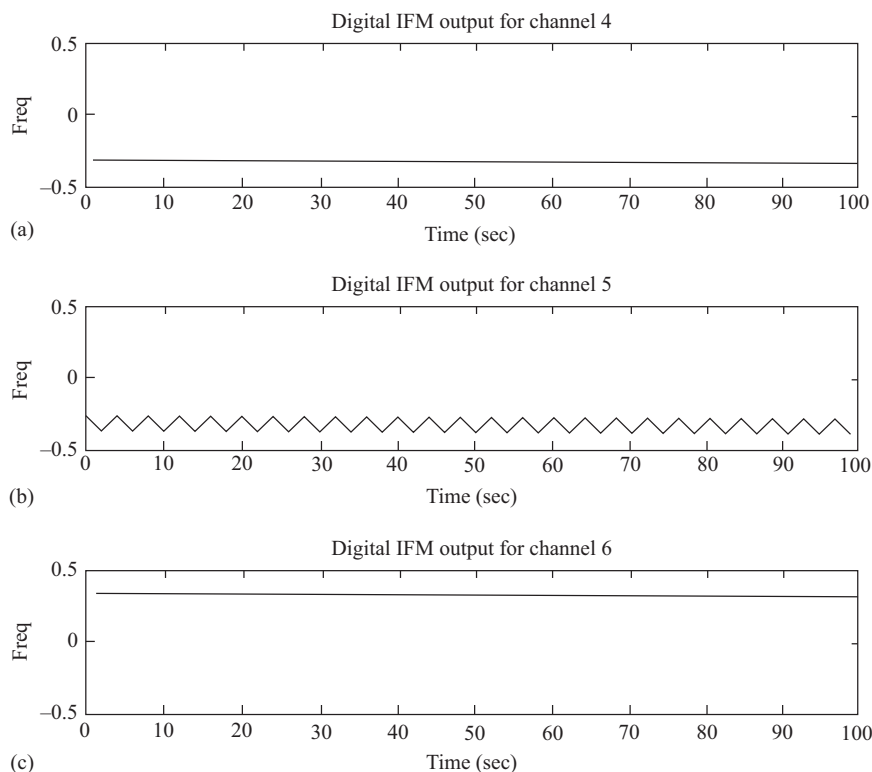


Figure 10.22 Frequency outputs from channels (a) 4, (b) 5, and (c) 6.

10.17 Analog Filters Followed by Monobit Receivers [14]

There are two major deficiencies in the wideband monobit receiver discussed in Chapter 9. The first one is the low instantaneous dynamic range, which indicates that the receiver cannot process two signals separated more than 5 dB in amplitude. The receiver usually misses the weaker signal when two simultaneous signals are present. The second deficiency is that the receiver can process only two simultaneous signals in order to keep the design simple.

In order to remedy these deficiencies, a channelized approach can be used, as shown in Figure 10.23. The channelization can be accomplished using analog filters, which will separate the signals into different channels according to the input signal frequency. In each channel an amplifier can be used to increase the input signal level. These amplifiers can be limiting amplifiers as discussed in Section 9.6. Narrowband monobit receivers can be used at the output of the amplifiers. Each narrowband monobit receiver can process two simultaneous signals. Theoretically this approach can process 16 simultaneous signals. In each channel the instantaneous dynamic range is limited to about 5 dB. Because the filters limit out-of-band signals, the instantaneous dynamic range of the receiver depends on the analog filter shape.

Because the narrowband filters limit the input band to the ADC, each ADC only covers a portion of the receiver input band. Using the example in Section 10.14 and Figure 10.23, the

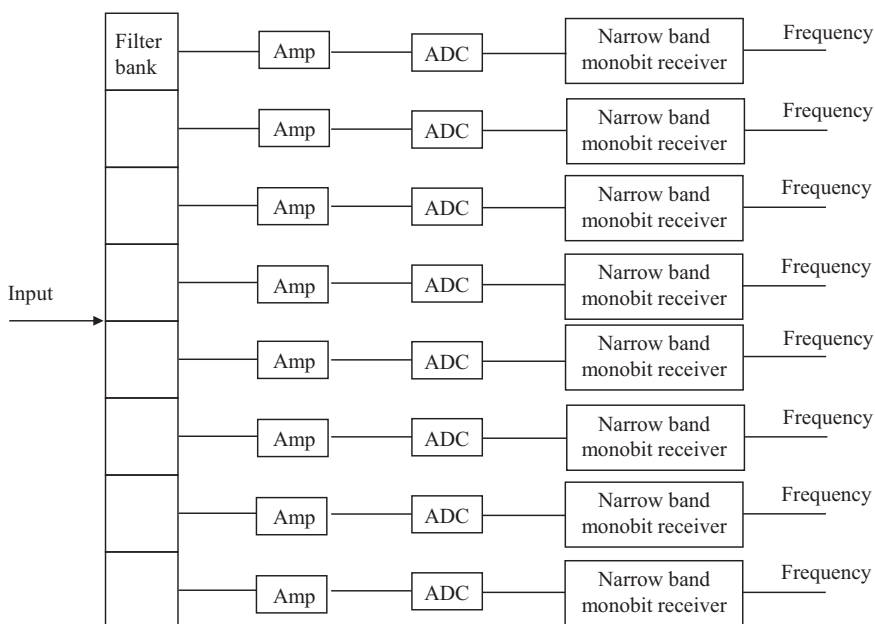


Figure 10.23 Analog filter bank followed by monobit receivers.

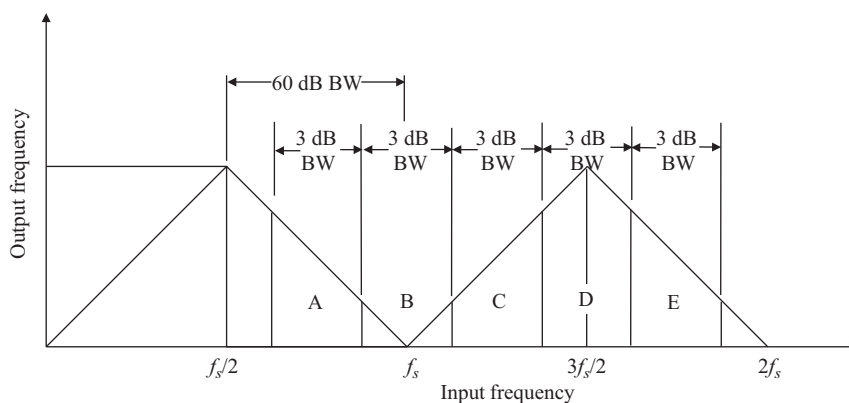


Figure 10.24 Adjacent channels aliased into the baseband.

3-dB bandwidth is 125 MHz ($1000/8$). The 60-dB bandwidth is about 250 MHz. In order to eliminate ambiguity between channels, each channel should cover 250 MHz. The Nyquist sampling criterion requires the ADC to operate at two times the bandwidth for real signals, thus the ADC should operate at 500 MHz (2×250), which is much slower than the 2.5-GHz sampling rate used in the wideband monobit receiver. Operation at this sampling rate can cause a band overlapping problem, as shown in Figure 10.24. In this figure, let us assume that the 60-dB bandwidth matches the second alias region. The 3-dB bandwidth of channels A, B, and C are also shown. Channels A and C are at the centers of alias regions $f_s/2$ to f_s and

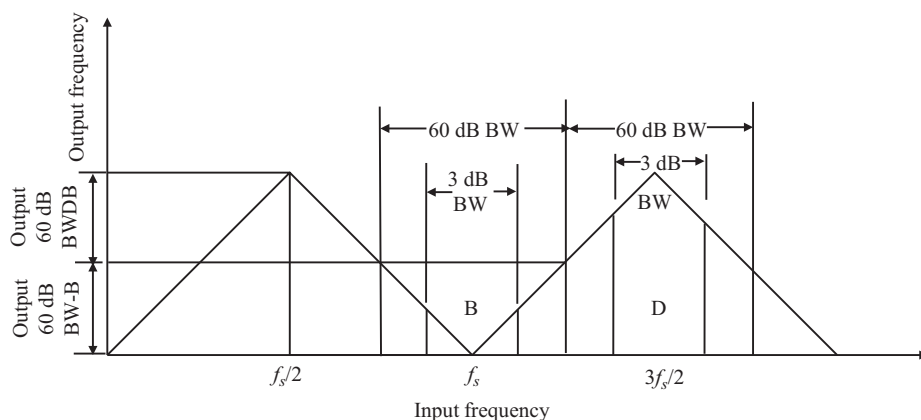


Figure 10.25 Band overlapping caused by aliasing.

f_s to $3f_s/2$, respectively. Under this condition, the output bandwidth equals the input bandwidth and there is no ambiguity problem. The center of channels B and D are at f_s and $3f_s/2$, respectively.

Figure 10.25 shows the bandwidth folding of channels B and D. It shows that the output bandwidth of channels B and D is only half of the input bandwidth for both the 3-dB and 60-dB cases. Therefore a signal on either side of f_s in channel B can be aliased to the same output frequency and cause ambiguity in the frequency reading. The same situation happens when an input signal is on either side of $3f_s/2$ in channel D.

One can see from Figure 10.24 that channels A, C, E, and G do not have this problem, although channel G is not shown. The other four channels—B, D, F, and H—have the band overlapping problem, although channels F and H are not shown. One way to eliminate this problem is to redesign the front end of the receiver. The channelized monobit can only accept input in the A, C, E, and G bands. The input frequency range of an EW receiver system is basically from 2 to 18 GHz; the EW receiver discussed in this book is a baseband receiver, which covers a bandwidth of about 1 GHz, as discussed in Section 2.5. The input signal must be converted to the input of the baseband receiver. One way to convert the input is shown in Figure 10.26. The input signal is channelized into eight consecutive frequency bands. The outputs from the odd channels 1, 3, 5, and 7 are converted to the frequency bands A, C, E, and G. The outputs from the even channels 2, 4, 6, and 8 are also converted to the frequency bands A, C, E, and G. With this arrangement there is no band overlapping.

Because the sampling frequency is 500 MHz, 64 samples will last 128 ns, which is slightly longer than the desired minimum PW of 100 ns. In this design the FFT is 64 points long and the input data rate is 500 MHz. If eight narrowband monobit receiver chips are used to build the receiver, each chip is rather simple. It is possible to put all eight monobit receivers on one chip. Simulated results using the front end in Figure 10.26 indicate that the receiver can process several simultaneous signals with high instantaneous dynamic range.

One possible approach [14] to enhance the outputs from the monobit receiver outputs is to add together amplitude outputs from adjacent channels, as shown in Figure 10.27. In this figure the outputs from two adjacent channels are shown. The bandwidth of the monobit receiver is equal to the 60-dB bandwidth of the filter. If one signal falls between two

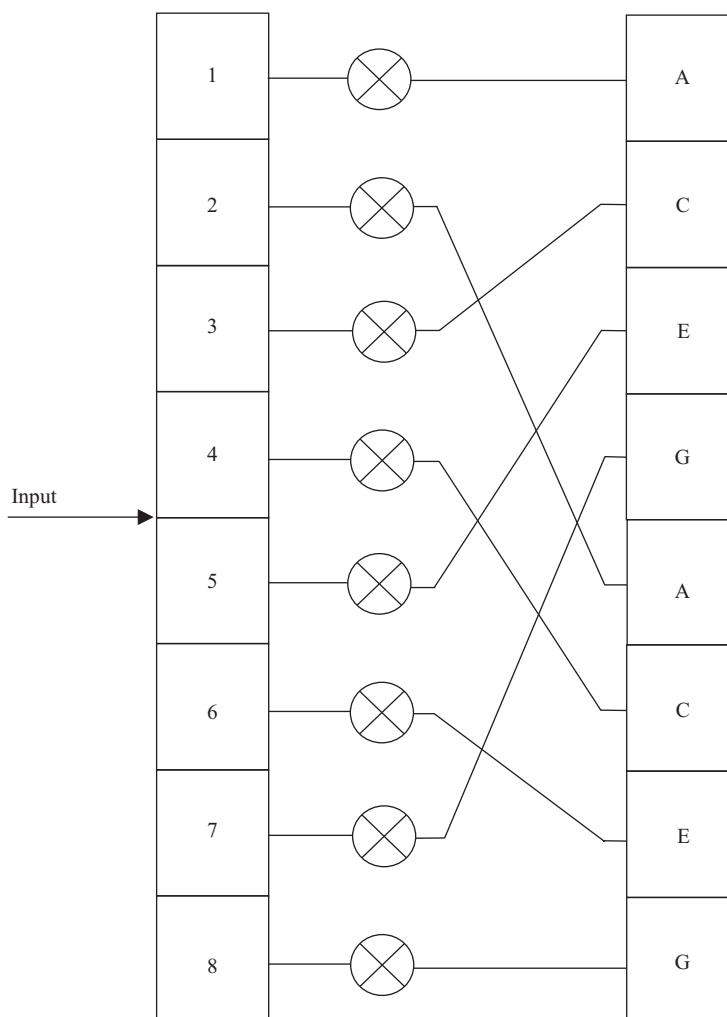


Figure 10.26 Front-end arrangement of an analog channelized monobit receiver.

channels, it will be read by both monobit receivers A and B. If the amplitudes of the outputs of these two monobit receivers are summed together, the strength of the signal should be enhanced. This approach will improve the detection sensitivity.

10.18 Considerations of Digital Filters Followed by Monobit Receivers

Digital filters usually have a wide bandwidth, which is limited by the operating speed of the FFT chip. A 32-point windowed FFT with a sampling frequency of 3 GHz will have a channel bandwidth of 93.75 MHz ($3000/32$), which is also the output sampling rate and

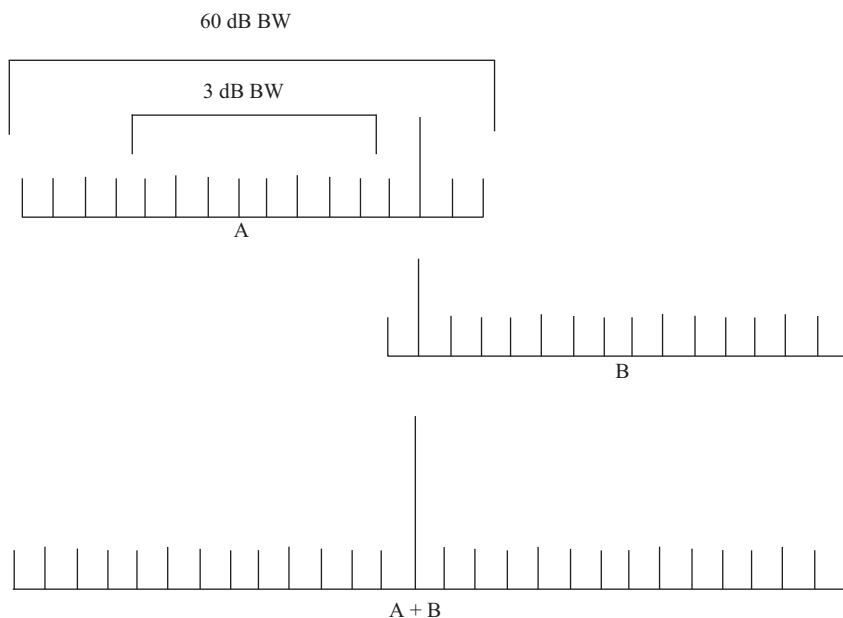


Figure 10.27 Summing outputs from two adjacent monobit receivers.

the FFT operation rate. With this bandwidth there is a greater chance of two simultaneous signals falling in the same channel. It is desirable to separate two signals in one channel, and a narrow monobit receiver can fulfill this requirement. As discussed in Section 10.15, two signals in one channel can be measured by the two neighboring channels. The basic operation is still one phase comparator measuring one signal. In this arrangement, two adjacent channels always process one signal. If the filter shape is chosen as shown in Figure 10.17(c), it is most likely that one channel processes only one signal. With this type of arrangement it is desirable to separate two signals in one channel since a phase comparator in one channel cannot separate two signals effectively.

The basic idea of using a narrowband monobit receiver after a digital filter bank is to perform an FFT operation on the output of the FFT. The monobit receivers used at the outputs of the FFT can be considered as part of the encoding circuit. One important factor in building this receiver is the bandwidth of the monobit receiver. In order to process signals falling between channels, the monobit receiver should have a bandwidth wider than the 3-dB bandwidth. It is desirable to have the bandwidth equal to the 60-dB bandwidth. If the channelization is accomplished in software, changing the output sampling rate is rather simple (Section 10.9). The input data can be shifted by any desired value through software operation. Although the software approach is flexible, it is limited to low-frequency operation. It is impractical to operate on input data digitized at 3 GHz. The channelization can only be achieved in hardware for a wideband digital receiver. The output sampling rate must be increased to match the desired bandwidth required for the narrowband receivers. The following section discusses increasing the output sampling rate in hardware.

10.19 Increase the Output Sampling Rate by Two [8, 15–17]

A general discussion on increasing the output sampling rate can be found in Crochiere and Rabiner [15]. We have already discussed the critical sampling rate. Under this critical sampling condition, the relation between the output frequency bin number and the output sampling rate is

$$K = M, \quad (10.32)$$

where K is the number of output frequency bins and M is the number of data points shifting per FFT operation, which is related to the output sampling rate. Set $K = M = 32$. The input sampling rate is 3000 MHz and the output sampling rate is 93.75 MHz (3000/32). Because the FFT outputs are complex, with a 93.75-MHz sampling rate the bandwidth is also 93.75 MHz, which is equal to the 3-dB bandwidth of the filter. As discussed in Section 10.13, the bandwidth of the monobit receiver should be equal to the 60-dB bandwidth of the filter. If the output sampling rate is increased to 187.5 MHz, which is double the 3-dB bandwidth, the monobit receiver can process signals from the filter bank, as shown in Figure 10.17(a). Therefore it is only required to increase the output sampling rate by a factor of two.

The relation in (10.32) can be modified as [15]

$$K = MI, \quad (10.33)$$

where I is an integer, which is referred to as the oversampling ratio. In this section only the oversampling ratio of 2 (or $I = 2$) will be discussed. If $K = 32$ and $I = 2$, then $M = 16$. This means the data will shift 16 points per FFT operation, which is the desired result.

When the filter reaches steady state, the input to the FFT operator is the same as in (10.23), which is rewritten here as

$$\begin{aligned} y(0) &= x(0)h(0) + x(32)h(32) + \cdots + x(224)h(224) \\ y(1) &= x(1)h(1) + x(33)h(33) + \cdots + x(225)h(225) \\ &\quad \dots \\ y(31) &= x(31)h(31) + x(63)h(63) + \cdots + x(255)h(255). \end{aligned}$$

In the second cycle in which the FFT operates, however, the input is different from the result of (10.24) and the desired results are

$$\begin{aligned} y(0) &= x(16)h(0) + x(48)h(32) + \cdots + x(240)h(224) \\ y(1) &= x(17)h(1) + x(49)h(33) + \cdots + x(241)h(225) \\ &\quad \dots \\ y(31) &= x(47)h(31) + x(79)h(63) + \cdots + x(287)h(255). \end{aligned} \quad (10.34)$$

The output starts from $x(16)$ instead of starting from $x(32)$. In order to perform this operation, the arrangement in Figure 10.13 should be modified. This discussion is similar to the discussion in Zahirniak *et al.* [16]. Figure 10.28 shows that the input is decimated into 16 outputs rather than 32. These 16 inputs are fed into filters 0 to 15. The inputs to filters 16 to 31 are obtained from delaying the first 16 outputs by one clock cycle. Some of the inputs are listed in this figure. For example, the input to filter 16 is obtained by delaying the input to filter 0 by one clock cycle, thus this input is $x(16), x(32), \dots$. When the input is decimated by 16 the output speed is doubled, as compared with being decimated by 32.

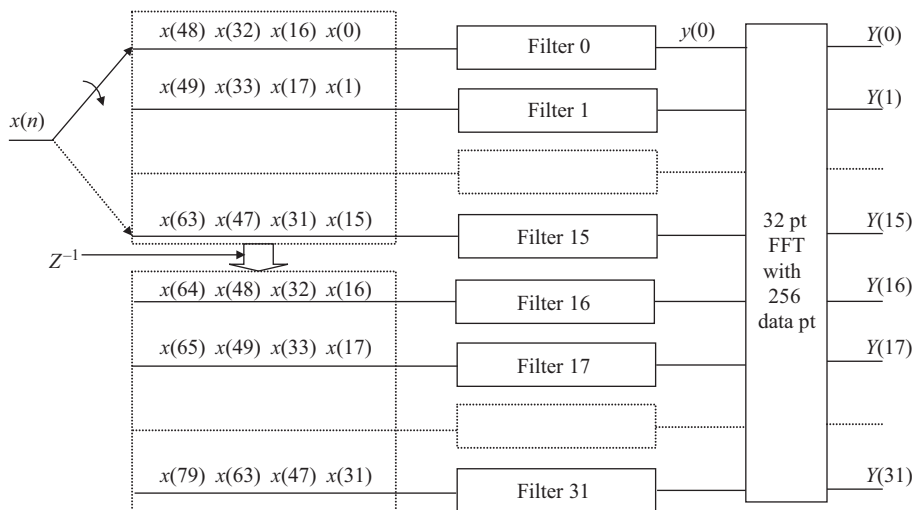


Figure 10.28 Arrangement of oversampling by two.

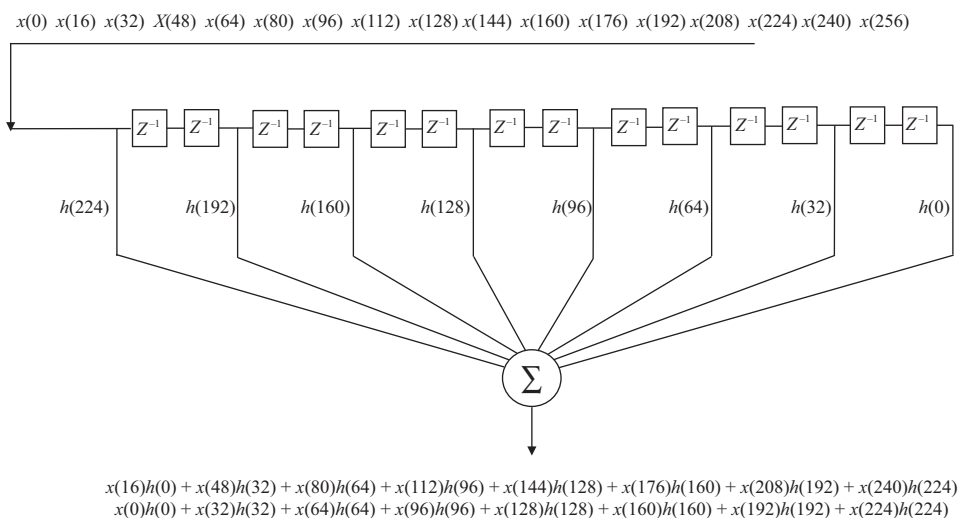


Figure 10.29 Modified filter structure.

The filters in this figure are also modified as shown in Figure 10.29. Between each output there are two delay cycles rather than one. The first two consecutive outputs are shown, and they match the result of the first equation in (10.23) and (10.34). From this arrangement one can see that the output rate to generate $Y(k)$ from the FFT is doubled. The output sampling rate is 187.5 MHz. This is the input rate to the monobit receiver. Because the outputs from the FFT are complex, the monobit receiver has a bandwidth of 187.5 MHz, which is double the 3-dB bandwidth of the output filter.

10.20 Digital Filters Followed by Monobit Receivers

A simulation is used to evaluate the performance of the digital filter followed by monobit receivers. Digital filters are generated through FFT operation. The filter shape is the same as shown in Figure 10.15. The only difference is that the output rate is doubled because each FFT is performed by shifting the input 16 points rather than 32 points. However, this sampling rate change cannot be shown in Figure 10.15. A threshold should be placed at the outputs of each filter. If the output is lower than the threshold, one should consider that there is no signal output from this channel and its output will not be processed. A possible arrangement is shown in Figure 10.30. In this figure, the threshold at the outputs of the filters is not shown. Although the 32-point FFT generates 32 outputs, only 16 of them carry independent information, thus only 16 narrowband monobit receivers are needed.

The channels with signals are processed with the monobit receivers. Only the highest 2 bits from the real and imaginary parts are used as the input of the monobit receiver. This operation should be equivalent to putting a limiting amplifier at the filter output, although a real limiting amplifier cannot operate on complex signals. This limiting action destroys the amplitude information on the input signals, thus the threshold is needed at the channel output. In the time domain, every 16 outputs from a certain frequency bin are used as one input frame of the monobit receiver. Because the output sampling rate is 187.5 MHz, the time to collect 16 samples is about 85 ns ($16/(187.5 \times 10^6)$). This is the same time needed to collect 256 input data points since $256/(3 \times 10^9)$ is also equal to 85 ns. Thus both the digital filter and the monobit receiver process 256 input signals. Because there are only 16 inputs to a monobit receiver, the design can be very simple. There are 16 outputs because the inputs are

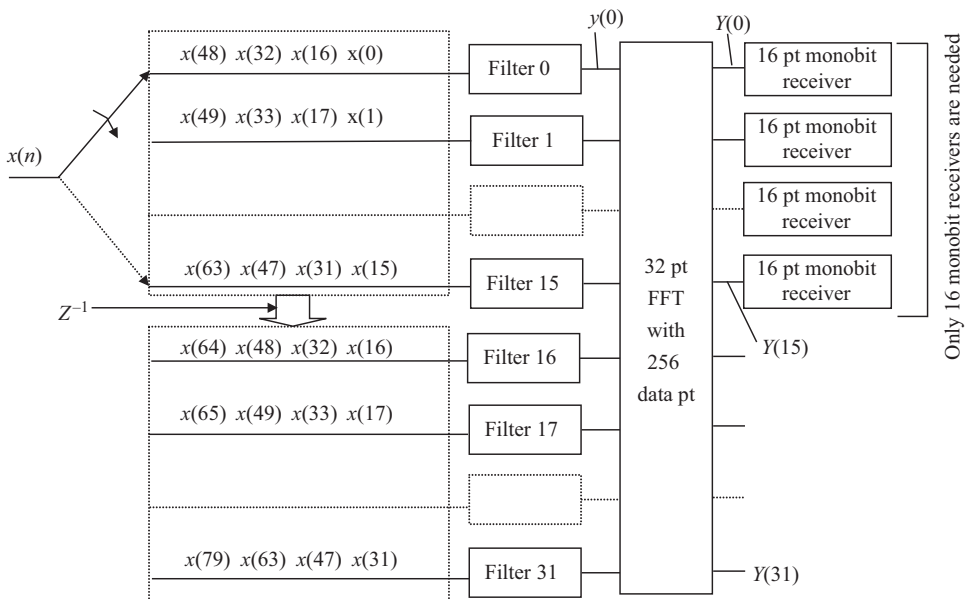


Figure 10.30 Oversampling filters followed by monobit receivers.

complex. These 16 outputs cover a bandwidth of 187.5 MHz, thus each individual output of the monobit receiver is about 11.72 MHz ($187.5/16$). This implies that the receiver can separate two signals separated by about 12 MHz. However, if the amplitudes of the two signals from a wideband digital filter are separated by more than 5 dB, the receiver will miss the weak signal.

Because the FFT operation takes only 16 input points, both the kernel functions in Figures 9.2 and 9.15 are used in the simulation. Although two signals in one channel are shown in both approaches, the eight-point kernel function provides a better result, which means higher instantaneous dynamic range. The monobit receiver limits the instantaneous dynamic range, thus two signals in one channel must have comparable amplitudes to be detected by the receiver. The instantaneous dynamic range of two signals in different channels is determined by the shape of the digital filter. In this simulation the threshold is not included. In order to integrate the monobit receiver into this system, a practical configuration and more thorough simulation are needed.

10.21 Digital Filter Bank Followed by Monobit Receivers and Phase Comparators

As discussed in Section 10.15, the single-frequency data resolution from a monobit receiver is about 9.77 MHz. In the above example, each channel has a width of 11.72 MHz. These bandwidths are narrow enough to separate two simultaneous signals but not fine enough to report a frequency reading. It is desirable to report a finer frequency reading. The fine frequency reading can be obtained by using a phase comparator after the monobit receiver, as discussed in Section 10.15.

Using this approach, a single-frequency precision of about 0.5 MHz should be achievable. In order to obtain this precision, the PW must be increased to a minimum of 512 data points at the input of the receiver, which is about 171 ns. The phase comparator idea as discussed in Section 8.5 can be applied to obtain even finer single-frequency precision on longer pulses. It is highly desirable to measure the frequency precision as a function of PW (i.e., to obtain low-frequency precision on short pulses and high-frequency precision on longer pulses).

10.22 Digital Filter Bank Followed by Another FFT

From the discussion in Section 10.20, one can see that narrowband monobit receivers following digital filters are very simple, performing a 16-point FFT at 187.5 MHz with only 2 bits of real and imaginary inputs. Because the monobit receiver has limited dynamic range, if two signals are in one filter, the monobit receiver may miss the weaker one. With the advances in digital signal processing, it appears that the narrowband monobit receiver can be replaced by a regular FFT operation, which is referred to as the second FFT operation (or chip). This FFT chip will take the channel outputs as input without bit truncation. It performs a 16-point FFT at 187.5 MHz and generates 16 independent outputs since the inputs are complex.

With this kind of arrangement, the receiver should have higher instantaneous dynamic range for signals in the same digital channel. The number of signals per channel is not

limited to two, as in the case of the monobit receivers. Theoretically it should be able to process 16 simultaneous signals per channel. Another potential advantage is that because the inputs to the second FFT are not truncated, the amplitude information on the signals is retained. Thresholds can be set at the outputs of the second FFT, and the detection circuit suggested in Section 10.20 may no longer be needed at the first channel outputs. The frequency data resolution generated from the second FFT operation is about 11.7 MHz ($187.5/16$). Detecting the signals at these outputs should result in higher sensitivity. It is anticipated that this approach may be considered in future wideband digital receiver designs. However, if the transient effect and the interpretation between channels are considered, it is not a trivial task.

References

- [1] Kay SM. *Modern Spectral Estimation Theory and Application*. Englewood Cliffs, NJ: Prentice Hall; 1988.
- [2] Harris FJ. 'Time domain signal processing with the DFT', in Elliott DF (ed.), *Handbook of Digital Signal Processing: Engineering Applications*. San Diego, CA: Academic Press; 1987.
- [3] Allen JB. 'Short term spectral analysis, synthesis, and modification by discrete Fourier transform'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1977;**25**(3):235–238.
- [4] Allen JB, Rabiner L. 'A unified approach to short-time Fourier analysis and synthesis'. *Proceedings of the IEEE* 1977;**65**(11):1558–1564.
- [5] Harris FJ. 'On the use of windows for harmonic analysis with the discrete Fourier transform'. *Proceedings of the IEEE* 1978;**66**(1):51–83.
- [6] Tran-Thong. 'Practical consideration for a continuous time digital spectrum analyser'. *Proceedings of IEEE International Symposium on Circuits and Systems*, vol. 2. New York: IEEE; 1989:1047–1050.
- [7] Crochiere RE, Rabiner LR. *Multirate Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1983.
- [8] Vaidyanathan PP. *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice Hall; 1992.
- [9] Vaidyanathan PP. 'Multirate digital filters, filter banks, polyphase networks, and applications: a tutorial'. *Proceedings of the IEEE* 1990;**78**(1):56–93.
- [10] Ansari R, Liu B. 'Multirate signal processing', in Mitra SK, Kaiser JF (eds.), *Handbook for Digital Signal Processing*. New York: John Wiley & Sons; 1993.
- [11] Oppenheim AV, Schaffer RW. *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1975.
- [12] Rabiner LR. *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1975.
- [13] Fields TW, Sharpin DL, Tsui JB. 'Digital channelized IFM receiver'. *Proceedings of IEEE MTT-S International Microwave Symposium Digest*, vol. 3. New York: IEEE; 1994:1667–1670.
- [14] McCormick W. Professor of electrical engineering, Wright State University, Dayton, OH. Private communication.

- [15] Crochiere RE, Rabiner LR. *Multirate Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1983:311.
- [16] Zahirniak DR, Sharpin DL, Fields TW. 'A hardware-efficient, multirate, digital channelized receiver architecture'. *IEEE Transactions on Aerospace and Electronic Systems* 1998;**34**(1):137–152.
- [17] Vary P, Heute U. 'A short-time spectrum analyzer with polyphase-network and DFT'. *Signal Processing* 1980;**2**(1):55–65.

High-Resolution Spectrum Estimation

11.1 Introduction

In previous chapters, most of the discussion on spectrum estimation has concentrated on the fast Fourier transform (FFT) since it is used in digital receiver designs. In this chapter, other spectrum estimation approaches are introduced that will be referred to as high-resolution spectrum estimations. Their major advantage is that they can provide higher frequency resolution than the FFT, especially on simultaneous signals.

If there are two signals with frequencies very close, an FFT operation may generate one peak containing both signals. High-resolution spectrum estimation may separate the two signals by generating two sharp peaks. The major drawback of applying high-resolution spectrum estimation to digital microwave receivers is the complexity of the operation. Because of the large number of operations required to estimate the frequencies, this type of estimation will probably not be implemented for real-time application in the near future. However, it might be implemented for special applications. For example, if the peak of an FFT output appears to contain more than one signal, high-resolution spectrum estimation might be used to find the frequencies, thus the operation may not be required on all the input data.

Many different high-resolution approaches can be used to estimate frequencies from digitized input data. In this chapter, seven high-resolution methods are discussed:

1. Linear predication (or autoregressive [AR])
2. Prony's
3. Least squares Prony's
4. Multiple signal classification (MUSIC)
5. Estimation of signal parameters via rotational invariance techniques (ESPRIT)
6. Minimum norm
7. Minimum norm with discrete Fourier transform (DFT).

The input data can be manipulated before the linear prediction is applied. Depending on the manipulation of the data, the AR method can be briefly subdivided into the forward, backward, and Burg methods.

Finally, an adaptive method will be introduced. This method takes the input data and strips off one signal at a time.

Some of these methods can produce satisfactory visual displays so that one can determine the frequencies by observing the output result. It has been emphasized numerous times in the previous chapters that an electronic warfare (EW) receiver must generate digital words as output. The outputs from these spectrum estimation methods must be converted into digital words with real-time processing. In some methods, determining the order of operation must also be implemented in real time.

Two excellent references are Childers [1] and Kesler [2], both of which contain a collection of papers. Computer programs used to generate some of the figures are listed in the appendix of this chapter.

11.2 AR Method [1–18]

In time series, a powerful model is called the prediction method. It is assumed that the present value can be predicted from past values. For example, prediction can be used in many areas (i.e., environmental trends, weather forecasting, stock market movement), although the reliability is questionable.

If it is used in spectrum estimation, the present value can be written as a linear combination of input and output:

$$x(n) = -\sum_{i=1}^p a_i x(n-i) + Gu(n) + G \sum_{l=1}^q b_l u(n-l), \quad (11.1)$$

where $x(n)$ is the digitized data, a_i and b_l are constants, G is the gain of the system, and $u(n)$ represents white noise. In statistics, this equation is called the autoregressive moving average (ARMA) model.

If one takes the z transform of this equation, the result is

$$X(z) = -\sum_{i=1}^p a_i X(z)z^{-i} + GU(z) + G \sum_{l=1}^q b_l U(z)z^{-l}. \quad (11.2)$$

In this equation, the white noise is usually considered as input and the data are the output. Thus the transfer function $H(z)$ of this equation is defined as the output divided by the input. The result is

$$H(z) = \frac{X(z)}{U(z)} = G \frac{1 + \sum_{l=1}^q b_l z^{-l}}{1 + \sum_{i=1}^p a_i z^{-i}}. \quad (11.3)$$

This equation is called the general pole-zero form because the transfer function has both zeros and poles. The zeros are the z values that cause the numerator to be zero and the poles are the z values that cause the denominator to be zero.

If all the a_i values are zero in (11.1), the equation becomes

$$x(n) = Gu(n) + G \sum_{l=1}^q b_l u(n-l). \quad (11.4)$$

This equation is called the moving average (MA) model. Its corresponding transfer function is

$$H(z) = \frac{X(z)}{U(z)} = G \left(1 + \sum_{l=1}^q b_l z^{-l} \right). \quad (11.5)$$

This is an all-zero model. In filter design, this is referred to as finite impulse response (FIR) filtering. To solve the constants b_l from $x(n)$, nonlinear equations will be formed, therefore it is difficult to solve.

If all the b_l values are zero in (11.1), the equation becomes

$$x(n) = - \sum_{i=1}^p a_i x(n-i) + Gu(n). \quad (11.6)$$

This is called the AR model. It is also called the linear prediction model, because the present value can be predicted by a linear combination of past output values. Its corresponding transfer function is

$$H(z) = \frac{X(z)}{U(z)} = \frac{G}{1 + \sum_{i=1}^p a_i z^{-i}}. \quad (11.7)$$

This is an all-pole model. In filter design, this model is also known as the infinite impulse response (IIR) filter.

The spectrum generated by (11.7) is narrowband since $H(z)$ has a very sharp peak when the denominator approaches zero (i.e., a pole is close to unit circle). Also, to obtain the constants a_i in (11.6), the processing is linear.

11.3 Yule–Walker Equation [1–21]

The linear prediction defined in (11.6) and (11.7) can be considered as a filter with noise $u(n)$ as input and $x(n)$ as output. This filter is shown in Figure 11.1. Figure 11.1(a) shows the overall function of the filter and Figure 11.1(b) shows the feedback circuit. The first step is to find the coefficients of the filter. If the input $u(n)$ is assumed to be an unknown response, the signal $x(n)$ can be predicted only approximately from a linearly weighted summation of the past terms. As a result, the linear prediction expression can be written as

$$\hat{x}(n) = - \sum_{i=1}^p a_i x(n-i), \quad (11.8)$$

where $\hat{x}(n)$ represents the estimate of $x(n)$. For simplicity, $x(n)$ is used for $\hat{x}(n)$ in later discussions. Replacing $x(n)$ by $x(n-1)$, $x(n-2)$, ..., $x(n-p+1)$, one can obtain a set of p linear equations. For example, if there are four data points $x(1)$, $x(2)$, $x(3)$ and $x(4)$, and $p=2$, there will be two equations with constants a_1 and a_2 . These equations can be written as

$$\begin{aligned} -x(4) &= a_1 x(3) + a_2 x(2) \\ -x(3) &= a_1 x(2) + a_2 x(1). \end{aligned} \quad (11.9)$$

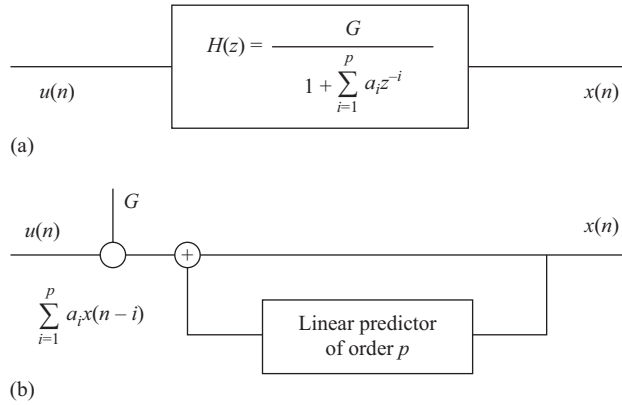


Figure 11.1 The AR model: (a) plane result and (b) equivalent circuit.

Since all the $x(n)$ are known, theoretically, it is possible to determine the a values. This result will be equivalent to Prony's method, which originates from a slightly different point of view.

The above equation has one possible shortcoming in that the data points are usually contaminated with noise. The usual way to solve this problem is by least squares. This approach can be demonstrated through the following procedure. Equation (11.6) can be rewritten as

$$x(n) = - \sum_{i=1}^p a_i x(n-i) + u(n), \quad (11.10)$$

where the gain $G = 1$ is assumed. Multiplying both sides by $x^*(n-k)$, where the superscript $*$ represents the complex conjugate, and taking the expectation of them, the result is

$$E[x(n)x^*(n-k)] = -E \left[\sum_{i=1}^p a_i x(n-i)x^*(n-k) \right] + E[u(n)x^*(n-k)]. \quad (11.11)$$

The following "sample" autocorrelation is used in place of the expectation value:

$$E[x(n)x(n-k)] = \frac{1}{N} \sum_{n=0}^{N-k-1} x(n)x(n-k). \quad (11.12)$$

From the definition of autocorrelation $R(k-i)$, the expectation value can be written as

$$R(k-i) = E[x(n-i)x^*(n-k)]. \quad (11.13)$$

The data point $x(n)$ is composed of two parts, the signal and the noise, which can be written as

$$x(n) = x_s(n) + u(n), \quad (11.14)$$

where $x_s(n)$ is the signal without noise. Thus

$$E[u(n)x^*(n-k)] = E[u(n)x_s^*(n-k)] + E[u(n)u^*(n-k)]. \quad (11.15)$$

The first part of this equation is zero since the signal and the noise are uncorrelated. The second part can be written as

$$\begin{aligned} E[u(n)u^*(n-k)] &= 0 && \text{when } k \neq 0 \\ &= \sigma^2 && \text{when } k = 0 \end{aligned} \quad (11.16)$$

since the noise is uncorrelated. When $k = 0$, the expectation value equals the noise power. From (11.12) through (11.16), (11.11) can be written as

$$\begin{aligned} R(k) &= -\sum_{i=1}^p a_i R(k-i) \quad \text{for } k \neq 0 \\ R(k) &= -\sum_{i=1}^p a_i R(k-i) + \sigma^2 \quad \text{for } k = 0. \end{aligned} \quad (11.17)$$

These two equations are the Yule–Walker equations. If p is given, these two equations can be written explicitly as

$$\begin{aligned} -R(0) &= a_1 R(-1) + a_2 R(-2) + \cdots + a_p R(-p) - \sigma^2 \quad \text{for } k = 0 \\ -R(1) &= a_1 R(0) + a_2 R(-1) + \cdots + a_p R(-p+1) \quad \text{for } k = 1 \\ -R(p) &= a_1 R(p-1) + a_2 R(p-2) + \cdots + a_p R(0) \quad \text{for } k = p. \end{aligned} \quad (11.18)$$

These equations are linear equations of a_i and they can be rearranged and written in matrix form as

$$\begin{bmatrix} R(0) & R(-1) & \cdots & R(-p) \\ R(1) & R(0) & \cdots & R(-p+1) \\ \cdots & \cdots & \cdots & \cdots \\ R(p) & R(p-1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (11.19)$$

If the first equation in (11.18) is removed, the remaining equations can be written in matrix form as

$$\begin{bmatrix} R(0) & R(-1) & \cdots & R(-p+1) \\ R(1) & R(0) & \cdots & R(-p+2) \\ \cdots & \cdots & \cdots & \cdots \\ R(p-1) & R(p-2) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(p) \end{bmatrix} \quad (11.20)$$

The Yule–Walker equations can also be obtained from the least mean square (LMS) approach. This approach is illustrated as follows. From (11.8), the error can be written as

$$e(n) = x(n) - \hat{x}(n) = x(n) + \sum_{i=1}^p a_i x(n-i). \quad (11.21)$$

The coefficients a_i can be obtained by minimizing the summation of the error square, which can be written as

$$\sum_{n=0}^{N-i-1} |e(n)|^2 = \sum_{n=0}^{N-i-1} \left| x(n) + \sum_{i=1}^p a_i x(n-i) \right|^2. \quad (11.22)$$

To minimize the above equation, one can take the derivative with respect to a_i and set the result to zero. Since a_i is a complex number, taking the derivative with respect to it will break it into real and imaginary parts. This is known as the result of the LMS solution [4, 19]:

$$\begin{aligned} \sum_{n=0}^{N-2} \left(x(n) + \sum_{i=1}^p a_i x(n-i) \right) x^*(n-1) &= 0 \\ \sum_{n=0}^{N-3} \left(x(n) + \sum_{i=1}^p a_i x(n-i) \right) x^*(n-2) &= 0 \\ &\dots \\ \sum_{n=1}^{N-p-1} \left(x(n) + \sum_{i=1}^p a_i x(n-i) \right) x^*(n-p) &= 0. \end{aligned} \quad (11.23)$$

Using the definition of autocorrelation, these results are the same as those shown in (11.18) if all the expected values are replaced by estimates except the first one.

The first equation in (11.18) can be obtained as follows. By substituting these results into (11.22), the minimum of the summation of error squared is equal to the noise, which can be written as

$$\begin{aligned} \frac{1}{N} \sum |e_{\min}(n)|^2 &= \frac{1}{N} \sum_{n=1}^N \left| x(n) + \sum_{i=1}^p a_i x(n-i) \right|^2 \\ &= \frac{1}{N} \sum_{n=1}^N \left(x(n) + \sum_{i=1}^p a_i x(n-i) \right) \left(x(n) + \sum_{i=1}^p a_i x(n-i) \right)^* \\ &= \frac{1}{N} \sum_{n=1}^N x(n) x^*(n) + \frac{1}{N} \sum_{n=1}^N \sum_{i=1}^p a_i x(n-i) x^*(n). \end{aligned} \quad (11.24)$$

After multiplying the two terms in this equation, the end result will contain four terms. The last two terms are zero if the relation of (11.23) is used. By using the definition of autocorrelation, this equation is the same as the first one in (11.18). Therefore the Yule–Walker equation can be obtained from a different approach.

11.4 Levinson–Durbin Recursive Algorithm [1–21]

In (11.19), all the diagonal elements including the off diagonal of the R matrix are equal. This kind of matrix is called Toeplitz. A Toeplitz matrix equation can be solved through the Levinson–Durbin algorithm, which is a recursive approach and more calculation-efficient

than directly solving from the Yule–Walker equation. The results of the recursive equation can be written as

$$\begin{aligned}
 \sigma_0^2 &= R(0) \\
 \sigma_j^2 &= (1 - a_{j,j}^2) \sigma_{j-1}^2 \\
 &\quad - \left[R(j) + \sum_{i=1}^{j-1} a_{i,j-1} R(j-i) \right] \\
 a_{j,j} &= \frac{\left[R(j) + \sum_{i=1}^{j-1} a_{i,j-1} R(j-i) \right]}{\sigma_{j-1}^2} \\
 a_{i,j} &= a_{i,j-1} + a_{j,j} a_{j-i,j-1}^*,
 \end{aligned} \tag{11.25}$$

where σ_j is called the prediction error power, which may be used to determine the order of the AR method. The first subscript in $a_{j,i}$ is the numerical order of the constants and the second subscript is the number of recursions. The value of σ_j should decrease when j increases. When the correct order is reached theoretically, its value will stay constant.

The illustration of using (11.25) can be shown as follows. For a final order of $p = j = 2$, the first order is $j = 1$, there are $R(0)$ and $R(1)$, with $\sigma_0 = R(0)$ and $a_{1,1} = -R(1)/R(0)$. For the second order, there are $R(0)$, $R(1)$, and $R(2)$ $a_{2,2} = -[R(2) + a_{1,1}R(1)]/\sigma_1^2$, where $\sigma_1 = (1 - a_{1,1}^2)\sigma_0^2$ and $a_{1,2} = a_{1,1} + a_{2,2}a_{1,1}^*$. The final results are $a_{1,2} = a_1$ and $a_{2,2} = a_2$, where a_1 and a_2 are the constants in (11.20).

In the above discussion, the autocorrelations are used to calculate the constants a_i . These quantities are obtained from averaging the input data. Therefore the coefficients a_i calculated should be better than the results obtained from (11.8).

After the values of a_i are obtained, the results can be substituted into (11.7) to find the spectrum response. The gain of the system equals the variance of the noise power and the variable z is replaced by

$$z = e^{j2\pi f t_s}. \tag{11.26}$$

The power spectrum obtained from the AR model can be obtained from (11.7) as

$$P_{AR}(f) = |H(e^{j2\pi f t_s})|^2 = \frac{\sigma^2}{\left| 1 + \sum_{i=1}^p a_i e^{-j2\pi f} \right|^2}, \tag{11.27}$$

where t_s is assumed. The values of σ^2 can be obtained as discussed in the Levinson–Durbin recursive equation.

Let us use an example to demonstrate the AR model. Assume that the input signal consists of three sinusoidal waves without noise. The data are generated by

$$x(n) = \cos(2\pi \cdot 0.21n + 0.1) + 2 \cos(2\pi \cdot 0.36n) + 1.9 \cos(2\pi \cdot 0.38n), \tag{11.28}$$

where $n = 0, 1, \dots, 31$, for a total of 32 data points. This input contains three signals, two of them with frequencies close together, but there is no noise.

The data are padded with 4064 zeros, to a total length of 4096 points, and the FFT is performed. The result is shown in Figure 11.2. In this figure, the two frequencies 0.36 and 0.38 cannot be separated and they form a single peak.

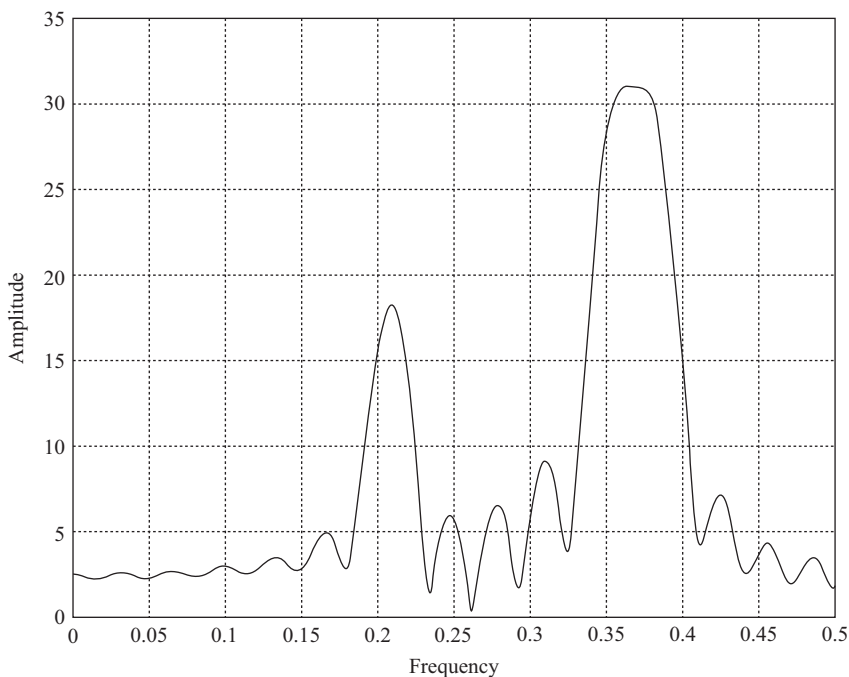


Figure 11.2 The FFT output of the input signal.

Figure 11.3 shows the result obtained using the AR modeling approach and (11.27). In Figures 11.3(a) through 11.3(c), the orders of the process are $p = 14, 20$, and 30 , respectively. From these figures one can see that when the order is low, signals with frequencies close together may not show. When the order is too high, spurious signals may appear. This problem may also occur in other high-resolution spectrum estimation methods. Hence correct order determination is an important issue in AR modeling or any high-resolution method.

11.5 Input Data Manipulations [3, 4, 13, 19, 22]

In (11.13), the R matrix is obtained from the autocorrelations with different lags. However, this is not the only way to form the R matrix. The data can be manipulated differently to obtain different results. In other words, the autocorrelation matrix is not the only approach to obtain the a_i constants in (11.18). Some of the approaches can improve the quality of the spectrum estimation. In order to use other ways to adopt the input data, (11.19) is rewritten as

$$\begin{bmatrix} T_{00} & T_{01} & \cdots & T_{0p} \\ T_{10} & T_{11} & \cdots & T_{1p} \\ \cdots & \cdots & \cdots & \cdots \\ T_{p0} & T_{p1} & \cdots & T_{pp} \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (11.29)$$

This equation is identical to (11.19), except that the matrix T is used to replace R .

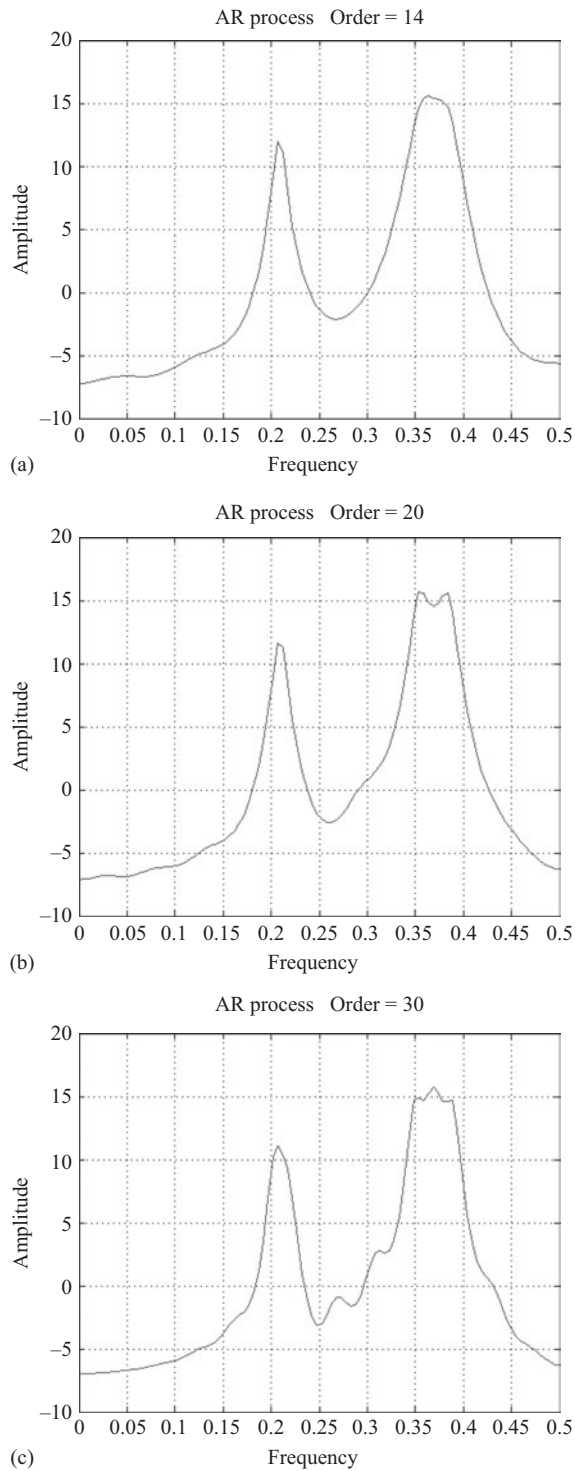


Figure 11.3 Results from the AR process: (a) $p = 14$, (b) $p = 20$, and (c) $p = 30$.

Different ways of generating the T matrix from the same input data can produce quite different results. Two different ways to obtain T will be discussed. Assume there are N input data points from $x(0)$ to $x(N-1)$.

11.5.1 Covariance Method

In this method the input data are related to matrix T as

$$\begin{bmatrix} T_{00} & T_{01} & \cdots & T_{0p} \\ T_{10} & T_{11} & \cdots & T_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ T_{p0} & T_{p1} & \cdots & T_{pp} \end{bmatrix} = \frac{1}{N-p} \begin{bmatrix} x^*(p) & x^*(p+1) & \cdots & x^*(N-1) \\ x^*(p-1) & x^*(p) & \cdots & x^*(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ x^*(0) & x^*(1) & \cdots & x^*(N-p-1) \end{bmatrix} \\ \times \begin{bmatrix} x(p) & x(p-1) & \cdots & x(0) \\ x(p+1) & x(p) & \cdots & x(1) \\ \vdots & \vdots & \ddots & \vdots \\ x(N-1) & x(N-2) & \cdots & x(N-p-1) \end{bmatrix}. \quad (11.30)$$

This matrix is not Toeplitz. Although a special recursive method can be developed for this approach, the Levinson–Durbin method does not apply directly. In this approach, all the elements in the T matrix contain the same number of terms from the data points.

If $p = 1$ and $N = 3$, the data are $x(0)$, $x(1)$, and $x(2)$ and the T matrix is

$$\begin{bmatrix} T_{00} & T_{01} \\ T_{10} & T_{11} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} x^*(1) & x^*(2) \\ x^*(0) & x^*(1) \end{bmatrix} \begin{bmatrix} x(1) & x(0) \\ x(2) & x(1) \end{bmatrix}. \quad (11.31)$$

If $p = 1$ and $N = 4$, the data are $x(0)$, $x(1)$, $x(2)$, and $x(3)$ and the T matrix is

$$\begin{bmatrix} T_{00} & T_{01} \\ T_{10} & T_{11} \end{bmatrix} = \frac{1}{3} \begin{bmatrix} x^*(1) & x^*(2) & x^*(3) \\ x^*(0) & x^*(1) & x^*(2) \end{bmatrix} \begin{bmatrix} x(1) & x(0) \\ x(2) & x(1) \\ x(3) & x(2) \end{bmatrix}. \quad (11.32)$$

This is a common way to obtain the R matrix. In this chapter, some of the examples use this method to obtain the R matrix. This approach provides accurate frequency estimates for no noise and correct order.

11.5.2 Autocorrelation Method

In this approach, the result of the T matrix is identical to that obtained from the autocorrelation function and that is why the name is used. In this approach, the data points

outside of the range $x(0)$ to $x(N-1)$ are assumed to be zeros, and this implies a window function is applied. The T matrix is related to the input data as

$$\begin{bmatrix} T_{10} & T_{11} & \cdots & T_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ T_{p0} & T_{p1} & \cdots & T_{pp} \end{bmatrix} = \frac{1}{N} \begin{bmatrix} 0 & x^*(0) & \cdots & \cdots & x^*(N-1) & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & x^*(0) & \cdots & \cdots & x^*(N-1) \end{bmatrix} \\ \times \begin{bmatrix} x(0) & 0 & \cdots & 0 \\ x(1) & x(0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x(N-1) & x(N-2) & \cdots & x(0) \\ 0 & x(N-1) & \cdots & x(1) \\ 0 & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & x(N-1) \end{bmatrix} \quad (11.33)$$

The number of zeros added in front of $x(0)$ and after $x(N-1)$ equals p . The T matrix obtained this way is Toeplitz, thus the Levinson–Durbin algorithm can be used to solve for the coefficients in (11.29).

If $p = 1$ and $N = 3$, the data are $x(0)$, $x(1)$, and $x(2)$ and the T matrix is

$$\begin{bmatrix} T_{00} & T_{01} \\ T_{10} & T_{11} \end{bmatrix} = \frac{1}{3} \begin{bmatrix} x^*(0) & x^*(1) & x^*(2) & 0 \\ 0 & x^*(0) & x^*(1) & x^*(2) \end{bmatrix} \begin{bmatrix} x(0) & 0 \\ x(1) & x(0) \\ x(2) & x(1) \\ 0 & x(2) \end{bmatrix}. \quad (11.34)$$

If $p = 1$ and $N = 4$, the data are $x(0)$, $x(1)$, $x(2)$, and $x(3)$ and the T matrix is

$$\begin{bmatrix} T_{00} & T_{01} \\ T_{10} & T_{11} \end{bmatrix} = \frac{1}{4} \begin{bmatrix} x^*(0) & x^*(1) & x^*(2) & x^*(3) & 0 \\ 0 & x^*(0) & x^*(1) & x^*(2) & x^*(3) \end{bmatrix} \begin{bmatrix} x(0) & 0 \\ x(1) & x(0) \\ x(2) & x(1) \\ x(3) & x(2) \\ 0 & x(3) \end{bmatrix}. \quad (11.35)$$

The same number of zeros is added as in the above case. This approach may not produce the correct frequency, even though there is no noise and the correct order is selected.

11.6 Backward Prediction and the Modified Covariance Method [3, 4, 13, 19, 22]

The two methods discussed in the next section can be considered as subcategories of the AR method. The purpose of introducing the backward prediction and modified

covariance methods is to prepare for the discussion of the Burg method. The linear prediction discussed in previous sections is referred to as forward prediction, since past data are used to predict present data. In backward prediction, past data are predicted from present data. This may sound a little absurd since the present data are already known. However, if one considers the time series as a data set rather than a sequence occurring in time, one can linearly predict in either direction. The backward prediction can be written as

$$\hat{x}(n-p) = - \sum_{i=1}^p c_i x(n-p+i), \quad (11.36)$$

where c_i is the backward prediction coefficient.

The relation between the backward and forward coefficients can be found as follows. The Yule–Walker equation can be obtained from (11.36) by multiplying both sides by $x^*(n-p-k)$ and taking the expectation value. The result is

$$E[x(n-p)x^*(n-p-k)] = - \sum_{i=1}^p c_i E[x(n-p+i)x^*(n-p-k)]. \quad (11.37)$$

From the definition of autocorrelation, this equation can be written as

$$R(k) = - \sum_{i=1}^p c_i R(k+i). \quad (11.38)$$

Written in separate equations, the result is

$$\begin{aligned} -R(-1) &= c_1 R(0) + c_2 R(1) + \cdots + c_p R(p-1) & k = -1 \\ -R(-2) &= c_1 R(-1) + c_2 R(0) + \cdots + c_p R(p-2) & k = -2 \\ &\dots \\ -R(-p) &= c_1 R(-p+1) + c_2 R(-p+2) + \cdots + c_p R(0) & k = -p. \end{aligned} \quad (11.39)$$

In matrix form it is

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(p-1) \\ R(-1) & R(0) & \cdots & R(p-2) \\ \cdots & \cdots & \cdots & \cdots \\ R(-p+1) & R(-p+2) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_p \end{bmatrix} = - \begin{bmatrix} R(-1) \\ R(-2) \\ \vdots \\ R(-p) \end{bmatrix} \quad (11.40)$$

Now, let us find the relation between the forward prediction coefficients a_i and the backward prediction coefficients c_i . Taking the complex conjugate on both sides of the above equation, the result is

$$\begin{bmatrix} R^*(0) & R^*(1) & \cdots & R^*(p-1) \\ R^*(-1) & R^*(0) & \cdots & R^*(p-2) \\ \cdots & \cdots & \cdots & \cdots \\ R^*(-p+1) & R^*(-p+2) & \cdots & R^*(0) \end{bmatrix} \begin{bmatrix} c_1^* \\ c_2^* \\ \vdots \\ c_p^* \end{bmatrix} = - \begin{bmatrix} R^*(-1) \\ R^*(-2) \\ \vdots \\ R^*(-p) \end{bmatrix}. \quad (11.41)$$

Applying the relation $R^*(i) = R(-i)$, the above equation is

$$\begin{bmatrix} R(0) & R(-1) & \cdots & R(p-1) \\ R(1) & R(0) & \cdots & R(p-2) \\ \cdots & \cdots & \cdots & \cdots \\ R(p-1) & R(p-2) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} c_1^* \\ c_2^* \\ \vdots \\ c_p^* \end{bmatrix} = - \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(p) \end{bmatrix}. \quad (11.42)$$

Comparing with (11.20), it is obvious that

$$c_i^* = a_i. \quad (11.43)$$

This is the relation between the coefficient of the forward and backward prediction.

A backward prediction error $b(n)$ is similar to the forward prediction error $e(n)$ in (11.21) and can be defined from (11.36) as

$$\begin{aligned} b(n) &= x(n-p) - \hat{x}(n-p) = x(n-p) + \sum_{i=1}^p c_i x(n-p+i) \\ &= x(n-p) + \sum_{i=1}^p a_i^* x(n-p+i), \end{aligned} \quad (11.44)$$

which can be written in a slightly different form as

$$\begin{aligned} b(n) &= x(n) - \hat{x}(n) = x(n) + \sum_{i=1}^p c_i x(n+i) \\ &= x(n) + \sum_{i=1}^p a_i^* x(n+i). \end{aligned} \quad (11.45)$$

The square of the backward error can be minimized to generate an equation similar to the Yule–Walker equations.

The modified covariance method minimizes the sum of the square of the average linear prediction error. The square of the average linear prediction error $\epsilon(n)$ is defined in terms of the forward and backward linear prediction errors as

$$\epsilon(n) = \frac{1}{2(N-p)} \left[\sum_{n=p}^{N-1} e^2(n) + \sum_{n=0}^{N-p-1} b^2(n) \right], \quad (11.46)$$

where $e(n)$ and $b(n)$ are the forward and backward prediction errors, respectively, defined in (11.21) and (11.44). By minimizing $\epsilon(n)$ by taking the derivative with respect to a_i and setting the result to zero, an equation identical to (11.29) can be obtained. The corresponding T matrix can be written as

$$\begin{bmatrix} T_{00} & T_{01} & \cdots & T_{0p} \\ T_{10} & T_{11} & \cdots & T_{1p} \\ \cdots & \cdots & \cdots & \cdots \\ T_{p0} & T_{p1} & \cdots & T_{pp} \end{bmatrix} = \left[\frac{1}{2(N-p)} \right] \left\{ \begin{bmatrix} x^*(p) & x^*(p+1) & \cdots & x^*(N-1) \\ x^*(p-1) & x^*(p) & \cdots & x^*(N-2) \\ \cdots & \cdots & \cdots & \cdots \\ x^*(0) & x^*(1) & \cdots & x^*(N-p-1) \end{bmatrix} \right. \\ \times \begin{bmatrix} x(p) & x(p-1) & \cdots & x(0) \\ x(p+1) & x(p) & \cdots & x(1) \\ \cdots & \cdots & \cdots & \cdots \\ x(N-1) & x(N-2) & \cdots & x(N-p-1) \end{bmatrix} \\ + \begin{bmatrix} x(0) & x(1) & \cdots & x(N-p-1) \\ x(1) & x(2) & \cdots & x(N-p) \\ \cdots & \cdots & \cdots & \cdots \\ x(p) & x(p+1) & \cdots & x(N-1) \end{bmatrix} \\ \left. \times \begin{bmatrix} x^*(0) & x^*(1) & \cdots & x^*(p) \\ x^*(1) & x^*(2) & \cdots & x^*(p+1) \\ \cdots & \cdots & \cdots & \cdots \\ x^*(N-p-1) & x^*(N-p) & \cdots & x^*(N-1) \end{bmatrix} \right\}. \quad (11.47)$$

The modified covariance method uses the data more times than either the forward or backward prediction method. The modified covariance method appears to be less sensitive to noise because of the enlarged data set. It is also less sensitive to the initial phase of the input signal than the autocorrelation method. Haykin [22] developed a relatively efficient method to solve for the coefficients.

11.7 Burg Method [3, 4, 13, 23–32]

One of the most popular approaches in linear prediction spectrum estimation is the Burg method, which has also been called the maximum entropy method (MEM). If there are n points of data from $x(0)$ to $x(N-1)$ and $p+1$ lags of autocorrelations from $R(0)$ to $R(p)$, Burg suggested that the unknown autocorrelation lags from $R(p+1)$, $R(p+2)$, ... can be extrapolated from the input data points. There are an infinite number of ways to extrapolate the autocorrelations. Burg further suggested that the extrapolation of the autocorrelations should not add any new information arbitrarily to the sequence. The information is measured in terms of entropy from Shannon's theorem. Maximizing the entropy implies that the time series is in the most random state and no new information is arbitrarily added to the series. Thus the name MEM is used.

Later investigators showed that in order to use the MEM, the autocorrelation of the time series must be known. However, the data obtained from most experiments are, in general, a series of real or complex values as a function of time. In other words, the only known data

are the time series, not the autocorrelations of the input signal. The autocorrelations calculated using the time series are not the true values, only estimations. Thus the promise of the MEM is never delivered in practice. As a result, the term MEM is no longer popular. Instead, the Burg method is used to honor the inventor.

Instead of estimating the autocorrelation lags from $R(p+1)$, $R(p+2)$, ... directly, Burg devised a new method. This method is quite similar to the modified covariance method. The sum of the squares of the average linear prediction error is minimized. The only difference between these methods is that the Burg method uses a constraint in the process of the minimization to ensure the filter is stable. The constraint is the Levinson recursive condition in (11.25):

$$a_{i,j} = a_{i,j-1} + a_{j,j}a_{j-i,j-1}^* \quad (11.48)$$

In this section, only the results of the Burg method will be presented. Let us define

$$\begin{aligned} e_{0,n} &= x(n) & \text{for } n = 1, 2, \dots, N-1 \\ b_{0,n} &= x(n) & \text{for } n = 0, 1, \dots, N-2 \\ e_{i,n} &= e_{i-1,n} + \Gamma_i b_{i-1,n-1} \\ b_{i,n} &= b_{i-1,n-1} + \Gamma_i^* e_{i-1,n-1}, \end{aligned} \quad (11.49)$$

where $e_{i,n}$ and $b_{i,n}$ represent the forward and backward errors, respectively. The coefficient a values can be obtained as

$$\begin{aligned} a_{j,j} &= \Gamma_j = \frac{-2 \sum_{n=j}^{N-1} e_{n,j-1} b_{n-1,j-1}^*}{\sum_{n=j}^{N-1} \left[|e_{n,j-1}|^2 + |b_{n-1,j-1}|^2 \right]} \\ a_{i,j} &= a_{i,j-1} + a_{j,j} a_{j-i,j-1}^* \quad \text{for } j = 1, 2, \dots, i-1 \\ \epsilon_j &= \left(1 - |\Gamma_j|^2 \right) \epsilon_{j-1}, \end{aligned} \quad (11.50)$$

where Γ_j is referred to as the j th reflection coefficient and ϵ_j is the j th error. The double subscripts are used here because the approach is recursive. The first subscript represents the i th order of the coefficient $a_{i,j}$ and the second subscript represents the recursive order.

The Burg method can produce very high spectrum peaks with short data lengths and can resolve signals close in frequency. However, the frequency bias depends on the initial phase of the input signal and the length of the data. Another problem with this method is spectral line splitting. This means that when there is only one signal, the spectrum may show two very close spectra. Spectral line splitting depends on the initial phase of the input signal.

Windows can be added to the Burg method to reduce bias error and spectral line splitting. The window function can be inserted into the reflection coefficient as

$$\Gamma_j = \frac{-2 \sum_{n=j}^{N-1} w_{n,j} e_{n,j-1} b_{n-1,j-1}^*}{\sum_{n=j}^{N-1} w_{n,j} \left[|e_{n,j-1}|^2 + |b_{n-1,j-1}|^2 \right]}. \quad (11.51)$$

Here, $w_{n,j}$ is the window function.

The result of the Burg method can be found in MATLAB® (MathWorks, Natick, MA, USA) with instructions “lpc” and “freqz.” The result of the Burg method is shown in Figure 11.4.

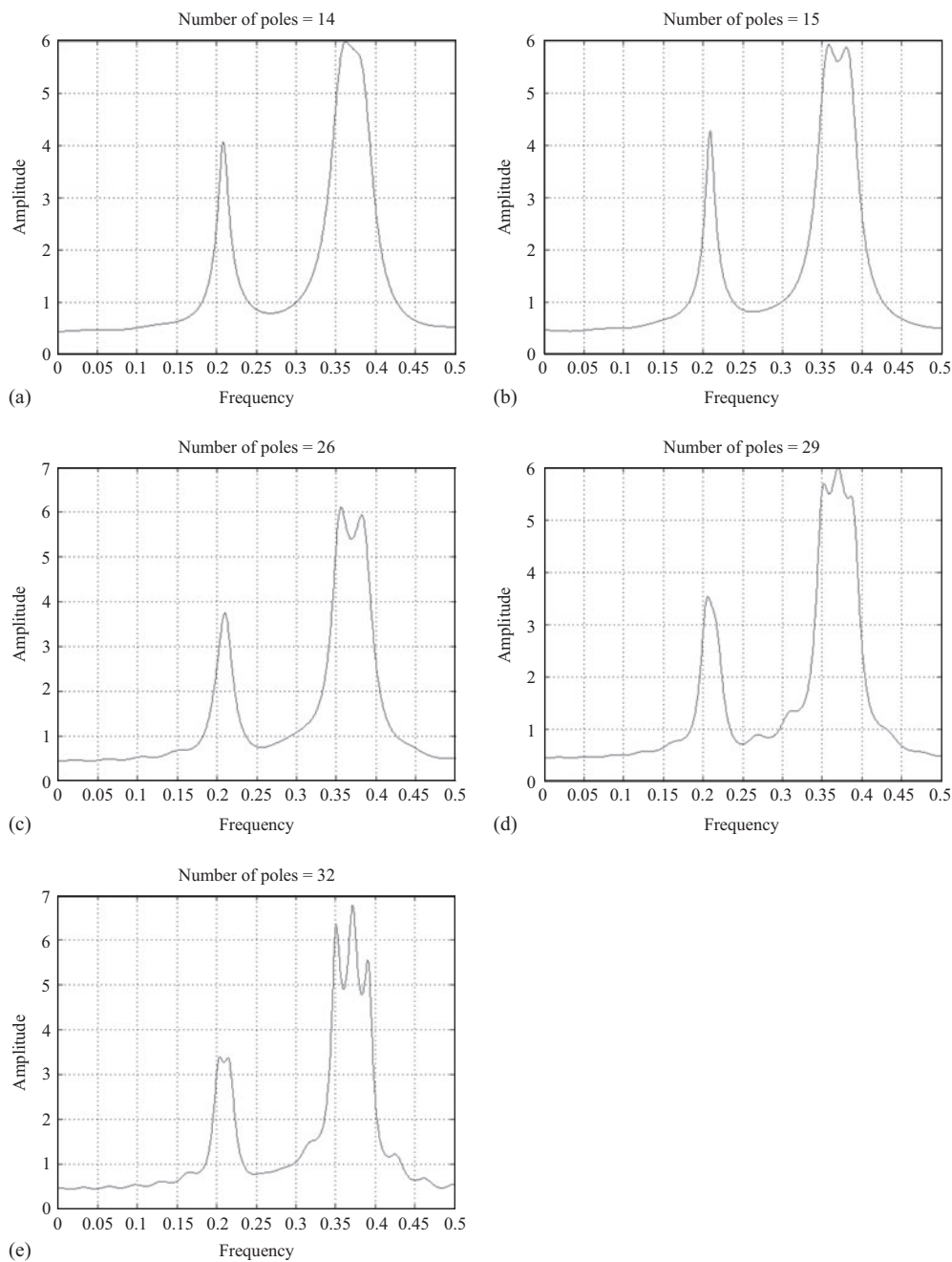


Figure 11.4 Spectrum generated by the Burg method: (a) $p = 14$, (b) $p = 15$, (c) $p = 26$, (d) $p = 29$, and (e) $p = 32$.

In these figures, five different p values (14, 15, 26, 29, and 32) are chosen. When $p < 14$, the two frequencies at 0.36 and 0.38 are difficult to identify. With $p = 15$ and 26, all three frequencies can be identified. When $p = 29$, an additional frequency is generated. When $p = 32$, five peaks can be found. This phenomenon is referred to as spectral splitting.

11.8 Order Selection [3, 4, 13, 27, 33, 34]

From the previous section, it is clear that when the wrong order in the Burg method is selected, the spectrum does not reflect the real input signal. If the order is too low, closely spaced frequencies cannot be detected. If the order is too high, spurious frequencies will be generated. These situations cannot be tolerated in receiver designs, thus it is important to select the correct order of the linear model, but this is a difficult task.

One intuitive approach is to use the recursive approach to find the coefficients and monitor the prediction error. If the data can be truly described by a finite-order linear model, when the correct order is reached the error will either reach a minimum or stay constant. However, this approach may not work. The prediction error may not converge or change monotonically. As a result, there is no easily detected minimum.

The four common methods to choose the order of the linear model are 1) the final prediction error (FPE), 2) the Akaike information criterion (AIC), 3) the criterion auto-regression transfer (CAT), and 4) the minimum description length (MDL). The results of the four approaches are shown in (11.52):

$$\begin{aligned}
 FPE_p &= \frac{\sigma_p^2(N + p + 1)}{(N - p - 1)} \\
 AIC_p &= N \ln(\sigma_p^2) + 2p \\
 CAT_p &= \frac{1}{N} \left(\sum_{i=1}^p \frac{1}{\sigma_i^2} \right) - \frac{1}{\sigma_p^2} \\
 MDL_p &= N \ln(\sigma_p^2) + p \ln(N)
 \end{aligned} \tag{11.52}$$

In order to obtain the p value, one of above equations will be minimized. For example, the AIC method determines the p value by minimizing AIC_p . If the data do not fit an AR model, the above approaches are not very useful.

Ulrych and Clayton [27] suggest an empirical approach, meaning that when

$$\frac{N}{3} < p < \frac{N}{2}, \tag{11.53}$$

satisfactory results can be obtained. Using this relation to check the result from the example in Figure 11.4, where $N = 32$, the result is $11 < p < 16$. It appears the lower bound is too low but the upper bound is satisfactory.

For digital receivers, the important issue is to determine the number of signals without generating spurious signals. The maximum number of anticipated signals can be considered as four sinusoidal waves. This is a common requirement for EW applications and is determined because of the possibility of pulse overlap. This requirement may create a different criterion for selecting the order of the linear prediction methods.

11.9 Prony's Method [3–5, 13, 35–40]

There are different ways to derive Prony's method. Only one approach is presented in this section, which is based on Hildebrand [35]. Prony's method solves a particular set of simultaneous nonlinear equations, which must fit in a certain form. An input signal consisting of only sinusoidal waves can be written in the desired form. In order to simplify the discussion, a simple example will be used to illustrate the basic idea and then a general case will be presented.

Let us assume that the input signal contains two complex sinusoidal waves without noise. The signal can be written as

$$x(t) = A_1 e^{j(2\pi f_1 t + \theta_1)} + A_2 e^{j(2\pi f_2 t + \theta_2)}, \quad (11.54)$$

where A_1 and A_2 , f_1 and f_2 , and θ_1 and θ_2 are the amplitude, the frequency, and the initial phase of the two signals, respectively. The amplitude and the frequency are of primary interest. The amplitudes and initial phases A_1 and θ_1 and A_2 and θ_2 can be combined into two unknowns. In order to solve for these unknowns, four equations are required. If these signals are digitized at $t = 0, 1, 2$, and 3 , the results are

$$\begin{aligned} x(0) &= A_1 e^{j\theta_1} + A_2 e^{j\theta_2} \equiv c_1 + c_2 \\ x(1) &= c_1 e^{j2\pi f_1} + c_2 e^{j2\pi f_2} \equiv c_1 z_1 + c_2 z_2 \\ x(2) &= c_1 e^{j2\pi f_1 2} + c_2 e^{j2\pi f_2 2} \equiv c_1 z_1^2 + c_2 z_2^2 \\ x(3) &= c_1 e^{j2\pi f_1 3} + c_2 e^{j2\pi f_2 3} \equiv c_1 z_1^3 + c_2 z_2^3, \end{aligned} \quad (11.55)$$

where

$$\begin{aligned} c_1 &= A_1 e^{j\theta_1} & c_2 &= A_2 e^{j\theta_2} \\ z_1 &= e^{j2\pi f_1} & z_2 &= e^{j2\pi f_2}. \end{aligned} \quad (11.56)$$

Prony developed a clever method to solve the above equation by converting this essentially nonlinear problem to a linear one. First, multiply the first three equations by a_2 , a_1 , and -1 , respectively, where a_2 and a_1 are unknown. The results are

$$\begin{aligned} a_2 x(0) &= a_2 (c_1 + c_2) \\ a_1 x(1) &= a_1 (c_1 z_1 + c_2 z_2) \\ -x(2) &= -(c_1 z_1^2 + c_2 z_2^2). \end{aligned} \quad (11.57)$$

Next, multiply the last three equations in (11.55) by a_2 , a_1 , and -1 , respectively, and the results are

$$\begin{aligned} a_2 x(1) &= a_2 (c_1 z_1 + c_2 z_2) \\ a_1 x(2) &= a_1 (c_1 z_1^2 + c_2 z_2^2) \\ -x(3) &= -(c_1 z_1^3 + c_2 z_2^3). \end{aligned} \quad (11.58)$$

Add the left side and the right side of (11.57) and set the result to zero:

$$-x(2) + a_1 x(1) + a_2 x(0) = c_1 (-z_1^2 + a_1 z_1 + a_2) + c_2 (-z_2^2 + a_1 z_2 + a_2) = 0. \quad (11.59)$$

Similar results can be obtained by adding both sides of (11.58) and setting the result to zero:

$$-x(3) + a_1 x(2) + a_2 x(1) = c_1 z_1 (-z_1^2 + a_1 z_1 + a_2) + c_2 z_2 (-z_2^2 + a_1 z_2 + a_2) = 0 \quad (11.60)$$

From these operations, two linear simultaneous equations can be formed:

$$\begin{aligned} -x(2) + a_1x(1) + a_2x(0) &= 0 \\ -x(3) + a_1x(2) + a_2x(1) &= 0. \end{aligned} \quad (11.61)$$

In these equations, a_1 and a_2 are the unknowns and the $x(i)$ with $i = 0, 1, 2$, and 3 are the known digitized data. Thus a_1 and a_2 can be solved from this equation. This is a similar equation to that shown in (11.9). Thus this operation leads to linear prediction.

In order to make (11.59) and (11.60) equal zero, the following relation must hold:

$$z_i^2 - a_1z_i - a_2 = 0 \quad \text{for } i = 1, 2. \quad (11.62)$$

Since a_1 and a_2 are known, by solving (11.61), one can solve for z_i . Once z_i is obtained, the frequency of the signal f_i can be obtained from (11.56) and the constant c_i can be found from (11.55).

The same approach can be extended to more than two signals. If there are M signals, the result can be written as

$$x(t) = \sum_{i=1}^M A_i e^{j(2\pi f_i t + \theta_i)}. \quad (11.63)$$

This equation can be written as

$$x(t) = \sum_{i=1}^M c_i e^{j2\pi f_i t}, \quad (11.64)$$

where $c_i = A_i e^{j\theta_i}$ is the complex amplitude. There are $2M$ unknowns in this equation, therefore $2M$ points of sampling data are needed and they can be written as

$$\begin{aligned} x(0) &= c_1 + c_2 + \cdots + c_M \\ x(1) &= c_1 z_1 + c_2 z_2 + \cdots + c_M z_M \\ &\quad \dots \\ x(2M-1) &= c_1 z_1^{2M-1} + c_2 z_2^{2M-1} + \cdots + c_M z_M^{2M-1}. \end{aligned} \quad (11.65)$$

To solve these equations, M unknowns (a_1 to a_M) are introduced and a set of linear equations of a_i can be formed.

The first equation can be obtained from the following procedure. Take the first $M+1$ equations, from $x(0)$ to $x(M)$, in the above equation and multiply the first one by a_M , the second one by a_{M-1} , and so forth. The last equation that starts with $x(M)$ is multiplied by -1 . These $M+1$ equations, after the above operation, will be

$$\begin{aligned} a_M x(0) &= a_M c_1 + a_M c_2 + \cdots + a_M c_M \\ a_{M-1} x(1) &= a_{M-1} c_1 z_1 + a_{M-1} c_2 z_2 + \cdots + a_{M-1} c_M z_M \\ &\quad \dots \\ -x(M) &= -c_1 z_1^M - c_2 z_2^M - \cdots - c_M z_M^M. \end{aligned} \quad (11.66)$$

Add both sides of the equations and set the results equal to zero. From the left side, the result is

$$a_M x(0) + a_{M-1} x(1) + \cdots + a_1 x(M) - x(M+1) = 0. \quad (11.67)$$

This is one of the equations needed to solve for the constant value of a_i . It is interesting to note that this is a linear prediction equation [i.e., the value of $x(M+1)$ can be written as a linear combination of $x(0)$ to $x(M)$ with an unknown coefficient of a_i].

The rest of the $M-1$ equations can be obtained in a similar way. For example, to obtain the second equation, another set of $M+1$ equations from $x(1)$ to $x(M+1)$ in (11.65) will be selected. Multiply the first equation by a_M , the second one by a_{M-1} , and so on, and the last one by -1 . When the left side of all the equations is added and set to zero, the result is

$$a_M x(1) + a_{M-1} x(2) + \cdots + a_1 x(M+1) - x(M+2) = 0. \quad (11.68)$$

The right side of the equations will be discussed separately. If one considers all the equations, the result is written in matrix form as

$$\begin{bmatrix} x(0) & x(1) & \cdots & x(M) \\ x(1) & x(2) & \cdots & x(M+1) \\ \vdots & \vdots & \ddots & \vdots \\ x(M) & x(M+1) & \cdots & x(2M-1) \end{bmatrix} \begin{bmatrix} a_M \\ a_{M-1} \\ \vdots \\ a_1 \end{bmatrix} = \begin{bmatrix} x(M+1) \\ x(M+2) \\ \vdots \\ x(2M) \end{bmatrix}. \quad (11.69)$$

From the equation, the coefficients a_i can be solved linearly.

Now let us look at the right side of (11.66). The result is

$$\begin{aligned} c_1 (a_M + a_{M-1} z_1 + \cdots + a_1 z_1^{M-1} - z_1^M) + c_2 (a_M + a_{M-1} z_2 + \cdots + a_1 z_2^{M-1} - z_2^M) + \cdots \\ + c_M (a_M + a_{M-1} z_M + \cdots + a_1 z_M^{M-1} - z_M^M) = 0. \end{aligned} \quad (11.70)$$

This equation can be written in matrix form as

$$\begin{bmatrix} c_1 & c_2 & \cdots & c_M \\ c_1 z_1 & c_2 z_2 & \cdots & c_M z_M \\ \vdots & \vdots & \ddots & \vdots \\ c_1 z_1^{M-1} & c_2 z_2^{M-1} & \cdots & c_M z_M^{M-1} \end{bmatrix} \begin{bmatrix} a_M + a_{M-1} z_1 + \cdots - z_1^M \\ a_M + a_{M-1} z_2 + \cdots - z_2^M \\ \vdots \\ a_M + a_{M-1} z_M + \cdots - z_M^M \end{bmatrix} = 0. \quad (11.71)$$

To fulfill this equation, the following relation must be true:

$$a_M + a_{M-1} z_i + \cdots + a_1 z_i^{M-1} - z_i^M = 0 \quad i = 1, 2, \dots, M-1. \quad (11.72)$$

Or in a slightly different form,

$$z_i^M - a_1 z_i^{M-1} - \cdots - a_{M-1} z_i - a_M = 0, \quad (11.73)$$

where z_i is the root of the equation. These are all the required equations for the Prony's method. In these equations, $x(0), x(1), \dots, x(2M)$ are the measured values.

For the above discussion, the Prony's method can be summarized into four steps as follows.

1. From (11.69), the coefficients $a_i, i = 1, 2, \dots, M$ can be obtained.
2. From (11.73), $z_i, i = 1, 2, \dots, M$ can be obtained.

3. From (11.56), $z_i = e^{j2\pi fi}$, the frequency of the input signal, can be obtained.
4. The amplitude and the initial phase of the input signals, which are expressed as $c_i = A_i e^{j\theta_i}$, can be found from (11.65) once z_i is obtained.

When the signal-to-noise (S/N) ratio is high, Prony's method can produce very accurate results. However, when the input S/N is low, the error generated through the Prony's method can be rather high.

A different approach to derive the Prony's method through the z transform can be found in Kumaresan [40].

11.10 Prony's Method Using the Least Squares Approach [3–5, 13, 19, 35–40]

To improve the performance of Prony's method, one can use more data points. The data points are used in the least squares approach to generate the desired a_i coefficients. Let us use a simple example to demonstrate this idea, then a general case will be presented.

Assume that there are only two signals that require a minimum of four complex data points. However, in order to obtain better accuracy, six data points are acquired. The result is

$$\begin{aligned}
 x(0) &= c_1 + c_2 \\
 x(1) &= c_1 z_1 + c_2 z_2 \\
 x(2) &= c_1 z_1^2 + c_2 z_2^2 \\
 x(3) &= c_1 z_1^3 + c_2 z_2^3 \\
 x(4) &= c_1 z_1^4 + c_2 z_2^4 \\
 x(5) &= c_1 z_1^5 + c_2 z_2^5
 \end{aligned} \tag{11.74}$$

Since there are only two signals (or Prony's method is limited to a second-order equation), only two coefficients a_1 and a_2 will be introduced to generate the linear functions, as mentioned in the previous section. The results are

$$\begin{aligned}
 x(2) &= a_1 x(1) + a_2 x(0) \\
 x(3) &= a_1 x(2) + a_2 x(1) \\
 x(4) &= a_1 x(3) + a_2 x(2) \\
 x(5) &= a_1 x(4) + a_2 x(3).
 \end{aligned} \tag{11.75}$$

There are four linear equations, but with only two unknowns. To solve these equations, the least squares method is used and the result is [4, 19]

$$\begin{aligned}
 &\begin{bmatrix} x^*(1) & x^*(2) & x^*(3) & x^*(4) \\ x^*(0) & x^*(1) & x^*(2) & x^*(3) \end{bmatrix} \begin{bmatrix} x(1) & x(0) \\ x(2) & x(1) \\ x(3) & x(2) \\ x(4) & x(3) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \\
 &= \begin{bmatrix} x^*(1) & x^*(2) & x^*(3) & x^*(4) \\ x^*(0) & x^*(1) & x^*(2) & x^*(3) \end{bmatrix} \begin{bmatrix} x(2) \\ x(3) \\ x(4) \\ x(5) \end{bmatrix}.
 \end{aligned} \tag{11.76}$$

In this equation, all six known values are used to solve for the two unknowns a_1 and a_2 . Once the a_i are obtained, the corresponding z values can be obtained from (11.72) and the input frequency can be obtained from the definition of z . The only difference is in the calculation of a_i . All the other steps are the same as mentioned at the end of the previous section.

In general, assume that there are N known $x(0)$ to $x(N-1)$ and M signals in order to solve for the input signal frequencies $N \geq 2M$. If $N > 2M$, the least squares method can be used to solve for a_i and the result is

$$\begin{aligned}
 & \begin{bmatrix} x^*(M-1) & x^*(M) & \cdots & x^*(N-2) \\ x^*(M-2) & x^*(M-1) & \cdots & x^*(N-3) \\ \cdots & \cdots & \cdots & \cdots \\ x^*(0) & x^*(1) & \cdots & x^*(N-M-1) \end{bmatrix} \\
 & \times \begin{bmatrix} x(M-1) & x(M-2) & \cdots & x(0) \\ x(M) & x(M-1) & \cdots & x(1) \\ \cdots & \cdots & \cdots & \cdots \\ x(N-2) & x(N-3) & \cdots & x(N-M-1) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix} \\
 & = \begin{bmatrix} x^*(M-1) & x^*(M) & \cdots & x^*(N-2) \\ x^*(M-2) & x^*(M-1) & \cdots & x^*(N-3) \\ \cdots & \cdots & \cdots & \cdots \\ x^*(0) & x^*(1) & \cdots & x^*(N-M-1) \end{bmatrix} \begin{bmatrix} x(M) \\ x(M+1) \\ \vdots \\ x(N-1) \end{bmatrix}. \quad (11.77)
 \end{aligned}$$

After a_i ($i = 1, 2, \dots, M$) are obtained, one nonlinear equation of z (11.72) can be solved to obtain z and the individual frequency of the input signal can be found. The performance of the least squares method should produce a better result since more data points are used in the calculation.

11.11 Eigenvectors and Eigenvalues [3–5]

In this section the concepts of eigendecomposition, eigenvectors, and eigenvalues is introduced. These concepts will be used in later sections to estimate frequencies.

If A is a given square matrix, a constant λ and a vector $\mathbf{X}$ can be found such that

$$A \cdot \mathbf{X} = \lambda \mathbf{X}, \quad (11.78)$$

where λ is called an eigenvalue and $\mathbf{X}$ is called its corresponding eigenvector. This process is called the eigendecomposition of A . To find λ and $\mathbf{X}$, the above equation can be written as

$$(A - \lambda I) \cdot \mathbf{X} = 0, \quad (11.79)$$

where I is the identity matrix. In order to have a nontrivial solution (i.e., $\mathbf{X} \neq 0$), the determinant of $A - \lambda I$ should equal zero. For example, if

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad \text{then } A - \lambda I = \begin{vmatrix} 1 - \lambda & 2 \\ 3 & 4 - \lambda \end{vmatrix} = (1 - \lambda)(4 - \lambda) - 6 = 0. \quad (11.80)$$

The eigenvalues $\lambda_1 = -0.3723$ and $\lambda_2 = 5.3723$, which can be found using the MATLAB[®] “eig” command. With each eigenvalue there is an eigenvector. The corresponding eigenvector $\mathbf{X}_i = [x_{i1} \ x_{i2}]^T$, where superscript T , representing the transpose of a matrix, can be solved from (11.78). The eigenvectors are found from MATLAB[®] as $\mathbf{X}_1 = [-0.8246 \ 0.5658]^T$ and $\mathbf{X}_2 = [-0.4160 \ -0.9094]^T$ with the restriction $x_{i1}^2 + x_{i2}^2 = 1$, $i = 1, 2$.

Let us use a simple example to demonstrate the application of eigenvectors and eigenvalues. If the input signal is

$$x(i) = Ae^{j(2\pi fi + \phi)} + u(i), \quad (11.81)$$

where A , f , and ϕ are the amplitude, frequency, and initial phase of a sine wave, respectively, and $u(i)$ is white Gaussian noise, the autocorrelation with lag k is

$$\begin{aligned} R(k) &= E[x(i+k)x(i)^*] \\ &= E[\{Ae^{j[2\pi f(i+k) + \phi]} + u(i+k)\}\{Ae^{-j[2\pi f(i) + \phi]} + u(i)^*\}], \end{aligned} \quad (11.82)$$

where $E[\]$ is the expectation value. Since the cross product of signal and noise is zero and the noise is uncorrelated, the result will be

$$R(k) = A^2 e^{j2\pi fk} + \sigma^2 \delta_{0k}, \quad (11.83)$$

where σ^2 is the variance of the noise and δ_{0k} is the Kronecker delta, which has the following property:

$$\begin{aligned} \delta_{ij} &= 1 \quad i = j \\ &= 0 \quad i \neq j. \end{aligned} \quad (11.84)$$

The correlation matrix R can be found as

$$\begin{bmatrix} R(0) & R(-1) \\ R(1) & R(0) \end{bmatrix} = \begin{bmatrix} A^2 + \sigma^2 & A^2 e^{-j2\pi f} \\ A^2 e^{j2\pi f} & A^2 + \sigma^2 \end{bmatrix}. \quad (11.85)$$

The eigenvalue and eigenvector of R can be written as

$$\begin{bmatrix} R(0) & R(-1) \\ R(1) & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a \end{bmatrix} = \lambda \begin{bmatrix} 1 \\ a \end{bmatrix}. \quad (11.86)$$

The minimum eigenvalue corresponds to the noise power and this can be proven as follows. Multiplying both sides of the above equation by $[1 \ a^*]$, the result is

$$[1 \ a^*] \begin{bmatrix} R(0) & R(-1) \\ R(1) & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a \end{bmatrix} = \lambda [1 \ a^*] \begin{bmatrix} 1 \\ a \end{bmatrix} = \lambda (1 + |a|^2). \quad (11.87)$$

Substitute the $R(k)$ values into the left side of above equation and the result is

$$\begin{aligned}
 \begin{bmatrix} 1 & a^* \end{bmatrix} \begin{bmatrix} R(0) & R(-1) \\ R(1) & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a \end{bmatrix} &= \begin{bmatrix} R(0) + a^*R(1) & R(-1) + a^*R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a \end{bmatrix} \\
 &= R(0) + a^*R(1) + aR(-1) + aa^*R(0) \\
 &= A^2 + \sigma^2 + A^2 a^* e^{j2\pi f} + A^2 a e^{-j2\pi f} + |a|^2 A^2 + \sigma^2 |a|^2 \\
 &= \sigma^2 (1 + |a|^2) + A^2 (1 + a^* e^{j2\pi f} + a e^{-j2\pi f} + |a|^2) \\
 &= \sigma^2 (1 + |a|^2) + A^2 |1 + a e^{-j2\pi f}|^2.
 \end{aligned} \tag{11.88}$$

Comparing (11.87) and (11.88), one can obtain the following results:

$$\lambda (1 + |a|^2) = \sigma^2 (1 + |a|^2) + A^2 |1 + a e^{-j2\pi f}|^2. \tag{11.89}$$

The right side of this equation contains two nonnegative terms. The minimum value occurs when $A^2 |1 + a e^{-j2\pi f}|^2 = 0$, thus one can conclude that the minimum eigenvalue is

$$\lambda_{\min}^2 = \sigma^2. \tag{11.90}$$

This proves that the minimum eigenvalue is equal to the noise power. This simple example can be generalized to multiple signal cases.

11.12 MUSIC Method [5, 41–43]

The MUSIC method was developed by Schmidt in 1981. MUSIC stands for Multiple Signal Classification. The basic idea of the MUSIC method is to separate signal from noise through eigenvalue decomposition. The approach is to find all the eigenvalues and eigenvectors of the R matrix and the result can be written as

$$\begin{aligned}
 \begin{bmatrix} R(0) & R(1)^* & \cdots & R(p)^* \\ R(1) & R(0) & \cdots & R(p-1)^* \\ \cdots & \cdots & \cdots & \cdots \\ R(p) & R(p-1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} V_{00} & V_{01} & \cdots & V_{0p} \\ V_{10} & V_{11} & \cdots & V_{1p} \\ \cdots & \cdots & \cdots & \cdots \\ V_{p0} & V_{p1} & \cdots & V_{pp} \end{bmatrix} \\
 = \begin{bmatrix} \lambda_0 V_{00} & \lambda_1 V_{01} & \cdots & \lambda_p V_{0p} \\ \lambda_0 V_{10} & \lambda_1 V_{11} & \cdots & \lambda_p V_{1p} \\ \cdots & \cdots & \cdots & \cdots \\ \lambda_0 V_{p0} & \lambda_1 V_{p1} & \cdots & \lambda_p V_{pp} \end{bmatrix}.
 \end{aligned} \tag{11.91}$$

The first subscript represents the individual element of the matrix. The second subscript represents the numerical order to the eigenvector. For example, the first eigenvector is the

first column of the $\mathbf{V}$ matrix and the second eigenvector is the second column of the $\mathbf{V}$ matrix, and so on.

The eigenvalues are represented by λ_i . If there are M signals, there are M eigenvalues $\lambda_0 \dots \lambda_{M-1}$ corresponding to the signals and the rest of the eigenvalues $\lambda_M \dots \lambda_P$ corresponding to noise. These eigenvalues can be listed in decreasing order as $\lambda_0 > \lambda_1 > \dots > \lambda_{M-1} > \lambda_M = \dots = \lambda_P = \sigma^2$.

The eigenvectors $\mathbf{V}$ with eigenvalues $\lambda_0 \dots \lambda_{M-1}$ are referenced as the signal subspace, which is represented by $\mathbf{V}_s$. Eigenvectors $\mathbf{V}$ with eigenvalues $\lambda_M \dots \lambda_P$ are referenced as the noise subspace, which is represented by $\mathbf{V}_n$. These two subspaces, the $\mathbf{V}_s$ and $\mathbf{V}_n$ matrices, can be written as

$$\mathbf{V}_s = \begin{bmatrix} V_{00} & V_{01} & \cdots & V_{0M-1} \\ V_{10} & V_{11} & \cdots & V_{1M-1} \\ \cdots & \cdots & \cdots & \cdots \\ V_{p0} & V_{p1} & \cdots & V_{pM-1} \end{bmatrix} \quad \mathbf{V}_n = \begin{bmatrix} V_{0M} & V_{0M+1} & \cdots & V_{0p} \\ V_{1M} & V_{1M+1} & \cdots & V_{1p} \\ \cdots & \cdots & \cdots & \cdots \\ V_{pM} & V_{pM+1} & \cdots & V_{pp} \end{bmatrix} \quad (11.92)$$

The signal subspace is orthogonal to the noise subspace.

The basic idea of the MUSIC method is to use the orthogonal property of the signal and noise subspaces. Assume that the input vector $\mathbf{s}$ is

$$\mathbf{s} = [1 \quad e^{-j2\pi f} \quad \dots \quad e^{-j2\pi(N-1)f}]. \quad (11.93)$$

This vector is orthogonal to the noise subspace. One can write a function P_{MUS} of variable frequency f as

$$P_{\text{MUS}}(f) = \frac{1}{\mathbf{s} \mathbf{V}_n \mathbf{V}_n^H \mathbf{s}^H} \quad (11.94)$$

since $\mathbf{s}$ is a function of f . If the value of f is equal to the frequency of the input signal, the denominator of $P_{\text{MUS}}(f)$ is zero (actually it has a minimum) because the signal $\mathbf{s}$ is orthogonal to the noise subspace $\mathbf{V}_n$. Thus, by plotting the $P_{\text{MUS}}(f)$ as a function f , the peaks represent the input frequencies. One can use a root method to find the peak of the $P_{\text{MUS}}(f)$ and this method is discussed in Therrien [5].

The following is a summary of the MUSIC method:

1. From the input data $x(0), x(1), \dots, x(N-1)$, create the R matrix. The common approach to form the R matrix is through the modified covariance method as shown in (11.47). In this step, the value of p should be selected. Since p should be larger than the maximum number of signals M_{max} , a p value greater than M_{max} is required. If there are N points of data, $p = 2N/3$ seems to be a good selection.
2. Use eigendecomposition on the R matrix to find the eigenvalues λ_i and the $\mathbf{V}$ matrix. If the number M of the input signals is known, the first large M eigenvalues will be chosen. The rest are the noise eigenvalues. However, if the number of input signals is unknown, one must examine the eigenvalues to determine the number of signals. Large eigenvalues correspond to signals and small eigenvalues correspond to noise. However, the determination is somewhat subjective and S/N dependent. If two input signals are close in

frequency, sometimes it is difficult to separate the signal eigenvalues from the noise eigenvalues. Once the eigenvalues are selected, the corresponding noise eigenvectors $\mathbf{V}_n$ can be obtained from (11.92).

3. Use (11.94) to find the plot of $P_{\text{MUS}}(f)$ as a function of f . The peaks in the $P_{\text{MUS}}(f)$ plot represent the input frequencies.

A computer program is used to perform the MUSIC calculation. This program does not compare the amplitudes of the eigenvalues to determine the number of signals. This program requires two inputs: the number of signals M and the order of the filter p (or the covariance matrix). The order of the filter must be equal to or greater than $2M + 1$.

The result of the MUSIC method is shown in Figure 11.5. The data used are the same as in previous examples. In Figures 11.5(a) and 11.5(b), $M = 4$ and $p = 9$ and 27, respectively. The results show the desired three peaks. These peaks are sharper than those obtained from the Burg method, as shown in Figure 11.4. In Figure 11.5(c), $M = 4$ and $p = 28$, and the result does not represent the true input signal. In Figure 11.5(d), $M = 6$ and $p = 13$, and the result shows four peaks. The lowest peak is a spurious response.

From these results, it appears that the order of p is not very critical since in Figures 11.5(a) and 11.5(b) the p value varies from 9 to 27 and the results are satisfactory. Thus $p = 2N/3 \approx 21$, ($N = 32$) seems to be a reasonable choice. However, the selection of the number of signals seems to be very important. If the number is too high, spurious responses will appear. If the number is too low, the desired peak will not appear.

In a wideband receiver, it is very important to determine the number of input signals. In the eigendecomposition methods, one must develop an effective method to determine the number of signals.

11.13 ESPRIT Method [5, 44–48]

The ESPRIT method was first introduced by Paulraj *et al.* [44]. ESPRIT stands for Estimation of Signal Parameters via Rotational Invariance Techniques. In the original approach, sometimes the R matrix was ill-conditioned and the eigenvalues were not accurate. Later, several approaches were used to improve the performance, such as the total least squares ESPRIT [46, 48] and the procrustes rotation [47]. In this section, the original ESPRIT is presented and the discussion emphasizes the procedure.

1. Assume that there are N points of data represented by $x(n)$, where $n = 0$ to $N - 1$. These data points can be divided into two groups, each consisting of $N - 1$ points. The first group G_1 contains data from $x(0)$ to $x(N - 2)$ and the second group G_2 contains data from $x(1)$ to $x(N - 1)$, which can be written as

$$\begin{aligned} G_1 &= x(0), x(1), x(2), \dots, x(N - 2) \\ G_2 &= x(1), x(2), x(3), \dots, x(N - 1). \end{aligned} \quad (11.95)$$

These data will be used to make two R matrices, R_{yy} and R_{yz} , through the covariance approach. The matrix R_{yy} is obtained from data in G_1 and R_{yz} is obtained from data in G_1 and G_2 . In obtaining R_{yz} , the first matrix in (11.30) uses data from G_1 and the second matrix uses data from G_2 .

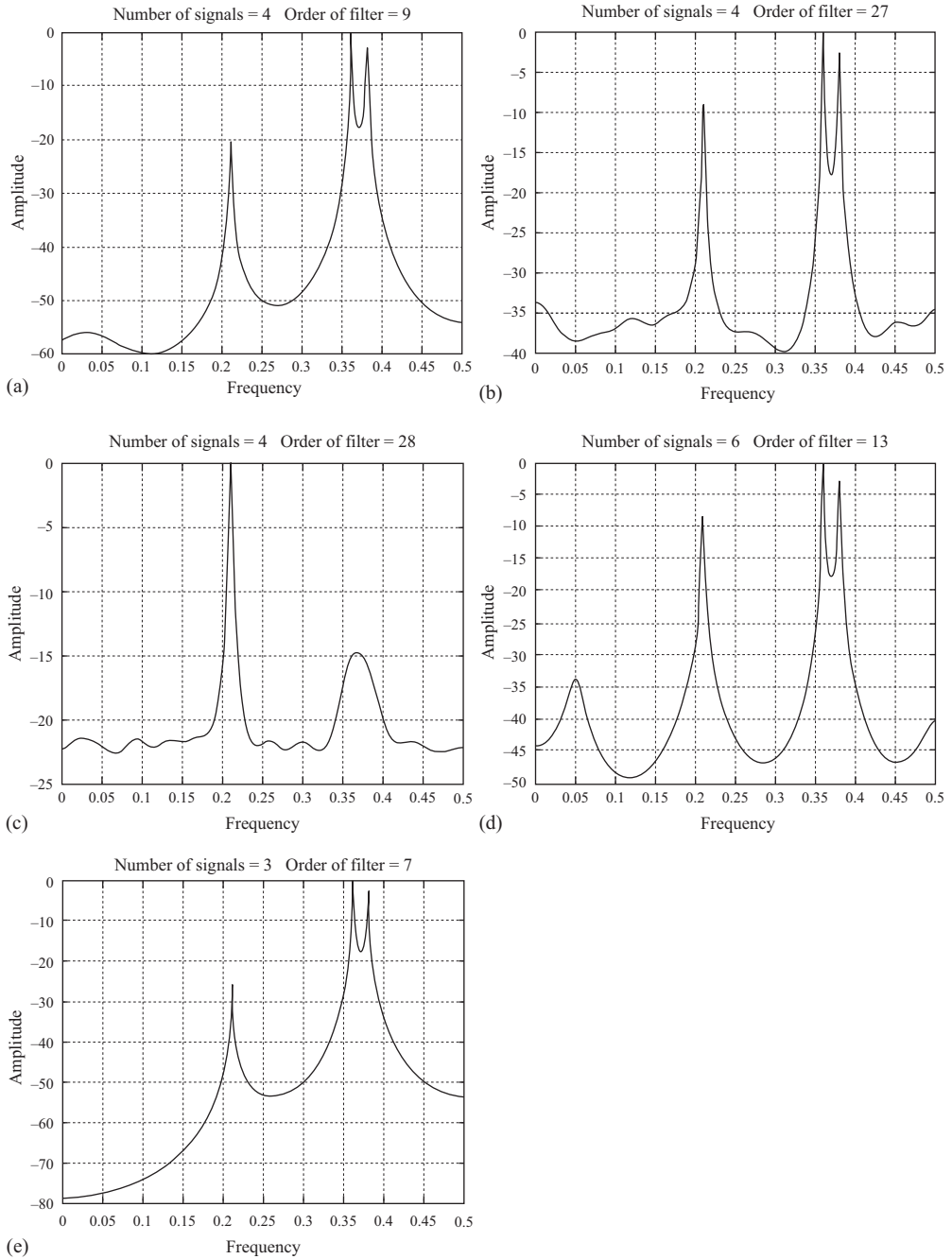


Figure 11.5 Result from the MUSIC method: (a) $M = 4, p = 9$; (b) $M = 4, p = 27$; (c) $M = 4, p = 28$; (d) $M = 6, p = 13$; and (e) $M = 3, p = 7$.

As an example, if N is even and the order $p = N/2 - 1$, the data can be arranged into the following two matrices y and z as

$$y = \begin{bmatrix} x(0) & x(1) & \cdots & x\left(\frac{N}{2} - 1\right) \\ x(1) & x(2) & \cdots & x\left(\frac{N}{2}\right) \\ \cdots & \cdots & \cdots & \cdots \\ x\left(\frac{N}{2} - 1\right) & x\left(\frac{N}{2}\right) & \cdots & x(N - 2) \end{bmatrix} \quad (11.96)$$

$$z = \begin{bmatrix} x(1) & x(2) & \cdots & x\left(\frac{N}{2}\right) \\ x(2) & x(3) & \cdots & x\left(\frac{N}{2} + 1\right) \\ \cdots & \cdots & \cdots & \cdots \\ x\left(\frac{N}{2}\right) & x\left(\frac{N}{2} + 1\right) & \cdots & x(N - 1) \end{bmatrix}.$$

From these two matrices, one can form two R matrices as

$$R_{yy} = yy^H, \quad R_{yz} = yz^H, \quad (11.97)$$

where the superscript H represents the hermitian of a matrix, which is the transpose and conjugate of the matrix. The value R_{yy} can be considered as the autocorrelation matrix and R_{yz} as the cross-correlation matrix.

2. Perform eigendecomposition on R_{yy} to find the eigenvectors and eigenvalues as

$$R_{yy}\mathbf{e}' = \lambda'\mathbf{e}', \quad (11.98)$$

where $\mathbf{e}'$ and λ' are the eigenvectors and eigenvalues, respectively. From the distribution of eigenvalues, the number of signals can be determined. Large eigenvalues correspond to signal and small eigenvalues correspond to noise. This is the same argument used in the MUSIC method.

3. Define two matrices—the I and D matrices. Both matrices have a dimension of $N/2 \times N/2$ to match the dimensions of R_{yy} and R_{yz} in (11.96). The I matrix is a diagonal matrix. The D matrix has only one diagonal line of 1s, all the other data are 0s. The diagonal line of 1s is under the main diagonal axis. These two matrices are

$$I = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix} \quad (11.99)$$

4. Form two more R matrices, R_s and R_t ,

$$\begin{aligned} R_s &= R_{yy} - \lambda_{\min} I \\ R_t &= R_{yz} - \lambda_{\min} D, \end{aligned} \quad (11.100)$$

where $\lambda_{\min}$ is the smallest eigenvalue from step 2.

5. Find the generalized eigendecomposition of R_s and R_t as

$$R_s \mathbf{e} = \lambda R_t \mathbf{e}, \quad (11.101)$$

where $\mathbf{e}$ is the eigenvector and λ is the eigenvalue.

6. Find the input frequencies. The procedure is to find the λ values close to the unit circle. Once these λ_i values are found, the input frequency can be found as

$$f_i = \frac{1}{2\pi} \tan^{-1} \left(\frac{Im \lambda_i}{Re \lambda_i} \right), \quad (11.102)$$

where Im and Re represent the imaginary and real parts of λ_i .

In this approach, the eigendecomposition is performed twice, which is computationally intensive. One major advantage is that once the eigenvalues of (11.101) are found, only the eigenvalues close to the unit circle will be selected and the corresponding signal frequencies can be found. It is not necessary to search the entire frequency range as in the AR and MUSIC methods. This method has the same problem as mentioned before, which is to determine the number of signals from the values of the eigenvalues obtained from (11.102). An ESPRIT program with $p = N/2$ is provided in Appendix 11.E.

11.14 Minimum Norm Method [49, 50]

The minimum norm method was derived by Kumaresan and Tufts [50]. This method and the MUSIC method are somewhat alike. The basic idea is to find a vector $\mathbf{d}$ that is a linear combination of eigenvectors in the noise subspace. This vector $\mathbf{d}$ can be written as

$$\mathbf{d} = [\mathbf{d}_0 \quad \mathbf{d}_1 \quad \cdots \quad \mathbf{d}_P]^T \equiv [1 \quad \mathbf{d}_1 \quad \cdots \quad \mathbf{d}_P]^T, \quad (11.103)$$

where the superscript T represents a transpose of a matrix. In this equation $\mathbf{d}_0$ is assigned to be unity ($\mathbf{d}_0 \equiv 1$). The square of the norm of this equation is

$$|\mathbf{d}|^2 = \sum_{i=0}^P \mathbf{d}_i^2. \quad (11.104)$$

This method minimizes this norm and that is why it is named as such.

The procedure of the minimum norm method is presented here. The first two steps are identical to the MUSIC method.

1. The first step is to find the eigenvectors $\mathbf{V}$ of the R matrix. The common approach to form the R matrix is through the modified covariance method as shown in (11.47).
2. Perform eigendecomposition on the R matrix to find the eigenvalues λ_i and the $\mathbf{V}$ matrix. If the number M of the input signals is known, the first large M eigenvalues will be chosen. The rest are the noise eigenvalues. However, if the number of input signals is unknown,

one must examine the eigenvalues to determine the number of signals. Large eigenvalues correspond to signals and small eigenvalues correspond to noise. The vector $\mathbf{d}$ can be found from either the noise subspace $\mathbf{V}_n$ or from the signal noise subspace $\mathbf{V}_s$.

3. Find the vector $\mathbf{d}$ from the noise subspace $\mathbf{V}_n$. The noise subspace $\mathbf{V}_n$ can be written as

$$\mathbf{V}_n = \begin{bmatrix} V_{0M} & V_{0M+1} & \cdots & V_{0p} \\ V_{1M} & V_{1M+1} & \cdots & V_{1p} \\ \cdots & \cdots & \cdots & \cdots \\ V_{pM} & V_{pM+1} & \cdots & V_{pp} \end{bmatrix} \equiv \begin{bmatrix} c^H \\ \mathbf{V}'_n \end{bmatrix}, \quad (11.105)$$

where the superscript H is the hermitian, c^H represents the first row of $\mathbf{V}_n$, and $\mathbf{V}'_n$ represents the rest of the $\mathbf{V}_n$ matrix, which can be written explicitly as

$$c^H = [V_{0M} \quad V_{0M+1} \quad \cdots \quad V_{0p}] \mathbf{V}'_n = \begin{bmatrix} V_{1M} & V_{1M+1} & \cdots & V_{0p} \\ \cdots & \cdots & \cdots & \cdots \\ V_{pM} & V_{pM+1} & \cdots & V_{pp} \end{bmatrix}. \quad (11.106)$$

4. From c_H and $\mathbf{V}'_n$, the $\mathbf{d}$ vector can be found as

$$\mathbf{d} = \begin{bmatrix} 1 \\ \cdots \\ \mathbf{V}'_n c / (c_H c) \end{bmatrix}. \quad (11.107)$$

5. The $\mathbf{d}$ vector can also be obtained from the signal subspace. The signal subspace $\mathbf{V}_s$ can be written as

$$\mathbf{V}_s = \begin{bmatrix} V_{00} & V_{01} & \cdots & V_{0M-1} \\ V_{10} & V_{11} & \cdots & V_{1M-1} \\ \cdots & \cdots & \cdots & \cdots \\ V_{p0} & V_{p1} & \cdots & V_{pM-1} \end{bmatrix} \equiv \begin{bmatrix} g^H \\ \mathbf{V}'_s \end{bmatrix}, \quad (11.108)$$

where g^H is the first row of the $\mathbf{V}_s$ matrix and $\mathbf{V}'_s$ is the rest of the matrix. These two matrices can be written as

$$g^H = [V_{00} \quad V_{01} \quad \cdots \quad V_{0M-1}] \mathbf{V}'_s = \begin{bmatrix} V_{10} & V_{11} & \cdots & V_{1M-1} \\ \cdots & \cdots & \cdots & \cdots \\ V_{p0} & V_{p1} & \cdots & V_{pM-1} \end{bmatrix}. \quad (11.109)$$

6. The $\mathbf{d}$ vector can be obtained as

$$\mathbf{d} = \begin{bmatrix} 1 \\ \cdots \\ -\mathbf{V}'_s g / (1 - g^H g) \end{bmatrix}. \quad (11.110)$$

It should be noted that if steps 3 and 4 are carried out, then steps 5 and 6 are not necessary. The converse is also true (i.e., steps 5 and 6 can replace steps 3 and 4).

7. Once the $\mathbf{d}$ vector is obtained, a function $P_{MN}(f)$ can be defined as

$$P_{MN}(f) = \frac{1}{\mathbf{s}^H \mathbf{d} \mathbf{d}^H \mathbf{s}}, \quad (11.111)$$

since $\mathbf{s}$ is a function of the input frequency f as shown in (11.93) and $P_{MN}(f)$ is a function of f . Plot $P_{MN}(f)$; the peaks represent the input frequencies. This step is similar to the MUSIC method. One can also use a root method to find the frequencies of $P_{MN}(f)$.

The result of the minimum norm is shown in Figure 11.6. The data used are the same as in previous examples. In this figure, the number of signals is selected as 4 and the p value is chosen as 20. The program uses the noise subspace matrix to find the vector $\mathbf{d}$. There are some low-level peaks in this approach.

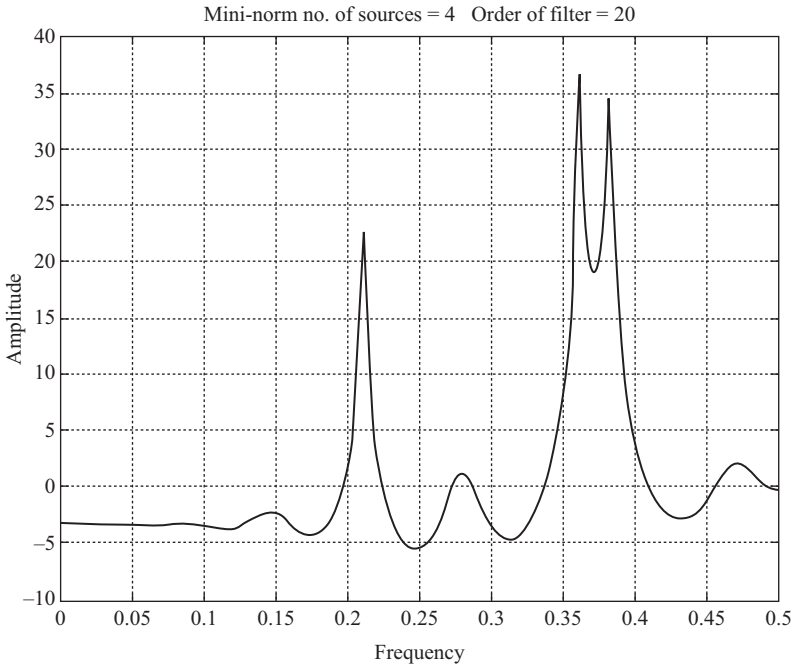


Figure 11.6 Frequency response from the minimum norm method.

11.15 Minimum Norm Method with DFT [51]

This method was derived by Shaw and Xia and this discussion is based their work [51]. It can be considered as a modified approach to the minimum norm discussed in the previous section. The main difference is to replace the eigendecomposition with a DFT. The DFT might be easier to carry out than the eigendecomposition from a hardware point of view. The difference is to replace steps 3 and 4 or 5 and 6 in the previous section. Instead of repeating all the steps in the previous sections, only the changes will be discussed here.

The correlation matrix R has a dimension of $(1 + p) \times (1 + p)$. The following steps are used to obtain the minimum norm.

1. Let us define a Fourier matrix E of dimension of $(1 + p) \times (1 + p)$ as

$$E = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{\frac{j2\pi}{p+1}} & e^{\frac{j4\pi}{p+1}} & \cdots & e^{\frac{j2\pi p}{p+1}} \\ 1 & e^{\frac{j4\pi}{p+1}} & e^{\frac{j8\pi}{p+1}} & \cdots & e^{\frac{j4\pi p}{p+1}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & e^{\frac{j2^p\pi}{p+1}} & e^{\frac{j2 \cdot 2^p\pi}{p+1}} & \cdots & e^{\frac{j2^p\pi p}{p+1}} \end{bmatrix}. \quad (11.112)$$

From this result, a matrix $\mathbf{F}$ is constructed as

$$\mathbf{F} = RE, \quad (11.113)$$

where R is the autocorrelation matrix. This $\mathbf{F}$ matrix has a dimension of $(1 + p) \times (1 + p)$ also.

2. Each column of the $\mathbf{F}$ matrix is considered as a vector, so the $\mathbf{F}$ matrix can be written as

$$\mathbf{F} = \begin{bmatrix} F_{00} & F_{01} & \cdots & F_{0p} \\ F_{10} & F_{11} & \cdots & F_{1p} \\ \cdots & \cdots & \cdots & \cdots \\ F_{p0} & F_{p1} & \cdots & F_{pp} \end{bmatrix} \equiv [f_0 \quad f_1 \quad \cdots \quad f_p], \quad (11.114)$$

where

$$f_i = [F_{0i} \quad F_{1i} \quad \cdots \quad F_{pi}]^T \quad (11.115)$$

and $(i = 0, 1, \dots, p)$.

3. Find the norm of all the $\mathbf{f}_i$ vectors and represent them by $|\mathbf{f}_i|$. The $|\mathbf{f}_i|$ with large values represent the signals. One has to determine the number of signals from the amplitude of $|\mathbf{f}_i|$. If one selects M signals, the largest M column in (11.114) will be used as the signal matrix. Thus, as

$$\mathbf{V}_s = \begin{bmatrix} F'_{00} & F'_{01} & \cdots & F'_{0,M-1} \\ F'_{10} & F'_{11} & \cdots & F'_{1,M-1} \\ \cdots & \cdots & \cdots & \cdots \\ F'_{p0} & F'_{p1} & \cdots & F'_{p,M-1} \end{bmatrix}, \quad (11.116)$$

where $M < p$, it is important to note that the columns of the $\mathbf{V}_s$ matrix are the columns of matrix $\mathbf{F}$ with largest $M |\mathbf{f}_i|$ values. For example, if there are two large norms and they are elements 2 and 4 from (11.114) and (11.115), the signal subspace matrix is

$$\mathbf{V}_s = \begin{bmatrix} F_{01} & F_{03} \\ F_{11} & F_{13} \\ \cdots & \cdots \\ F_{p1} & F_{p3} \end{bmatrix}. \quad (11.117)$$

Once $\mathbf{V}_s$ is obtained, (11.108) to (11.110) can be used to find vector $\mathbf{d}$. The frequency response can be found from (11.111) once $\mathbf{d}$ is known. An example is shown in Figure 11.7. The result is similar to the result in Figure 11.6, though no eigendecomposition is required.

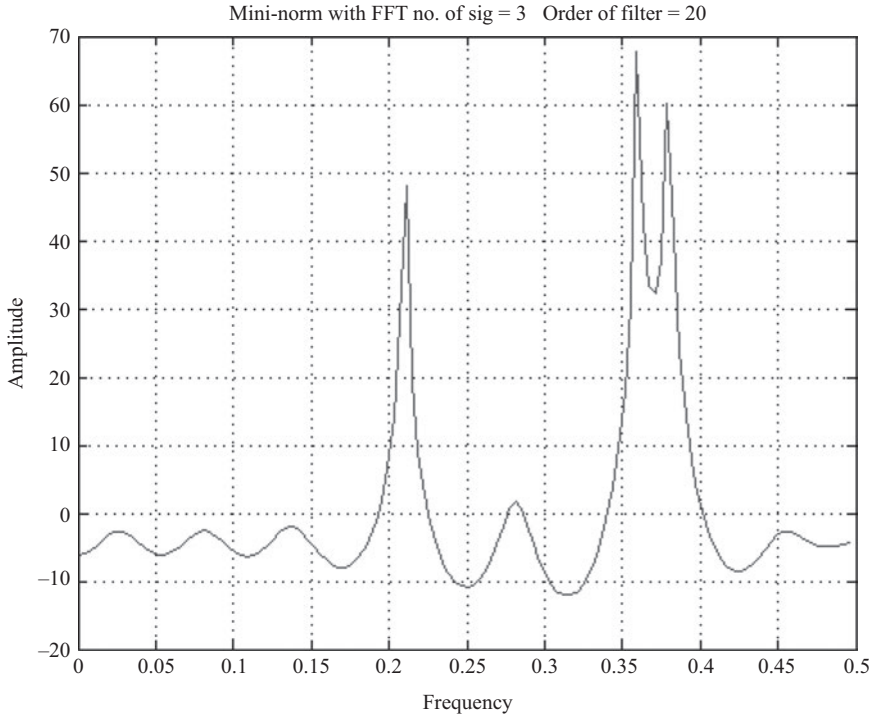


Figure 11.7 Frequency response of the minimum norm method with DFT.

11.16 Adaptive Spectrum Estimation [52–54]

The general idea of the adaptive method will not be discussed in this section, as there are many books on this subject [52, 53]. In this section, only sinusoidal inputs are considered since this input is suited for our application. This discussion is based on Cheung [54].

Let us assume that the input signal contains M sinusoidal waves. The digitized data can be written as

$$x(n) = \sum_{i=1}^M A_i e^{j2\pi f_i n} + u(n), \quad (11.118)$$

where A_i and f_i are the amplitude and frequency of the input sinusoidal wave and $u(n)$ is noise. The problem is, as usual, to find the amplitude and the frequency of the input signal given $x(n)$.

The basic relationship used in this approach is

$$e^{j2\pi f n} = 2 \cos(2\pi f) e^{j2\pi f(n-1)} - e^{j2\pi f(n-2)}. \quad (11.119)$$

This relation can be proved easily if one substitutes

$$\cos(2\pi f) = \frac{e^{j2\pi f} + e^{-j2\pi f}}{2} \quad (11.120)$$

into (11.119). A similar equation is used in Section 8.8 to find the zero crossing. Equation (11.119) indicates that the recent sample output $x(n)$ can be predicted by the two previous samples if the frequency f is known.

Separating the first term in (11.118), the following equations can be obtained:

$$\begin{aligned} x(n) &= A_1 e^{j2\pi f_1(n)} + \sum_{i=2}^M A_i e^{j2\pi f_i(n)} + u(n) \\ x(n-1) &= A_1 e^{j2\pi f_1(n-1)} + \sum_{i=2}^M A_i e^{j2\pi f_i(n-1)} + u(n-1) \\ x(n-2) &= A_1 e^{j2\pi f_1(n-2)} + \sum_{i=2}^M A_i e^{j2\pi f_i(n-2)} + u(n-2). \end{aligned} \quad (11.121)$$

Let us define a new quantity $x^{(1)}(n)$ as

$$\begin{aligned} x^{(1)}(n) &\equiv x(n) - 2 \cos(2\pi f_1) x(n-1) + x(n-2) \\ &\equiv x(n) - 2a_{1,n-1} x(n-1) + x(n-2), \end{aligned} \quad (11.122)$$

where

$$a_{1,n-1} = \cos(2\pi f_1). \quad (11.123)$$

The first subscript of a represents the numerical order of the signal and the second subscript represents the number of iterations. Substituting the results of (11.121) into (11.122), the result can be written as

$$x^{(1)}(n) = \sum_{i=2}^M \alpha_{i,1} A_i e^{j2\pi f_i n} + u^{(1)}(n), \quad (11.124)$$

where

$$\cos(2\pi f_1) \alpha_{k,m} = 2[\cos(2\pi f_k) - \cos(2\pi f_m)] \quad (11.125)$$

and $u^{(1)}(n)$ represents the residue noise. This operation removes the first sinusoidal signal from the $x(n)$ data and generates a second signal sequence $x^{(1)}(n)$.

This procedure can be applied to the new sequence $x^{(1)}(n)$ generated from (11.124). The result is

$$\begin{aligned} x^{(2)}(n) &\equiv x^{(1)}(n) - 2 \cos(2\pi f_2) x^{(1)}(n-1) + x^{(1)}(n-2) \\ &\equiv x^{(1)}(n) - 2a_{2,n-1} x^{(1)}(n-1) + x^{(1)}(n-2) \\ &= \sum_{i=3}^M \alpha_{i,1} \alpha_{i,2} A_i e^{j2\pi f_i n} + u^{(2)}(n), \end{aligned} \quad (11.126)$$

where

$$a_{2,n-1} = \cos(2\pi f_2). \quad (11.127)$$

This operation removes the second signal.

This process can be continued in a repetitive form until there are no further sinusoidal components, but only noise in the series. The general extraction process can be written as

$$\begin{aligned} x^{(k)}(n) &= x^{(k-1)}(n) - 2 \cos(2\pi f_k) x^{(k-1)}(n-1) + x^{(k-1)}(n-2) \\ &= x^{(k-1)}(n) - 2a_{k,n-1} x^{(k-1)}(n-1) + x^{(k-1)}(n-2) \\ &= \sum_{i=k+1}^M \left(\prod_{l=1}^k \alpha_{i,l} \right) A_i e^{j2\pi f_i n} + u^{(k)}(n) \quad \text{for } k = 1, 2, \dots, M \end{aligned} \quad (11.128)$$

$$x^{(0)}(n) = x(n)$$

$$x^{(k)}(0) = x(0),$$

where

$$a_{k,n-1} = \cos(2\pi f_k). \quad (11.129)$$

The final data $x^{(M)}(n)$ contain only noise. All the other terms $x^{(1)}(n), x^{(2)}(n), \dots, x^{(M-1)}(n)$ obtained through the intermediate steps contain signals.

It should be noted that $a_{k,n}$ is only related to $x^{(k)}$ and $x^{(k-1)}$. Once the final values of $a_{k,n}$ are obtained, the frequency can be obtained through (11.129) as

$$f_k = \frac{\cos^{-1}(a_{k,n})}{2\pi}. \quad (11.130)$$

The value of $a_{k,n}$ can be calculated from the gradient equation using the method of the steepest slope. Let us define the error as

$$\epsilon(n) \equiv x(n) - \hat{x}(n) \quad (11.131)$$

To find the gradient of the squared error is to take the derivative of $\epsilon^{(k)}(n)\epsilon^{(k)}(n)^*$ or

$$\begin{aligned} \nabla[\epsilon(n)\epsilon(n)^*] &= -2[\epsilon x^{(k-1)}(n-1)^* + \epsilon^* x^{(k-1)}(n-1)] \\ &= -2[x^M(n)x^{(k-1)}(n-1)^* + x^M(n)^* x^{(k-1)}(n-1)], \end{aligned} \quad (11.132)$$

where the superscript * represents the complex conjugate and $x^M(n)$ represents the noise, because there are only M signals.

To find $a_{k,n}$ from $a_{k,n-1}$ is done through the gradient of the squared error. The result is

$$\begin{aligned} a_{k,n} &= a_{k,n-1} + \mu \left(\frac{1}{-2} \right) \nabla[\epsilon^{(k)}(n)\epsilon^{(k)}(n)^*] \\ &= a_{k,n-1} + \mu [x^M(n)^* x^{(k)}(n-1) + x^M(n)x^{(k)}(n-1)^*]. \end{aligned} \quad (11.133)$$

This equation is obtained from the result of (11.132). The constant $-1/2$ is introduced to absorb the -2 into the constant μ , where μ is the step size. If μ is chosen too large, the result may not converge. If μ is chosen too small, the result will converge very slowly. The actual process is to calculate a_k values one at a time.

Let us put all the equations used in this calculation together and then use an example to demonstrate this operation. Two sets of equations are needed with initial conditions. If there are M signals, the initial conditions are

$$x^{(0)}(n) = x(n), \quad x^{(k)}(0) = x(0). \quad (11.134)$$

The first set of equations is

$$\begin{aligned}
 x^{(1)}(n) &= x^{(0)}(n) - 2a_{1,n-1}x^{(0)}(n-1) + x^{(0)}(n-2) \\
 x^{(2)}(n) &= x^{(1)}(n) - 2a_{2,n-1}x^{(1)}(n-1) + x^{(1)}(n-2) \\
 &\dots \\
 x^{(M)}(n) &= x^{(M-1)}(n) - 2a_{M,n-1}x^{(M-1)}(n-1) + x^{(M-1)}(n-2).
 \end{aligned} \tag{11.135}$$

The second set of equations is

$$\begin{aligned}
 a_{1,n} &= a_{1,n-1} + \mu [x^{(M)}(n)^* x^{(0)}(n-1) + x^{(M)}(n)x^{(0)}(n-1)^*] \\
 a_{2,n} &= a_{2,n-1} + \mu [x^{(M)}(n)^* x^{(1)}(n-1) + x^{(M)}(n)x^{(1)}(n-1)^*] \\
 &\dots \\
 a_{M,n} &= a_{M,n-1} + \mu [x^{(M)}(n)^* x^{(M-1)}(n-1) + x^{(M)}(n)x^{(M-1)}(n-1)^*].
 \end{aligned} \tag{11.136}$$

Now let us use an example to illustrate the applications of these equations. The input data are $x(0), x(1), \dots, x(N-1)$ and there are two signals ($M = 2$).

1. Arbitrarily choose $a_{1,0} = a_{2,0} = 0.25$ as the initial values and the initial conditions from (11.134) are

$$x^{(0)}(n) = x(n), \quad x^{(k)}(0) = x(0).$$

2. Use (11.135) to obtain $x^{(k)}(1)$ as

$$\begin{aligned}
 x^{(1)}(1) &= x^{(0)}(1) - 2a_{1,0}x^{(0)}(0) = x(1) - 2a_{1,0}x(0) \\
 x^{(2)}(1) &= x^{(1)}(1) - 2a_{2,0}x^{(0)}(0)
 \end{aligned}$$

In these equations, $x(-1)^*$ does not exist; thus only two terms are shown.

3. Use (11.136) to obtain $a_{k,1}$ as

$$\begin{aligned}
 a_{1,1} &= a_{1,0} + \mu [x^{(2)}(1)^* x^{(0)}(0) + x^{(2)}(1)x^{(0)}(0)^*] \\
 a_{2,1} &= a_{2,0} + \mu [x^{(2)}(1)^* x^{(1)}(0) + x^{(2)}(1)x^{(1)}(0)^*].
 \end{aligned}$$

4. Use (11.135) again to obtain $x^{(k)}(2)$ as

$$\begin{aligned}
 x^{(1)}(2) &= x^{(0)}(2) - 2a_{1,1}x^{(0)}(1) + x^{(0)}(0) \\
 x^{(2)}(2) &= x^{(1)}(2) - 2a_{2,1}x^{(1)}(1) + x^{(1)}(0).
 \end{aligned}$$

5. Use (11.136) again to obtain $a_{k,2}$ as

$$\begin{aligned}
 a_{1,2} &= a_{1,1} + \mu [x^{(2)}(2)^* x^{(0)}(1) + x^{(2)}(2)x^{(0)}(1)^*] \\
 a_{2,2} &= a_{2,1} + \mu [x^{(2)}(2)^* x^{(1)}(1) + x^{(2)}(2)x^{(1)}(1)^*].
 \end{aligned}$$

These two equations, (11.135) and (11.136), can be used alternatively to find the value of $a_{1,n}$ as a function of n . When $a_{1,n}$ converges, the value can be used to find the input frequency f_1 from (11.130).

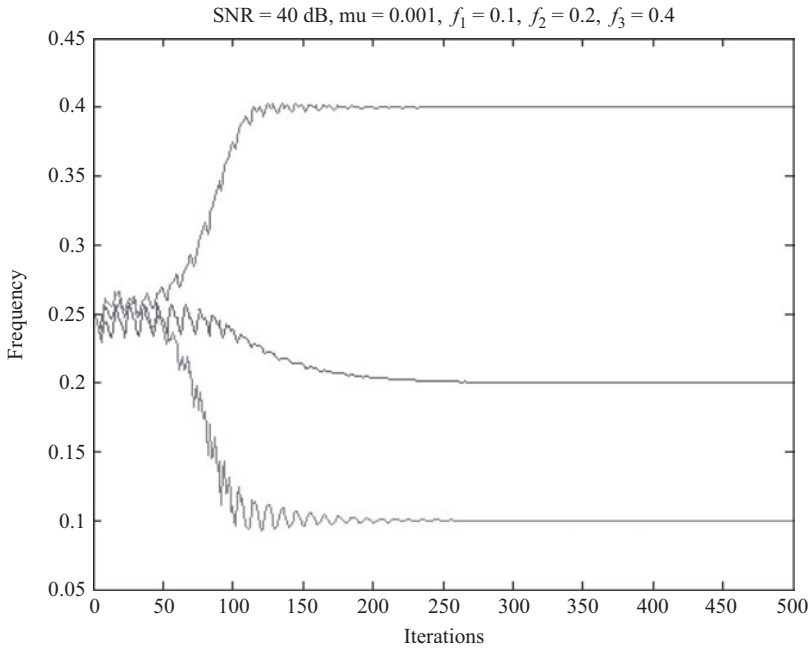


Figure 11.8 Adaptive frequency estimation.

If the k value in $a_{k,n}$ is smaller than the number of input signals M , the error $x^{(k)}(n)$ will oscillate, which means the remaining sinusoidal waves are still in the error signal. When the k value in $a_{k,n}$ is equal to the number of input signals M , the error $x^{(k)}(n)$ will approach a very small value, which is the noise level.

This method is sensitive to noise. If the input S/N level is low, the $a_{k,n}$ value will not converge fast or may not converge at all. It is also sensitive to step size and the number of iterations. A program to predict up to three signals is provided in Appendix 11.H. Figure 11.8 shows the result of three signals with frequencies at 0.1, 0.2, and 0.4 with the same amplitude and S/N ratio of 40 dB. Five hundred iterations are performed with $\mu = 0.001$. This specific case starts to converge to the desired frequencies at about 200 iterations.

References

- [1] Childers DG (ed.). *Modern Spectrum Analysis*. Piscataway, NJ: IEEE Press; 1978.
- [2] Kesler SB (ed.). *Modern Spectrum Analysis, II*. Piscataway, NJ: IEEE Press; 1986.
- [3] Marple SL Jr. *Digital Spectral Analysis with Applications*. Englewood Cliffs, NJ: Prentice Hall; 1987.
- [4] Kay SM. *Modern Spectral Estimation Theory and Application*. Englewood Cliffs, NJ: Prentice Hall; 1988.
- [5] Therrien CW. *Discrete Random Signals and Statistical Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1992.

- [6] Mitra SK, Kaiser JF. (eds.). *Handbook for Digital Signal Processing*. New York: John Wiley & Sons; 1993.
- [7] Parzen E. 'Some recent advances in time series modeling'. *IEEE Transactions on Automatic Control* 1974;**19**(6):723–730.
- [8] Makhoul J. 'Linear prediction: a tutorial review'. *Proceedings of the IEEE* 1975;**63**(4):561–580.
- [9] Jackson LB, Wood S. 'Linear prediction in cascade form'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1978;**26**(6):518–528.
- [10] Kay SM. 'The effects of noise on the autoregressive spectral estimator'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1979;**27**(5):478–485.
- [11] Kay SM. 'Noise compensation for autoregressive spectral estimates'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1980;**28**(3):292–303.
- [12] Toomey JP. 'High-resolution frequency measurement by linear prediction'. *IEEE Transactions on Aerospace and Electronic Systems* 1980;**AES-16**(4):517–525.
- [13] Kay SM, Marple SL Jr. 'Spectrum analysis—a modern perspective'. *Proceedings of the IEEE* 1981;**69**(11):1380–1419.
- [14] Kay SM. 'Robust detection by autoregressive spectrum analysis'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1982;**30**(2):256–269.
- [15] Kumaresan R. 'Accurate frequency estimation using an all-pole filter with mostly zero coefficients'. *Proceedings of the IEEE* 1982;**70**(8):873–875.
- [16] Cadzow JA. 'Spectral estimation: an overdetermined rational model equation approach'. *Proceedings of the IEEE* 1982;**70**(9):907–939.
- [17] Kay S, Makhoul J. 'On the statistics of the estimated reflection coefficients of an autoregressive process'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1983;**31**(6):1447–1455.
- [18] Tretter S. 'Estimating the frequency of a noisy sinusoid by linear regression'. *IEEE Transactions on Information Theory* 1985;**31**(6):832–835.
- [19] Tsui JBY. *Digital Microwave Receivers: Theory and Concepts*. Norwood, MA: Artech House; 1989.
- [20] Levinson N. 'The Wiener rms (root mean square) error criterion in filter design and prediction'. *Journal of Mathematical Physics* 1947;**25**:261–278.
- [21] Durbin J. 'The fitting of time series models'. *Review of the International Statistical Institute* 1960;**28**(3):233–244.
- [22] Haykin SH. *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice Hall; 1986.
- [23] Burg JP. 'Maximum entropy spectral analysis'. Paper presented at the Thirty-Seventh Society of Exploration Geophysicists Meeting, Oklahoma City, OK, October 1967.
- [24] Burg JP. 'A new analysis technique for time series data'. Paper presented at the NATO Advanced Study Institute on Signal Processing with Emphasis on Underwater Acoustics, Enschede, The Netherlands, August 1968:12–23.
- [25] Chen WY, Stegen GR. 'Experiments with maximum entropy power spectra of sinusoids'. *Journal of Geophysical Research* 1974;**79**(20):3019–3022.

- [26] Burg JP. 'Maximum entropy spectral analysis'. Ph.D. dissertation, Stanford University, 1975.
- [27] Ulrych TJ, Clayton RW. 'Time series modeling and maximum entropy'. *Physics of the Earth and Planetary Interiors* 1976;**12**(2–3):188–200.
- [28] Fougere PF, Zawalick EJ, Radoski HR. 'Spontaneous line splitting in maximum entropy power spectrum analysis'. *Physics of the Earth and Planetary Interiors* 1976;**12**(2–3):201–207.
- [29] Swingler DN. 'A comparison between Burg's maximum entropy method and a nonrecursive technique for the spectral analysis of deterministic signals'. *Journal of Geophysical Research: Solid Earth* 1979;**84**(B2):679–685.
- [30] Swingler DN. 'A modified Burg algorithm for maximum entropy spectral analysis'. *Proceedings of the IEEE* 1979;**67**(9):1368–1369.
- [31] Kaveh M, Lippert G. 'An optimum tapered Burg algorithm for linear prediction and spectral analysis'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1983;**31**(2):438–444.
- [32] Helme B, Nikias CL. 'A high-resolution modified Burg algorithm for spectral estimation'. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 9. New York: IEEE; 1984:531–534.
- [33] Akaike H. 'Statistical predictor identification'. *Annals of the Institute of Statistical Mathematics* 1970;**22**(2):203–217.
- [34] Akaike H. 'A new look at the statistical model identification'. *IEEE Transactions on Automatic Control* 1974;**19**(6):716–723.
- [35] Hildebrand FB. *Introduction to Numerical Analysis*. New York: McGraw-Hill; 1956.
- [36] Chuang CW, Moffatt DL. 'Natural resonances of radar targets via Prony's method and target discrimination'. *IEEE Transactions on Aerospace and Electronic Systems* 1976;**AES-12**(5):583–589.
- [37] VanBlaricum M, Mittra R. 'Problems and solutions associated with Prony's method for processing transient data'. *IEEE Transactions on Antennas and Propagation* 1978;**26**(1):174–182.
- [38] Kumaresan R, Tufts DW. 'Improved spectral resolution III: efficient realization'. *Proceedings of the IEEE* 1980;**68**(10):1354–1355.
- [39] Kumaresan R, Tufts DW, Scharf LL. 'A Prony method for noisy data: choosing the signal components and selecting the order in exponential signal models'. *Proceedings of the IEEE* 1984;**72**(2):230–233.
- [40] Kumaresan R. 'Spectral analysis', in Mitra SK, Kaiser JF (eds.), *Handbook for Digital Signal Processing*. New York: John Wiley & Sons; 1993.
- [41] Tufts DW, Kumaresan R. 'Singular value decomposition and improved frequency estimation using linear prediction'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1982;**30**(4):671–675.
- [42] Schmidt RO. 'Multiple emitter location and signal parameter estimation'. *IEEE Transactions on Antennas and Propagation* 1986;**34**(3):276–280.
- [43] Schmidt RO. 'A signal subspace approach to multiple emitter location and spectral estimation'. Ph.D. dissertation, Stanford University, 1981.

- [44] Paulraj A, Roy R, Kailath T. 'Estimation of signal parameters via rotational invariance techniques—Esprit'. *Proceedings of the Nineteenth Asilomar Conference on Circuits, Systems and Computers*. New York: IEEE; 1985:83–89.
- [45] Roy R, Paulraj A, Kailath T. 'ESPRIT—a subspace rotation approach to estimation of parameters of cisoids in noise'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1986;**34**(5): 1340–1342.
- [46] Roy RH. 'ESPRIT: estimation of signal parameters via rotational invariance techniques'. Ph.D. dissertation, Stanford University, 1987.
- [47] Zoltowski MD, Stavrinos D. 'Sensor array signal processing via a procrustes rotations based eigenanalysis of the ESPRIT data pencil'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1989;**37**(6):832–861.
- [48] Roy R, Kailath T. 'ESPRIT—estimation of signal parameters via rotational invariance techniques'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1989;**37**(7): 984–995.
- [49] Tufts DW, Kumaresan R. 'Estimation of frequencies of multiple sinusoids: making linear prediction perform like maximum likelihood'. *Proceedings of the IEEE* 1982;**70**(9):975–989.
- [50] Kumaresan R, Tufts DW. 'Estimating the angles of arrival of multiple plane waves'. *IEEE Transactions on Aerospace and Electronic Systems* 1983;**AES-19**(1):134–139.
- [51] Shaw AK, Xia W. 'Minimum-norm method without eigendecomposition'. *IEEE Signal Processing Letters* 1994;**1**(1):12–14.
- [52] Cowan CFN, Grant PM (eds.). *Adaptive Filters*. Englewood Cliffs, NJ: Prentice Hall; 1985.
- [53] Widrow B, Stearns SD. *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1985.
- [54] Cheung JY. 'A direct adaptive frequency estimation technique'. *Proceedings of the 30th Midwest Symposium on Circuits and Systems*. New York: IEEE; 1987:804–807.

Appendix 11.A

```
% df12_2.m generate input data
clear
N = 4096;
t = [0:1:31];
x = cos(2*pi*.21*t+.1) + 2*cos(2*pi.36*t) + 1.9*cos(2*pi*.38*t);
xp = [x zeros(1,N-32)];
xpf = fft(xp);
xpa = abs(xpf);
xaxis = linspace(0,.5,N/2);
plot(xaxis, xpa(1:N/2))
xlabel('Frequency')
ylabel('Amplitude')
grid
```

Appendix 11.B

```
% ***** df12_3 Autocorrelation method *****
% ***** Autocorrelation method (or the Yule-Walker approach)
clear
load xx.mat
N = length(x);
ip=input(' The order of the model ip = ?');
% ***** Generate the autocorrelation coefficient vector r_xx from x
r_xx=zeros(1,ip+1);
for k=1:ip+1,
    for n=1:N-k+1,
        r_xx(k)=r_xx(k)+conj(x(n))*x(n+k-1);
    end
end
r_xx=r_xx./N;
% ***** Generate the autocorrelation coefficient matrix R from r_xx
for ii=1:ip,
    for jj=ii:ip,
        R(jj,ii)=r_xx(jj-ii+1);
    end
end
for ii=1:ip-1,
    for jj=ii+1:ip,
        R(ii,jj)=R(jj,ii)';
    end
end

% ***** Find the autoregressive model coefficient vector a
a=inv(R)*(-r_xx(2:ip+1)).';
% ***** Find the power spectral density S_xx of the autoregressive model
f = linspace(0,.5,100);
for ii=1:100,
    s=0;
    for jj=1:ip,
        s=s+a(jj).'*exp(-i*2*pi*f(ii)*jj);
    end
    S_xx(ii)=1/(abs(1+s)^2);
end

% ***** plot *****
S_xxlog = 10*log10(S_xx);
xaxpt = length(S_xx);
xax = linspace(0,.5,xaxpt);
plot(xax, S_xxlog)
ips = conv_vs(ip);
title(['AR Process      order =', ips])
xlabel('Frequency')
ylabel('Amplitude')
grid
```

Appendix 11.C

```
% df12_4.m Burg method
clear
load xx.mat
p = input('enter # of poles = ');
a = Ipc(x,p);
H = freqz(1, a, 1000);
xaxis = linspace(0, .5, 1000);
plot(xaxis, abs(H))
xlabel('Frequency')
ylabel('Amplitude')
plab = conv_vs(p);
title(['Number of poles = ', plab])
grid
```

Appendix 11.D

```
% df12_5 MUSIC algorithm
% Updated: July 23 2015
% *****
clear
close all
load xx.mat
% *****
numpts = 512; % input('Enter Number of Output Data Points: ');
Ns = input('Enter Number of Sources: ');
mf = input('Enter Desired Filter Order (min = 2*source#+1) : ');
K = length(x); % Number of snapshots
% *****
% Compute the R matrix using the covariance method
% *****
C2 = [];
for k = 1:mf
    C1 = x(k:K-mf+k);
    C2 = [C2; C1];
end
Cb=C2';
rmat0 = C2*Cb/(K-mf);
```

```

% *****
[v0, d0] = eig(rmat0); % find eigenvalues
[lambda0,k10]=sort(diag(d0));
E0=v0(:,k10);
nspace0 = E0(:,1:mf-2*Ns);
% nspace = E(:,1:mf-Ns); % for complex data
for k = 1:numpts
    w0(k) = (k-1)*2*pi/(numpts);
    l = 1:(mf);
    s0 = exp(-1j*w0(k)*(l-1));
    pmu0(k) = inv(s0*(nspace0*nspace0')*s0');
end
pmu0 = abs(pmu0);
pmu0 = 10.0*log10(pmu0/max(pmu0));
% *****
xaxpt0 = length(pmu0)/2;
xax0 = linspace(0,.5,xaxpt0);
figure(1)
plot(xax0, pmu0(1:xaxpt0)-max(pmu0(1:xaxpt0)))
grid;

Nss = conv_vs(Ns);
mfs = conv_vs(mf);
title(['Number of signals = ', Nss, ' Order of filter = ', mfs]);
xlabel('Frequency');
ylabel('Amplitude');_

```

Appendix 11.E

```

% ESPRIT method
clear
load xx.mat
N=length(x);
x=x.';
mm=input('enter the number of signals = ');
l=N/2;
l=diag(diag(ones(l,l)));
% ***** Generate D1 matrix *****
% D1=tril(ones(l,l));
d2 = diag(diag(ones(l-1,l-1)));
d3 = zeros(l-1,1);
d4 = zeros(1,1);
d5 = [d2 d3];
d6 = [d4 d5'];
D1 = d6';

```

```

% ***** Generate Ryy Ryz matrices *****
for k=1:l,
    x_k(:,k)=x(k:k-1+1,1);
end
Ryy_est=x_k*x_k'/l;
y=x(2:N);
for k=1:l,
    y_k(:,k)=y(k:k-1+1,1);
end
Ryz_est=x_k*y_k'/l;
% ***** Generate Rs Rss matrices *****
[u_est,q]=eig(Ryy_est);
for ii=1:l,
    m(ii)=q(ii,ii);
end
[y1,g1]=min(abs(m));
var0=q(g1,g1);
Rs=Ryy_est-l.*var0;
Rss=Ryz_est-D1.*var0;
% ***** Find eigenvalues of Rs Rss *****
[u_est,q]=eig(Rs, Rss);
for ii=1:l,
    m(ii)=q(ii,ii);
end
% ***** Find Frequency *****
for ii=1:mm,
    [y1 ,g(ii)]=min(abs(1-abs(m)));
    m(g(ii))=0;
    z(ii) = q(g(ii),g(ii));
end
r=angle(z);
f=r/pi/2;
a = find(f>0);
f(a)

```

Appendix 11.F

```

% df12_6.m Minimum_Norm
% Updated: July 23, 2015
% ***** Input the input vector size and load input data *****
clear
load xx.mat;
X = x;

```

```

points = 512; % input('Enter Number of Output Data Points: ');
Ns = input('Enter the Number of Sources: ');
Fs = input('Enter Desired Filter Size: ');
K=length(X); % Number of snapshots

% ***** Compute the R matrix using covariance method *****

C2 = [];
for k = 1:Fs
    C1 = x(k:K-Fs+k);
    C2 = [C2; C1];
end
Cb=C2';
Rx = (C2*Cb)/(K-Fs);

% ***** Compute the eigenvalues and eigenvectors of Rx *****
[Ea,L] = eig(Rx);
EL = length(Ea);
[lambda,k1]=sort(diag(L));
E=Ea(:,k1);
En = E(:,1:Fs-2*Ns);
% En = E(:,1:Fs-Ns); % for complex data
Enp = En(2:EL,:);
c = Enp(1,:);
% ***** Compute the vector d *****
d = 1/(c*c')*(En*c');
% ***** Compute Pseudospectrum *****
for k=1:points
    w(k) = 2*pi*(k-1)/(points); % 2pi coverage
    [v,t]=size(d);
    l=1:v;
    B1 = exp(1j*w(k)*(l-1));
    B = B1(:);
    Pmn(k) = 1.0/(B'*(d*d')*B);
end
Pmn = Pmn(:);
Pmn = 10*log10(abs(Pmn));
% ***** Plot *****
xaxpt = length(Pmn)/2;
xax = linspace(0,.5,xaxpt);
plot(xax, Pmn(1:xaxpt))
xlabel('Frequency')
ylabel('Amplitude')
nss = conv_vs(Ns);
fss = conv_vs(Fs);
title(['Mini-norm no. of sources = ', nss, ' Order of filter = ', fss])
grid

```

Appendix 11.G

% df12_7.m DFT - min norm method (three real signals)

```

clear
load xx.mat
l= input('enter order of filter = ');
ns = input('enter number of signals = ');
N=length(x);
p=256; % (number of output data points)
f=0:1/p:1-1/p;
for ii=1:p/2,
    for jj=1:l,
        e_f(jj,ii)=exp(j*2*pi*(jj-1)*f(ii));
    end
end

for ii=1:l,
    for jj=1:l,
        D(ii,jj)=exp(i*(2*pi/l)*(ii-1)*(jj-1));
    end
end

for k=1:N-l+1,
    x_k(:,k)=x(k:k-1+1).';
end

for k=1:l,
    x_kk(k,:)=x_k(l-(k-1),:);
end
x_kk=conj(x_kk);

Rxx_est=zeros(1,1);
for k=1:N-l+1,
    Rxx_est=x_k(:,k)*x_k(:,k)'+x_kk(:,k)*x_kk(:,k)'+Rxx_est;
end
Rxx_est=Rxx_est./2./l;
v=Rxx_est*D;
for ii=1:l,
    m(ii)=norm(v(:,ii));
end

for ii=1:ns*2,
    [y,g(ii)]=max(abs(m));
    m(g(ii))=0;
    Es(:,ii)=v(:,g(ii));
end

```

```

g=Es(1,:)' ;
Es_=Es(2:1,:);
d_s(1,1)=1;
d_s(2:1,1)=-Es_*inv(Es_'*Es_)*g;

for ii=1:p/2,
    P_est(ii)=1/(abs(e_f(:,ii)'+d_s)^2);
end
plot([0:1/p:1/2-1/p],20*log10(P_est))
ylabel('Amplitude')
xlabel('Frequency')
nss = conv_vs(ns);
Is = conv_vs(1);
title(['Mini Norm with FFT # of sig = ', nss, 'filter order = ', Is])
grid

```

Appendix 11.H

```

% df12_8.m
% This program uses the direct adaptive frequency estimation
% technique (DAFE) by John Y. Cheung, using the least-mean-square
% algorithm (LMS).
% Use u = .001 and iter = 500.

%Steve Nunes
%5/29/92

clear

% Input Parameters.    'order' signifies the number of frequencies detected
% above the number present. 'avnum' is the number of iterations, over
% which, a given estimate is to be averaged. 'char' and 'blank'
% are used for printing input frequencies on plots, 'u' is a constant
% used in the adaptive process for convergence.

M=input('Enter the number of frequencies present (max 3): ');
order=M; avnum=49; char=4; blank=' ';
u=input('Enter value of u (max .01): ');
SNR=input('Enter the SNR: ');
iter=input('Enter # of iterations(min of 100): ');

% Set input signal with noise. Set random numbers to normal distribution.
nampl=1/(sqrt(2)*10^(SNR/20));% amplitude of the noise
noise=nampl*(randn(1,iter) + j*randn(1,iter));% and produce sequence.

```



```

f=[.1 .2 .4] ;      % Vectors containing input frequencies and
A=[2 2 2];          % amplitudes. M are used.
w=2*pi*f;           % Convert Hz to radians.
x3=zeros(1,iter);    % Initialize temp variable x3 to zeros.
k=linspace(1,iter,iter); % Initialize time vector.
x0=zeros(1 ,M);      % Initialize phase vector x0 to zeros.
x0(1)=j*2*pi*0.125;  % Initialize first phase to its value.

for i=1:M, % Loop through the number of
    x1=A(i)*exp(j*w(i)*k + x0(i)); % frequencies present adding
    x3=x3 + x1; % the next signal to the
end % previous.
x=x3 + noise; % Add the noise to signal.

%***** Iterate through the LMS algorithm *****
M1=order; % Set M1 equal to the order of the LMS algorithm
f_est=zeros(M1,iter); % used. Initialize the frequency estimate, f_est,
e=zeros(M1+1,iter+1); % the partial signal error, e, and the adaptation
a=zeros(M1,iter); % variable, a, to zero. Set the first partial
e(1,1: iter)=x; % signal, e(1,:), to the sampled sequence, x.

for l=1 :M1, % Initialize the first values of
    e(l+1,1)=x(1); % e and f_est, needed in
    f_est(l,1)=acos(a(l,1))/2/pi; % calculations below.
    e(l+1,2)=e(l,2) -2*cos(2*pi*f_est(l,1))*e(l,1);
end

for i=2:iter, % For each sample and for each signal, find next
    for l=1:M1, % values of a and e through iteration.
        a(l,i)=a(l,i-1) + u*(e(M1+1,i)'*e(l,i-1) + e(M1+1,i)*e(l,i-1)');
        f_est(l,i)=acos(a(l,i))/2/pi; % Calculate the estimates.
        if i==2, % The e(l,i-1) term is zero.
            e(l+1,i+1)=e(l,i+1) - 2*cos(2*pi*f_est(l,i))*e(l,i);
        else
            e(l+1,i+1)=e(l,i+1) - 2*cos(2*pi*f_est(l,i))*e(l,i) + e(l,i-1);
        end
    end
end

% ***** Convert SNR to string for plot printing *****
plot(f_est') % Plot the frequency estimates.
xlabel('Iterations')
ylabel('Frequency')
title(['SNR = ' num2str(SNR)' dB, mu = ' num2str(u) ', f1 = '
num2str(f(1)) ', f2 = '
num2str(f(2)) ', f3 = ' num2str(f(3)) '])

```

Appendix 11.I

```
% CONV_VS converts from a vector to string
% JT April 29 1992

function str = convs(r)
str=[];
for i = 1:length(r)
    eval(['str=[str, ',num2str(r(i)), ' '];']);
end
```


Detection of Biphase Shift Keying (BPSK) Signals

12.1 Introduction

In this chapter the detection of biphase shift keying (BPSK) is discussed. In this type of signal, the phase shift is π . There are different phase shift keying (PSK) signals, such as $\pi/2$ and $\pi/4$, especially in communication signals [1]. However, this study focuses on BPSK since it is more popular in radar applications [2–4]. Many different codes can be used to generate a BPSK signal. For example, the Global Position System (GPS) signal is a BPSK signal modulated by a Gold code [5]. The popular code used in radars is the Barker code, which has different lengths [6]. Thirteen bits is the longest Barker code available [7], and this code will be used here.

It should be pointed out that the main subject of this book is the design of a wideband receiver capable of detecting uncooperative signals over a broad bandwidth and the existing receiver is mainly designed to intercept pulsed continuous wave (CW) signals. Such receiver examples can be found in Tsui [8] and Chapter 17. The main purpose of this chapter is not to design a special receiver for detecting a BPSK signal, but to extend the capability of the existing receiver. Therefore this chapter is dedicated to the detection of the existence of a BPSK signal and locating the phase transition, thus determining the chip time. To extract coding of the transmitted signals is not considered here, although it can be a desirable function. The receiver specifications used in the simulation are a 2.56-GHz sampling rate, 128-point fast Fourier transform (FFT) frame size (equivalent to a 50 ns duration), and the Blackman window applied to reduce energy spreading. The input signal frequency is from 141 to 1140 MHz and the signal is processed through consecutive FFT frames windowed by a Blackman window. Such a setup is also suitable for pulsed CW signal detection [8].

Since this book emphasizes the introduction of general wideband receiver technologies rather than the detection of specific signals, to simplify the discussion only a perfect pulsed BPSK signal that exhibits perfect $\pm\pi$ phase transitions and instantaneous leading and falling edges is considered. The signal pulse width is assumed to be 1 μ s. Although it is possible to have a BPSK signal with a much longer pulse width, this signal length should be adequate to illustrate the detection concepts presented in this chapter.

The discussion is divided into three steps. The first step is to detect the existence of a BPSK signal. The sensitivity on detection is determined first. The detected signal can be

either a CW signal or a BPSK signal. Thus the second step is to differentiate the detected signal and determine whether it is CW or BPSK. There are two approaches to achieve this goal. The first one is from the measured pulse width (PW) to perform a relatively long data operation. This operation might complicate the actual receiver design since long data operations such as the FFT or eigenvalue method must be performed. The other approach is use a relatively short signal such as one or two frames of data, which is more suitable for receiver designs. The third step is to determine the location of the phase transition. From the location of the phase transition, the chip time, that is, the shortest time between two consecutive transitions, can be estimated. The inverse of the chip time is often referred to as the chip rate.

12.2 Basic Barker Code Properties [6, 7]

The Barker code is a short binary sequence that is popular for radar applications. The Barker codes have different sequence lengths, such as 2, 3, 4, 5, 7, 11, and 13. The sidelobes generated from the autocorrelation of a Barker code have constant amplitude. The maximum length of a Barker code is limited to 13 bits. It appears that longer code does not have the property of a constant sidelobe. The 11- and 13-bit Barker codes are shown in (12.1) [7]:

$$\begin{aligned} B_{11} &= [1, -1, 1, 1, -1, 1, 1, 1, -1, -1, -1] \\ B_{13} &= [1, 1, 1, 1, 1, 1, -1, -1, 1, 1, -1, 1, 1] \end{aligned} \quad (12.1)$$

The 1 and -1 represent the 0 and π phases in a sinusoidal wave. A 13-bit Barker code is shown in Figure 12.1. Figure 12.1(a) shows the code itself in the time domain and Figure 12.1(b) shows the autocorrelation of the code. In this figure all the sidelobes are of the

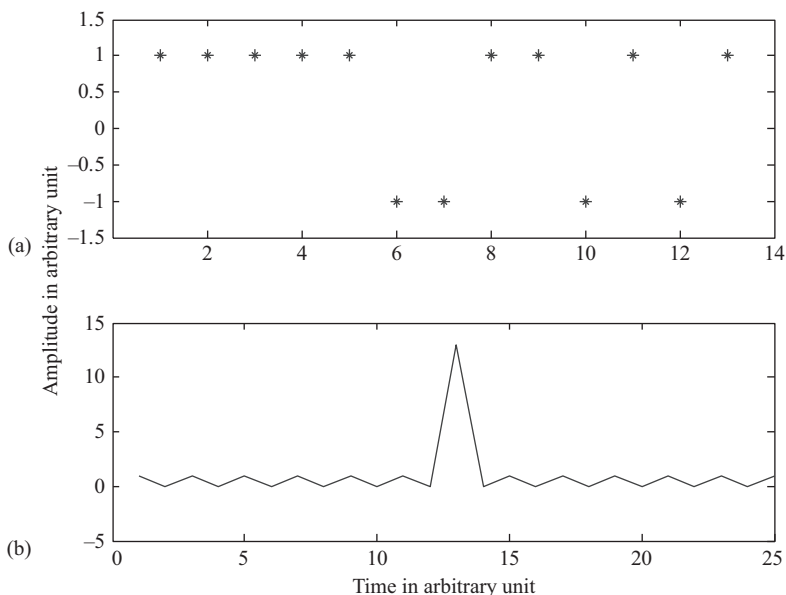


Figure 12.1 A 13-bit Barker code: (a) time domain and (b) autocorrelation.

same amplitude; this is a unique property of the Barker codes. In this chapter only the BPSK signal of 13-bit Barker code that contains more phase transitions is considered.

12.3 Generation of BPSK Signals with Synchronized Radio Frequency (RF) and Chip Time

It is anticipated that the RF carrier and the code in a BPSK radar are generated from a single crystal. In other words, the RF carrier and code are phase locked. The simulation will take this property into consideration. The 13-bit Barker code is used in this example and the chip time is fixed at approximately 77 ns ($\approx 1000/13$). The chip time varies with the input frequency as shown in the following procedures. The steps that generate a 1- μ s BPSK signal for (or from) 2560 points of digitized data are listed below.

1. Generate an RF signal with random initial phase θ_i and input frequency f_i . The corresponding time t_i for one RF cycle and the initial phase at time t_0 are

$$\begin{aligned} t_i &= \frac{1}{f_i} \\ t_0 &= \frac{\theta_i}{2\pi f_i}. \end{aligned} \quad (12.2)$$

In this specific study, the phase transition is chosen at zero phase of the RF waveform. Figure 12.2 shows the initial phase, the RF carrier cycle, the digitized point, and the first phase transition. This is just one special case and will be used to generate the desired signal for study. The phase transition can occur at any point of an RF cycle as long as they are synchronized, which means the phase transition crosses at the same point on all RF cycles.

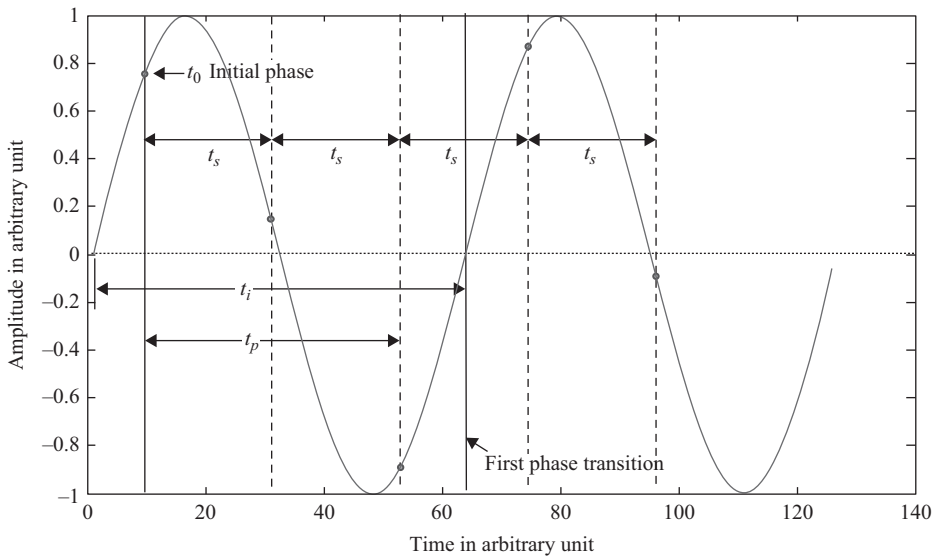


Figure 12.2 Input signal with initial digitizing point and first phase transition.

2. Calculate the total number of RF cycles N_r in a 1- μ s time interval:

$$N_r = f_i \times 1 \mu\text{s} = f_i \times 10^{-6}. \quad (12.3)$$

3. The number N_r is divided by 13, the total length of the Barker code, and takes only the integer part of the result. This approach synchronizes the RF and chip rate. The total number of RF cycles per chip N_c and the chip time t_c are

$$N_c = \text{floor}\left(\frac{N_r}{13}\right) \quad (12.4)$$

$$t_c = N_c \times t_i,$$

where floor means to take the lower integer value of the division.

4. The first phase transition time t_p can be found as

$$t_p = t_0 + \text{round}\left(\frac{t_i - t_0}{t_s}\right)t_s \quad (12.5)$$

$$t_s = \frac{1}{f_s},$$

where t_s is the sampling time, f_s is the sampling frequency, and round is choosing the nearest integer. The t_p obtained is a point close to the zero phase of the RF cycle but not at it because of the digitization effect shown in Figure 12.2. The rest of the phase transitions can be found by adding t_c to t_p and rounding off to an integer. Once all the phase transitions are found, the Barker coded signal can be generated.

Figure 12.3(a) shows the phase transition location of a 13-bit Barker code. Figures 12.3(b) to 12.3(g) show the individual phase change in the RF cycle in a Barker code, where the time matches the location in the code shown in Figure 12.3(a). The results agree with the designed signal goals, that is, the phase transition occurs near 0 phase of the RF cycle. In this figure, the input frequency is at 141 MHz, the lowest value of the receiver input band, because at a higher frequency it is difficult to visualize the phase change in the signal at the correct phase. The initial phase is arbitrarily chosen and there is no noise. The program used to generate a BPSK signal can be found in Appendix 12.A.

12.4 Comparison of 11- and 13-Bit Barker Codes in the Frequency Domain

In this section, the Barker codes with 11 and 13 bits are simulated to show their difference in the frequency domain. In these examples the input frequency is at 141 MHz and the initial phase is randomly chosen. The low input frequency is selected to be consistent with the previous example.

Figures 12.4(a) and 12.4(b) show the time and frequency domains of the 11-bit Barker code, respectively. The actual Barker code contains 2396 points from point 21 to 2416, which is 164 points less than the total digitized points of 2560, as shown in Figure 12.4(a). One code length is about 218 points and there are six +1s and five -1s. The chip time is close to 85 ns ($218/(2.56 \times 10^9)$). The FFT is only performed on the actual Barker code of 2396 points and the result is shown in Figure 12.4(b). In this figure, only the neighboring

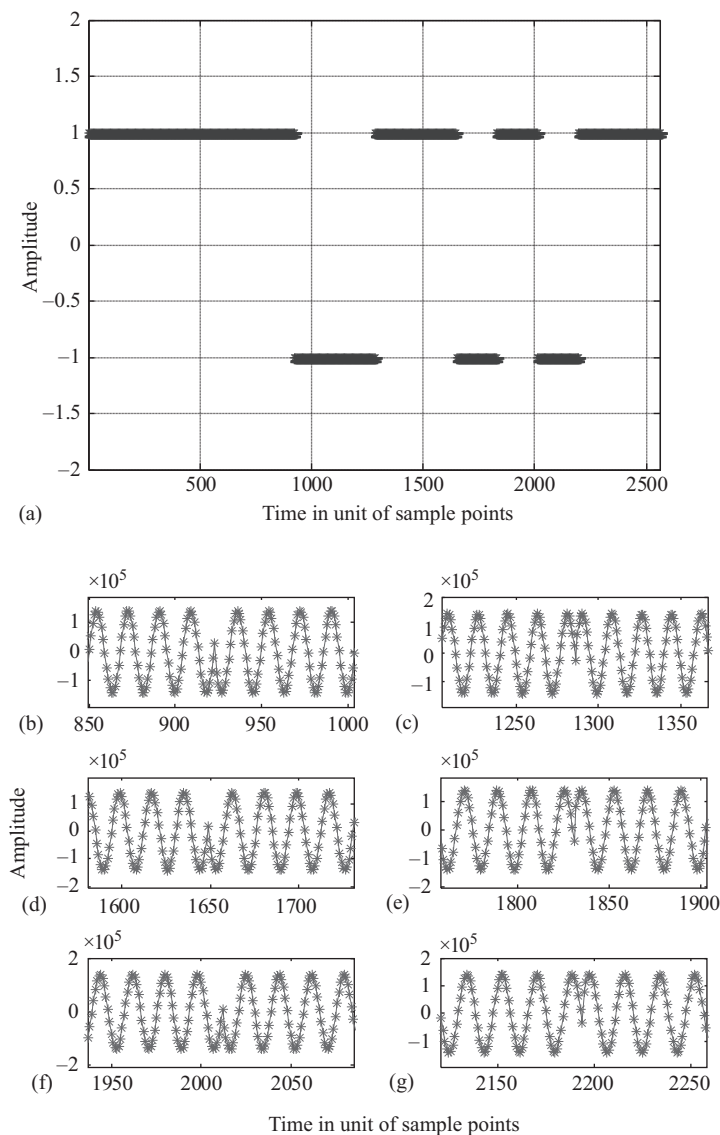


Figure 12.3 Phase transitions of a 13-bit Barker code: (a) phase code locations and (b–g) sequential phase shift along the code (amplitude unit: arbitrary).

points close to the maximum frequency bin are shown and the carrier frequency is suppressed. Since there are six +1s and five –1s, they are comparable in number (5 and 6) and the carrier is canceled; this is the commonly expected result.

Figures 12.4(c) and 12.4(d) show the time and frequency domains of the 13-bit Barker code, respectively. The actual Barker code contains 2360 points from point 21 to 2380, which is 200 points less than the total digitized points of 2560, as shown in Figure 12.4(c). One code length is 181 or 182 points and there are nine +1s and four –1s. The chip time is

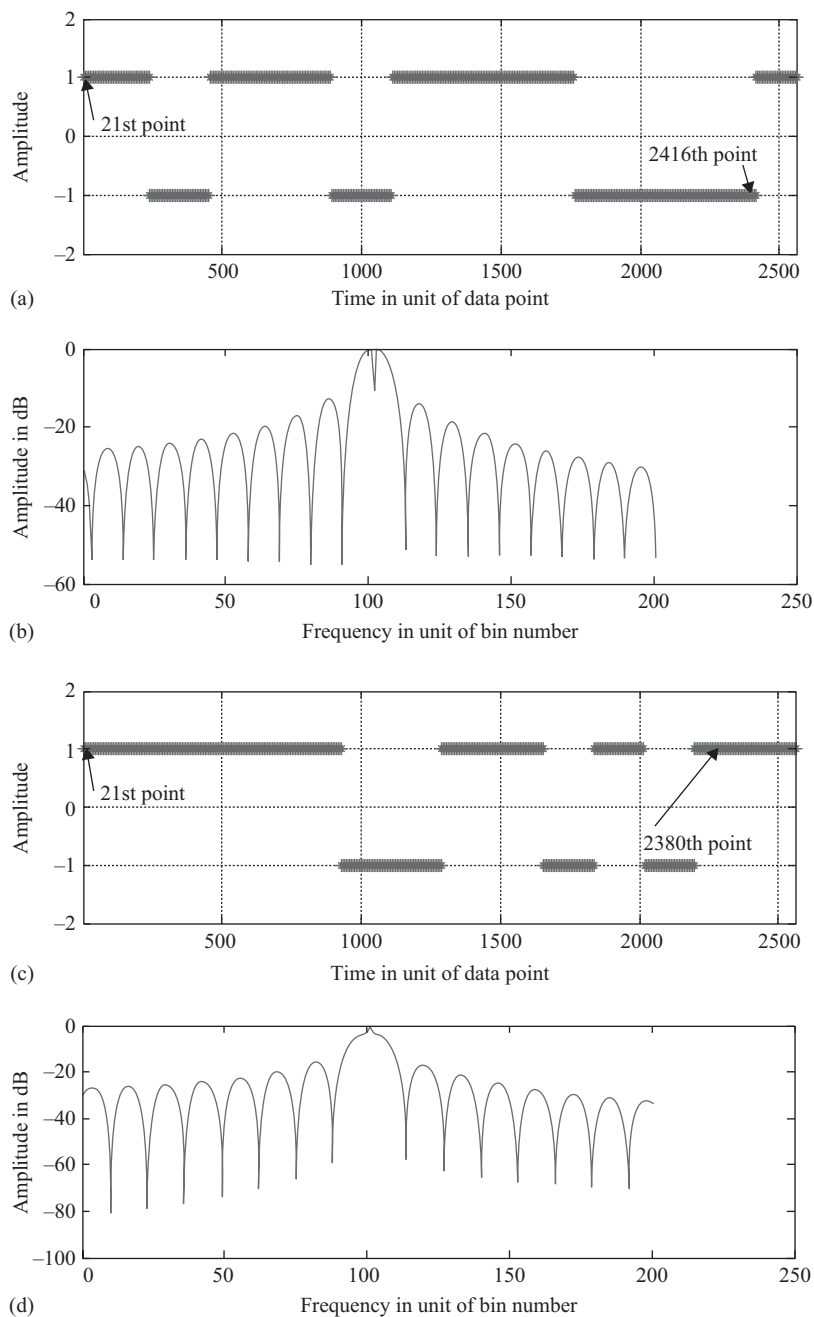


Figure 12.4 Plots of 11- and 13-bit Barker codes in the time and frequency domains with 2560 data points: (a) 11-bit Barker code in the time domain, (b) 11-bit Barker code in the frequency domain, (c) 13-bit Barker code in the time domain, and (d) 13-bit Barker code in the frequency domain.

close to 70 ns ($181/(2.56 \times 10^9)$). Since the frame time is 50 ns, one frame contains only one phase transition, but two phase transitions can be in two consecutive frames. The FFT operation is only performed on the actual Barker code of 2360 points and the result is shown in Figure 12.4(d). In this figure only the neighboring points close to the maximum frequency bin are shown and the carrier frequency is slightly higher rather than suppressed. The reason is that there are nine +1s and four -1s, and the positive ones are more than double the negative ones and the carrier frequency is slightly suppressed. Thus whether the carrier frequency of a BPSK signal is partially or totally suppressed depends on the number of positive and negative cycles in the signal. If slightly suppressed the carrier frequency may show a peak instead of a valley, as shown in Figure 12.4(d). The amplitude of the power spectrum is represented in decibels (relative to the peak value).

12.5 Threshold and Probability of Detection

In the following discussions, only the 13-bit Barker code is considered. If the BPSK radar radiates only one long Barker code, the signal should contain only 2360 points. At other carrier frequencies, the Barker code generates a different number of digitized points because of the signal generation procedure described in Section 12.3. If only the data with signal is used to study sensitivity, the total signal length will have an impact on the FFT length and sensitivity since the sensitivity is determined based on the total signal energy received, and 2360 points means an 8% energy loss compared with 2560 points. In order to eliminate this variation, the signal is extended to 2560 points for all input frequencies. The extra data points appear at the beginning and the ending of the digitized signal. The extra points at the beginning are less than the data points in one RF cycle and the extra ending points are determined by the extra RF cycles. This approach may not match the real-world radar signal, but it should produce a better accuracy on the sensitivity test.

Since the carrier frequency of the BPSK signal is a constant, detection of a CW signal through summation of amplitudes can be used. There are 2560 points of data, which can be divided into 20 frames, each with 128 data points. The threshold is obtained from summing 20 frames of noise output of Rayleigh distribution. The value obtained is 185, as discussed in detail in the next Chapter (Section 13.9 and listed in row 1 of Table 13.1). The probability of detection is obtained from 1000 runs at a constant input power level. Each run has a random input frequency from 141 to 1140 MHz with random initial phase; the results are shown in Figure 12.5. When the signal to noise (S/N) ratio is -11.3 dB, the probability of detection is about 90%. This value is about 0.4 dB less sensitive than a CW signal, which is shown in row 8 of Table 13.2. This result is expected because, among the 20 frames, if there is a phase transition in a certain frame, the amplitude in frequency domain may decrease slightly since the energy will spread to adjacent bins. As the result, the maximum summed output also decreases slightly. Since the sensitivity degrades only slightly, it is considered that signal detection is not a major issue.

The next problem is to determine whether the detected signal is a CW or a BPSK. One way is to estimate the signal length and use all the data available to make the decision. The other approach is to use frames of data to make the decision. This approach is more likely to be used in receiver designs, as it results in a quicker decision. The approach using all the data is discussed first.

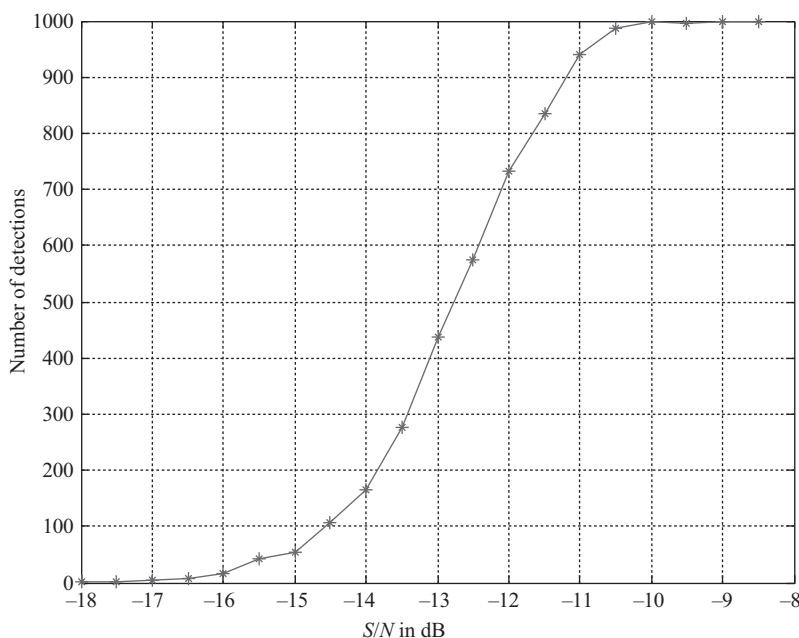


Figure 12.5 Probability of detection of a BPSK signal with a random input frequency from 141 to 1140 MHz and initial phase.

12.6 Using Long FFT to Determine the Existence of a BPSK Signal

This detection method is based on long FFT operation and may take special effort to implement in a receiver. If a relatively long signal is detected from the method in the previous section, the next step is to determine whether the signal is CW or BPSK. Let us use 2560 points as the length of the input signal. It should be noted that 2048 is a base 2 number (2^{11}) but 2560 is not. In order to be consistent, a 2560-point FFT will be used. As discussed in the previous section, the actual signal is shorter than 2560 points and the beginning and ending of the signal are padded with signals rather than zeros. The outputs in the frequency domain are shown in Figure 12.6. In this figure, the S/N ratio is set at 100 dB to eliminate the noise effect and a Blackman window of 2560 points is applied to the input data. The input frequency is randomly selected between 141 and 1140 MHz with a randomly selected initial phase. In order to magnify the shape of the output, only the peak value of the FFT output and ± 20 frequency bins are shown.

Figures 12.6(a) through 12.6(c) show the frequency outputs of a BPSK signal with different input frequencies. Although a Blackman window is used, the sidelobes are high because the spectrum of the signal is spread. For a BPSK signal, if there are equal plus and minus cycles, the carrier frequency will have a minimum. As discussed in the previous section, the carrier frequency of the spectrum is not suppressed. Figure 12.6(d) shows a CW signal with random input frequency and initial phase. This plot shows that the mainlobe is narrower and the sidelobes are relatively low. From these plots, a criterion is selected to

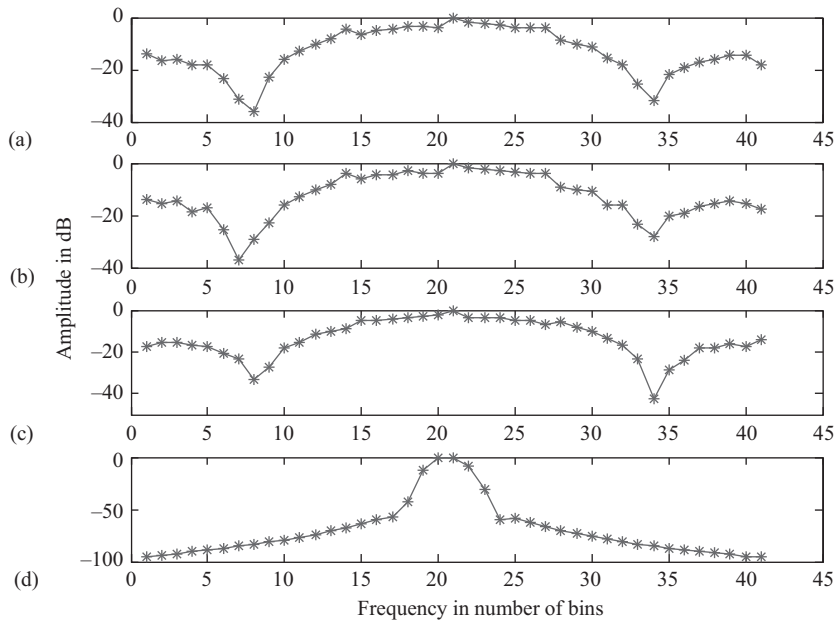


Figure 12.6 Frequency outputs from BPSK and CW signals: (a–c) BPSK signals with random frequency and initial phase and (d) CW signal.

differentiate a CW from a BPSK signal. First, select the maximum output in the frequency domain and four neighboring bins on each side of the maximum. This selection of nine frequency bins is determined empirically. Among these nine elements, the ratio of the minimum to maximum amplitude is calculated. Let us refer to this ratio as the minimum-to-maximum ratio for simplicity. If the signal is a BPSK, the minimum-to-maximum ratio should be large because the frequency bins all have relatively high values.

A threshold is set for this detection. First, the threshold is set by using noise as the input, and there are 100,000 runs. For each run, 2560 points of noise input data are multiplied by a Blackman window. Through a 2560-point FFT operation 1280 independent outputs in the frequency domain are generated. However, when the maximum occurs less than bin number 5 or higher than 1275, one cannot select four bins on either sides of the maximum. These runs are ignored and the total number of runs used in this specific case is 98,560. In other words, 1440 ($= 100,000 - 98,560$) runs are discarded. The maximum value is 0.674 and the threshold obtained from a Rayleigh distribution approximation is about 0.73. Although the plot in this figure is not a Rayleigh distribution, it is used to determine the threshold for convenience.

This threshold is rather high and will not be used to determine the existence of a BPSK signal since the purpose of the detection is to differentiate the BPSK from a CW signal, not to detect the existence of a signal in noise. If a CW signal is detected, the minimum-to-maximum ratio will be input S/N ratio dependent. For a strong signal its ratio will be low. Since the signal must be detected first and the sensitivity of detecting a BPSK signal is at a S/N ratio of -11.3 dB, this signal level is used to generate the threshold. The threshold generation procedure is as follows. A CW signal at $S/N = -11.3$ dB is used as an input and

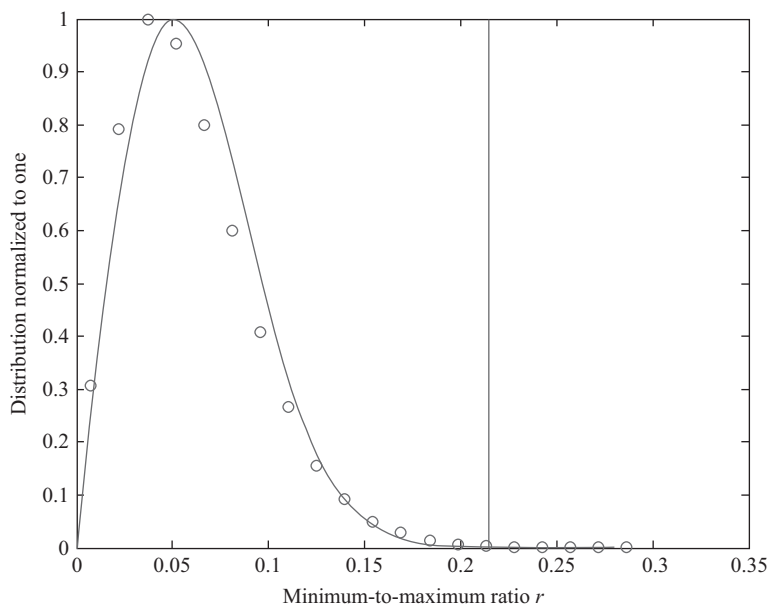


Figure 12.7 Minimum-to-maximum ratio from 100,000 runs with a Blackman windowed CW input at $S/N = -11.3$ dB.

generates 2560 data points with random input frequency and initial phase. After applying a Blackman window to the input data, an FFT operation is performed. The maximum amplitude frequency bin with four outputs on either side is selected and the minimum-to-maximum ratio is calculated. The result is shown in Figure 12.7 with 100,000 runs of random input frequency from 141 to 1140 MHz and random initial phase. With these frequency inputs, the maximum never occurs below the 5th or above the 1275th frequency bin, thus the plot contains 100,000 data points. The output is approximated by a Rayleigh distribution. With a probability of false alarm of 10^{-4} , a threshold of about 0.214 is obtained, as shown in Figure 12.7, and this threshold is obtained from three runs averaged. Another way to obtain the threshold is sorting the output value in descending order and choosing the 10th one, and the value obtained from three runs averaged is 0.243. Since the Rayleigh distribution does not match the input data well, 0.243 is used as the threshold, which is much smaller than the threshold of 0.73 obtained by noise input. This threshold is only for a CW signal at $S/N = -11.3$ dB. When the input S/N increases, the threshold is lower, and when the S/N decreases, the threshold increases.

The next step is to find the probability of detection. The input S/N changes from -12 to -3 dB and the resulting probability of detection is shown in Figure 12.8. For 90% detection of a BPSK signal, the S/N is at about -8.9 dB. From a few different runs, the probability of detection is very close to this value. These results can be explained as follows. When the input is a BPSK signal at S/N above -8.9 dB, about 90% of the BPSK signals can be recognized by this method of differentiation. If the signal is stronger, it has a greater chance of being recognized. If a strong CW signal is detected, it has a lesser chance of being

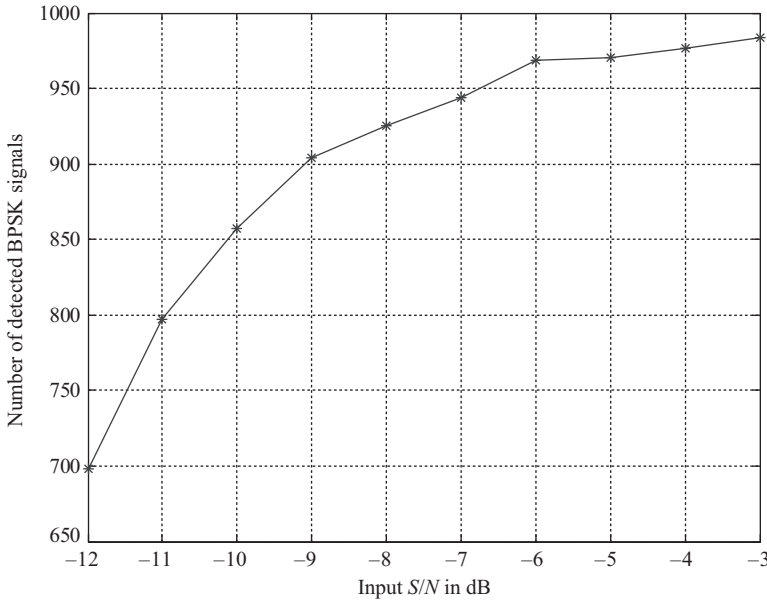


Figure 12.8 Probability of identifying a BPSK signal using the minimum-to-maximum ratio of FFT outputs.

classified as a BPSK signal. If a weaker CW signal is detected, it has a greater chance of being classified as a BPSK signal, a false detection.

12.7 Differentiate a BPSK Signal from a CW Signal Through Eigenvalue Calculation

It is known that eigenvalues can be used to determine the number of signals [9–12] and the signal modulation type [13]. In the following study, eigenvalues will be used to differentiate a BPSK signal from a CW signal. A 2×2 correlation matrix of the input signal with N data points can be obtained as

$$\begin{aligned}
 R(1, k) &= \begin{bmatrix} z(k-1)^* & z(k)^* & \dots & z(N)^* \\ z(0)^* & z(1)^* & \dots & z(N-k+1)^* \end{bmatrix} \begin{bmatrix} z(k-1) & z(0) \\ z(k) & z(1) \\ \vdots & \vdots \\ z(N) & z(N-k+1) \end{bmatrix} \\
 &= \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}, \quad (12.6)
 \end{aligned}$$

where 1 is the first lag, which is a fixed value, and * represents a complex conjugate. The second element k can be any value less than N . The window function cannot be used in the eigenvalue study because it spreads the spectrum and causes detection problems. Since the input data are real, one signal will affect two eigenvalues. In order to find out

whether the input contains a second signal (or one spread spectrum signal), three eigenvalues need to be obtained.

The three lags selection is based on a complex signal. The complex signal is generated by passing a real-valued signal through a 90-deg coupler, as discussed in Chapter 6. In that case, the complex signal generated has 561 data points. First the lags must be selected. The lags are selected through sequence search. For a 3×3 matrix the lags can be represented by $(1, x, y)$. A 3×3 correlation matrix is shown in (12.7):

$$R(1, x, y) = \begin{bmatrix} z(y-1)^* & z(y)^* & \dots & z(N)^* \\ z(x-1)^* & z(x)^* & \dots & z(N+x-y)^* \\ z(0)^* & z(1)^* & \dots & z(N-y+1)^* \end{bmatrix} \begin{bmatrix} z(y-1) & z(x-1) & z(0) \\ z(y) & z(x) & z(1) \\ \dots & \dots & \dots \\ z(N) & z(N+x-y) & z(N-y+1) \end{bmatrix}$$

$$= \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}, \quad (12.7)$$

where 1 is the first lag, which is a fixed value, and * represents a complex conjugate.

The input signal is a noiseless BPSK signal at an arbitrarily selected input frequency. The x value changes from 2 to 200 and the y value changes from 3 to 200 for 561 points of input data. Both changes are in unit steps. The third eigenvalue is monitored in every step and there are 39,402 outputs. The highest value occurs at $x \approx 45$ and $y \approx 115$, which is about 0.08 ($45/561$) and 0.2 ($115/561$) of the total data length. It appears that the maximum has a plateau and the values of x and y are not very critical. The input frequency of the BPSK signal is also not very critical. When the frequency changes from 141 to 1140 MHz, the values of x and y do not have large variation. Because of this, the x and y selected are about 0.08 and 0.2 of the total data length, however, the result was obtained from very limited testing.

The values of 0.08 and 0.2 are used for the lags, x and y , in this study. For 2560 points of input data, the three lags are 1, 205, and 512. When the input is noise, the largest eigenvalue is used to generate a threshold. The results from 100,000 runs are shown in Figure 12.9. A Gaussian distribution is used for approximation and the threshold is at 0.931 for a probability of false alarm of 10^{-4} . If the data are sorted in descending order, the 10th one is very close to 0.931. Because the Gaussian distribution provides a reasonable approximation, the threshold is obtained through this method.

The next step is to find the probability of detection. One thousand runs are performed with a random input frequency from 141 to 1140 MHz and the initial phase at certain S/N values. The S/N value changes from -8 to -3.5 dB and the result is shown in Figure 12.10. When the input S/N is at about -5.7 dB, the probability of detection is about 90%.

In a similar approach, instead of using noise, one can consider using CW signals at $S/N = -11.3$ dB to determine the threshold. When $S/N = -11.3$ dB, the probability of detection for a CW signal is about 90%. The similar method is used to find the three eigenvalues with the same lags. The minimum eigenvalue is used as the threshold rather than the largest eigenvalues for detecting whether the signal is a BPSK or not. The largest eigenvalue is used to detect the existence of signal regardless of its type. The threshold is obtained through a Gaussian approximation. With a probability of 10^{-4} , the threshold is

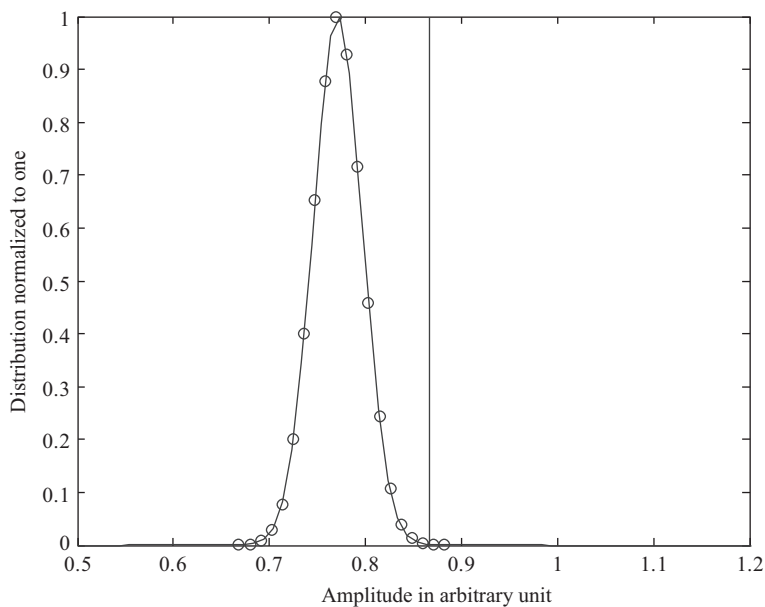


Figure 12.9 Amplitude variation of the largest eigenvalue in a 3×3 matrix with noise as the input from 100,000 runs.

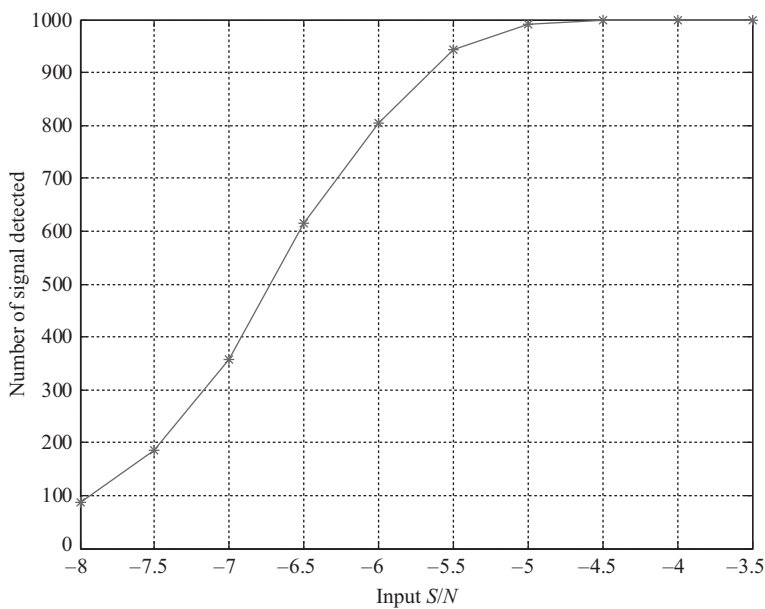


Figure 12.10 Probability of detecting a BPSK signal using eigenvalues of a 3×3 matrix.

about 0.899, which is slightly smaller than the threshold obtained through noise (which is 0.931). The corresponding S/N ratio requirement for a 90% probability of detection for a BPSK signal is about -6.6 dB if the threshold is determined with noise. If the CW signal is used to generate the threshold, the S/N ratio requirement is -8.9 dB.

Since the eigenvalue method can use the noise to generate a threshold but the FFT method cannot, the two results are different and can be explained as follows. There are several differences between Figures 12.8 and 12.10. First, the S/N ratio requirement for 90% detection is quite different (-8.9 and -5.7 dB, respectively). At $S/N = -8.9$ dB, it appears that the eigenvalue method cannot detect a BPSK signal at all. Second, when the input S/N is at -5 dB, the eigenvalue method can detect a BPSK signal with close to 100% correct detection, but the minimum-to-maximum ratio method can reach only about 97% correct detection. Third, the probability of false detection is different, but it cannot be shown in these two figures. For the minimum-to-maximum ratio method, the false detection depends on the CW signal strength. Detected weaker CW signal causes more false detection than a strong one. The eigenvalue method is somewhat like a conventional detection method in that with only noise input it has a very low probability of false alarm. Finally, the applications of these methods can be different. For the minimum-to-maximum ratio method, the signal must be detected through the summation method of the FFT outputs. Then this method is used to determine whether the signal is a CW or BPSK. The eigenvalue method can be used to detect the existence of a signal through the largest eigenvalue and also determines whether the signal is a CW or BPSK through the smallest eigenvalue. The conclusion is that these two methods produce different results and it is difficult to make a meaningful comparison.

12.8 Using Frame Outputs to Differentiate CW and BPSK Signals: FFT and Eigenvalue Methods

In the previous two sections, 2560-point FFT operation and correlation matrix are used to differentiate a CW from a BPSK signal. In this section similar operations are performed, but with only 20 input points from the 20 frames of FFT operations. There are 53 frequency outputs from one 128-point FFT operation, because among the 64 independent outputs only, the center portion is selected. From 20 frames of data there will be 1060 outputs (53×20). The prerequisite is that the signal must be detected through the approach mentioned in Section 12.5. In other words, the amplitude from the 53 frequency bins from 20 frame summations must be greater than the threshold of 185. Once a signal is detected, the corresponding frequency bin is determined, thus only 20 points are used in the operation. Two methods, the FFT and eigenvalue, will be used to differentiate a CW from a BPSK signal, and both methods require a threshold.

Before obtaining the threshold setting let us first examine the FFT outputs. A $1\text{-}\mu\text{s}$ CW signal with random frequency and initial phase is compared with a $1\text{-}\mu\text{s}$ BPSK signal with the same input frequency but different initial phase because the initial phase is randomly selected in the simulation. Both signals have no noise. The inputs are divided into twenty 128-point frames of data. A Blackman window of 128 points is applied to the signals and FFT operations are performed 20 times. The maximum amplitude summation of the frequency bins is selected and there are 20 points on this frequency bin. The FFT operation

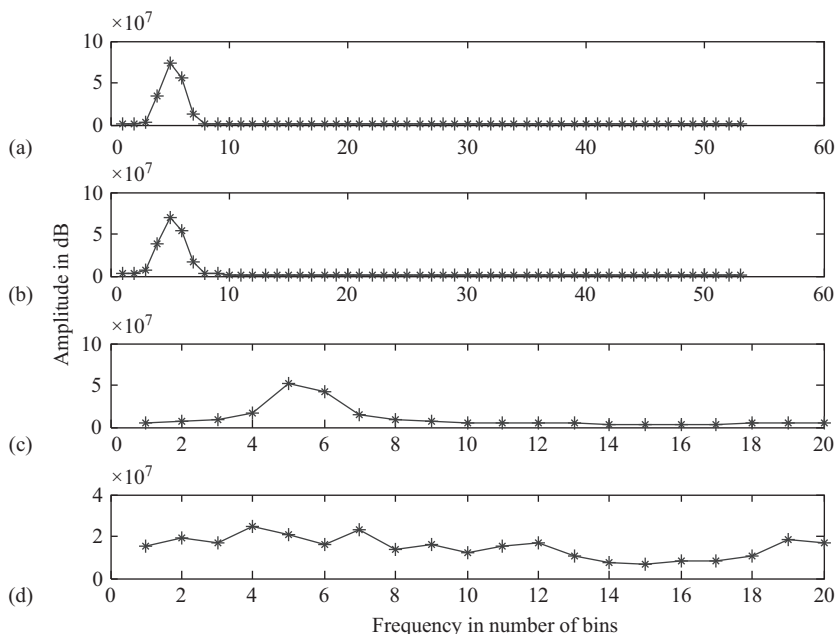


Figure 12.11 Magnitude spectrum of (a) the summation of 20 FFT results of a CW signal, (b) the summation of 20 FFT results of a BPSK signal, (c) the second FFT of a CW signal, and (d) the second FFT of a BPSK signal.

without a special window is performed on these 20 points of data; this operation is referred to as the second FFT operation. It should be noted that these are FFT outputs and they are complex. Since the total data length is $1\ \mu\text{s}$, the frequency resolution from the second FFT operation is 1 MHz. The results are shown in Figure 12.11. Figures 12.11(a) and 12.11(b) show the summation of the 20 FFT operations for a CW and a BPSK signal. As expected, these outputs are quite similar. Figures 12.11(c) and 12.11(d) show the second FFT outputs of a CW and a BPSK signal. In most cases the BPSK output is similar to the CW signal, or Figure 12.11(d) is similar to Figure 12.11(c). However, in this specific frequency the peaks are at different output bins, although the input frequencies are the same; the spectrum spreading is shown in Figure 12.11(d). From this observation, it appears that using the FFT method, it may be difficult to differentiate a BPSK signal from a CW signal.

To compare the performance of the FFT and correlation matrix using frame output, the threshold needs to be determined. The threshold setting of the FFT operation will be discussed first. The threshold is determined from a CW signal at $S/N = -11.3\ \text{dB}$. After the 20th frame of FFT operations, the sum of the highest output without comparing with the threshold of 185 is selected, which determines the frequency bins used as input to perform the second FFT operation. There are 20 complex data from this frequency bin. At the output of the second FFT operation, there are 20 outputs with a frequency resolution of 1 MHz and the highest output and its two adjacent frequency bins are selected. If the highest bin occurs at bin 1, bins 2 and 20 will be considered as the adjacent ones. If the maximum occurs at bin 20, bins 19 and 1 will be the adjacent bins. Among the three bins,

the ratio of the minimum to maximum value is used as the threshold. To find the threshold, 100,000 runs are performed with a CW signal at $S/N = -11.3$ dB, with a random input frequency and initial phase. The output cannot be easily fitted by a known distribution. If the highest value is used as the threshold, the threshold obtained from each of the 100,000 runs has a large variation. The approach used is to arrange of the results from 100,000 runs in an ascending order and the 99,990th value is used as the threshold. From this approach, the probability of false alarm can be considered as 10^{-4} . The ratio of the minimum to maximum value obtained from the average of 3 of 100,000 runs is about 0.838.

The threshold for the eigenvalue methods can be obtained in a similar manner. Since the first FFT outputs are complex, a 2×2 matrix can be used to find a second signal. Both the 2×2 and 3×3 matrices are evaluated. For the 3×3 matrix, the lag is [1, 2, 4], which is based on the argument in Section 12.7 with 20 data points. For the 2×2 matrix operation the lag is [1, 3] and the second lag is a point between 2 and 4. The threshold is determined by the ratio of the small eigenvalue to the large one. For the 3×3 matrix, the threshold is determined by the ratio of the minimum eigenvalue to the largest one. Both ratio distributions cannot fit easily in a known model. The threshold is determined through an approach similar to that discussed above. One hundred thousand runs with a CW signal at $S/N = -11.3$ dB are performed and the ratios are calculated. The 99,990th ratio is used as the threshold. The results are obtained from the average of 3 of 100,000 runs. The threshold for the 2×2 matrix is 0.551 and for the 3×3 matrix is 0.310. The three thresholds are listed in Table 12.1.

Once all the thresholds are determined, the probability of detection can be obtained. The input is a 1- μ s Barker coded signal with random frequency from 141 to 1140 MHz and random initial phase. The input S/N changes from -11 to -1 dB in 0.5 dB steps. At each S/N value, the input is tested 1000 times with different noise distribution. In this operation, the input must be detected first and the second step is to determine whether it is a BPSK signal. If the input signal is not detected, the result is considered as no detection.

The same input signal including noise is used for all three methods: FFT operation, 2-lag eigenvalue and 3-lag eigenvalue. For the FFT method the probability of detection changes between 10% and 45% without a clear relation to the input S/N . This result can be expected from the plot in Figure 12.11, and the threshold of 0.838 is a very high value.

Figures 12.12(a) and 12.12(b) show the results for the 2-lag and 3-lag eigenvalue methods, respectively. It appears that the 2-lag eigenvalue method can achieve 90% detection at about $S/N = -10.5$ dB and near 100% at -4 dB. From Figure 12.12(b), let us assign $S/N = -8$ dB as the S/N requirement for 90% detection, but it never approaches 100% detection.

All the sensitivities obtained with the threshold generated from a weak CW and the probability of false alarm at 10^{-4} are listed in Table 12.2. It appears that the eigenvalue method using frames of data and 2 lag provides the best result. It is both interesting and fortuitous that this is also the method most suitable for receiver design.

Table 12.1 Thresholds for differentiating CW against BPSK signals.

Method	Threshold	Ratio
FFT	0.838	Minimum to maximum of three FFT outputs at the largest output
Eigenvalue 2×2	0.551	Smaller to larger eigenvalue
Eigenvalue 3×3	0.310	Minimum to maximum eigenvalue

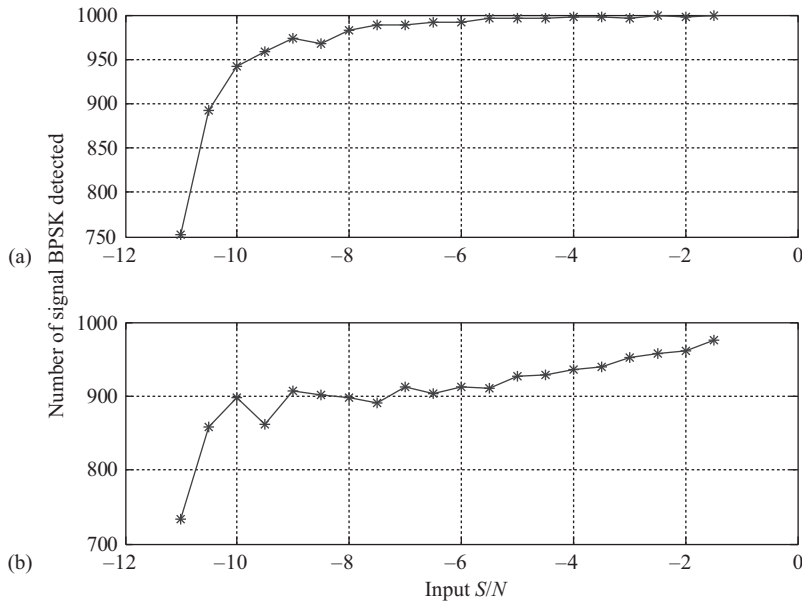


Figure 12.12 Probability of detection from the eigenvalue methods: (a) 2×2 matrix and (b) 3×3 matrix.

Table 12.2 Sensitivity obtained from different approaches.

Method	Sensitivity S/N (dB)
Long FFT operation	-8.9
Eigenvalue method using long data	-5.7
Eigenvalue method using frames of data 2 lag	-10.5
Eigenvalue method using frames of data 3 lag	-8.0

12.9 Signal Generation for Detection of Phase Transitions

The detection and classification methods discussed in the previous section use long data, and special operation is needed in receiver designs. The following methods using two frames of data are more likely to be included in receiver design. In addition, if a phase transition can be detected, it is likely that the location of the transition can be estimated.

In this study, the frame time is 50 ns and the chip time is selected to be longer than the frame time to avoid two phase transitions in one frame. Although theoretically with a fast chip rate one frame time may contain two phase transitions, this case will not be discussed because it is anticipated in future receiver design that the frame time may be shorter than the anticipated chip time. In order to study the general phase transition combinations, two cases of signals will be included. These cases should include all possible combinations of phase transition in frames. One is a single phase transition in a certain frame and in the leading and trailing frames there are no transitions. The other case includes two phase transitions in two

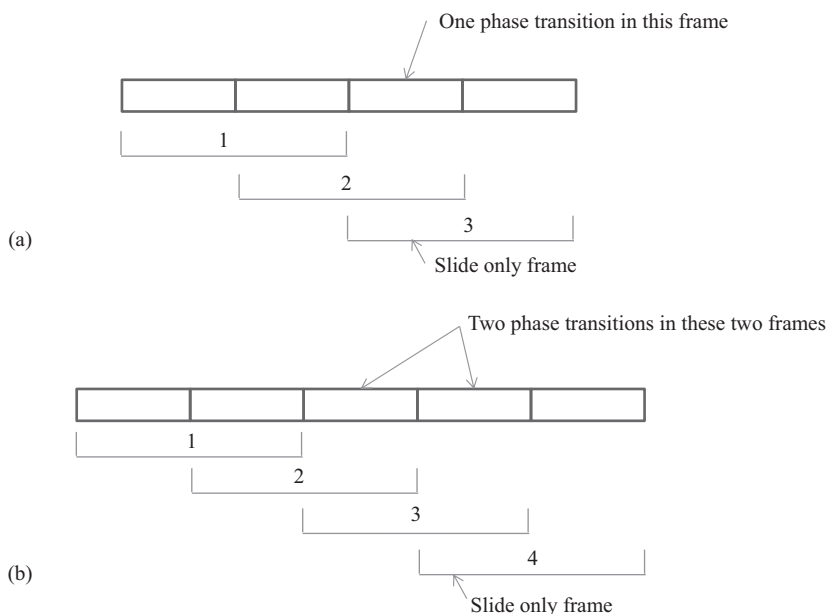


Figure 12.13 Locations of phase transitions: (a) one phase transition and (b) two phase transitions in consecutive frames.

consecutive frames. These two cases are shown in Figure 12.13. Figure 12.13(a) shows that the phase transition occurs at the third frame and Figure 12.13(b) shows that the two phase transitions occur at the third and fourth frames. In either case, the phase transition can occur in any location in a frame, such as close to the center or the edges. When two phase transitions are separated by one frame, such as in frames 2 and 4, the situation can be represented as in Figure 12.13(a). When three transitions occur in three consecutive frames, it can be represented as in Figure 12.13(b).

In order to study these two cases, it is desirable to generate a BPSK signal for testing. The BPSK generator program should generate BPSK signals based on the following requirements:

1. Input frequency with initial phase.
2. Total number of frames.
3. Signal strength in terms of S/N ratio.
4. Locations of phase transition in terms of frame number, such as in frames 2 and 3. A frame with a phase transition is referred to as the transition frame for simplicity.
5. Location of the phase transition within a frame, such as at the beginning or middle of a frame, and refer to it as the in-frame transition point. In the actual program, only six in-frame transition points will be provided: 0, $\frac{1}{4}$, $\frac{1}{2}$, $\frac{3}{4}$, 1, and 2. Zero means the transition is close to the beginning of the frame; 1 means it is close to the end of the frame; $\frac{1}{4}$, $\frac{1}{2}$, and $\frac{3}{4}$ means the transition is close to $\frac{1}{4}$, $\frac{1}{2}$, or $\frac{3}{4}$ of the frame, respectively; 2 means the in-frame transition point is random in the frame. The RF carrier and the chip phase should be locked. Because of this requirement, the in-frame transition point is limited.

Without the RF and chip phase lock requirement, the transition can be put in any location in a frame on a sampling point. Assume that the RF carrier and chip are locked, the phase transition must occur close to the zero phase of the RF cycle and on a sampling point. Thus the phase transition can occur only near the desired location.

6. If phase transitions occur in two consecutive frames, the two phase transitions should be separated by more than one frame time. For example, if in the first frames the in-frame transition point is randomly assigned, the in-frame transition point in the second frame will also be randomly assigned but separate from the first one by at least one frame time. If the phase transition point in the first frame is close to the end of the frame through random selection, the transition in the second frame will be close to the end of the frame. However, under this condition the separation between the two transitions is allowed to be slightly less than one frame time because the RF and chip must be locked. Random phase transitions will be used for most of the following discussions.

This program is tested before it is used for actual simulation. The results are shown in Figure 12.14. Figure 12.14(a) shows a signal generated through the computer program with input frequency at 141 MHz with random initial phase and a total of 4 frames of data, or 512 points (128×4). The transition frames are 2 and 3, the frame transition points are both random, and $S/N = 100$ dB (no noise). The low frequency is used in this plot because the phase transition can be more easily shown than at a higher frequency.

Figure 12.14(b) shows the phase transition locations in frame 2 of 1000 runs with random input frequency and initial phase. The transition locations are between points 128 and 256, as indicated by the horizontal lines. Figure 12.14(c) shows the phase transition locations in

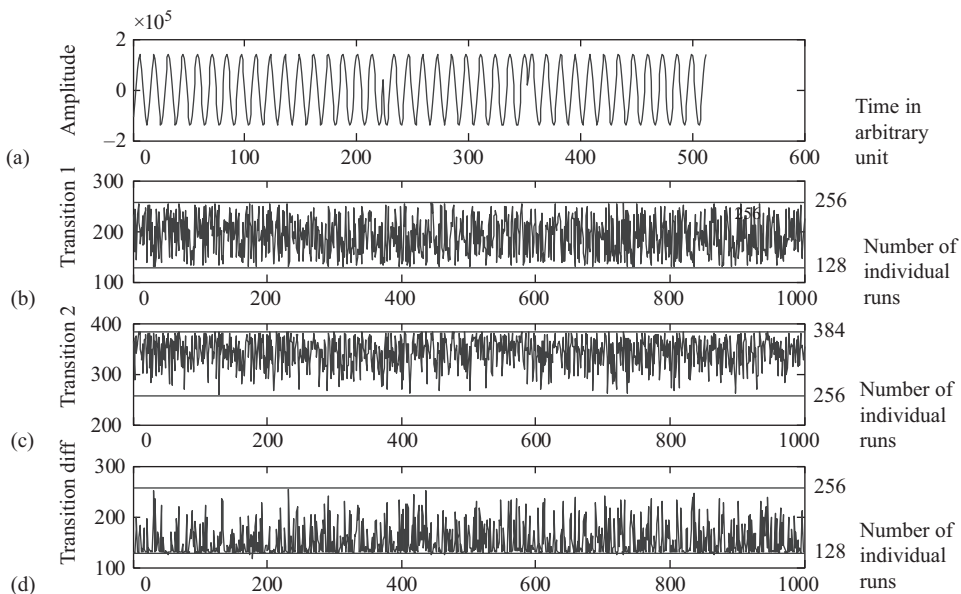


Figure 12.14 Simulation results of generating two phase transitions in two consecutive frames: (a) RF plot with phase transitions, (b) transition point in frame 2, (c) transition point in frame 3, and (d) difference in transition between frames 3 and 2.

frame 3, which are between points 256 and 384. Figure 12.14(d) shows the difference between phase transition locations in frames 3 and 2. These values should be between 128 and 256 points. However, there are few points where the difference is less than 128. These results indicate that all the requirements are fulfilled and the program will be used to generate signals for the following studies.

12.10 Approaches to Detect Phase Transitions

The rest of this chapter is dedicated to a discussion on detecting phase transition in a BPSK signal. Three methods will be introduced. They are the FFT operation, the eigenvalue calculation, and phase measured between two frames. The FFT and the eigenvalue methods measure the energy of a second frequency and will be referred to as the second frequency method. When the energy of the second frequency is high, it is classified as a spread spectrum signal. The phase comparison method uses two frames of data and measures the phase difference between consecutive frames.

For the second frequency detection method, one or two frames of data can be used as input. For one frame of data, when the phase transition is near the center of the frame, the detection sensitivity is highest because the spectrum is spread. When the phase transition is close to the end of a frame, the sensitivity is lowest because it behaves like a CW signal. For the two-frame case, the input data shift only one frame, as shown in Figure 12.13, and the least sensitive detection is when the phase transition is close to the center of a frame. Under this condition, the phase transition is close to $\frac{1}{4}$ of the total data. When the phase transition is close to the boundary between two frames, it appears at the center of the data and the detection is most sensitive.

The phase comparison method requires at least four frames of data, which will produce two double phase differences as a reference. If in three consecutive frames there is no phase transition, the double difference is close to zero. When a phase transition occurs near the boundary of two frames, the phase comparison method will be more sensitive. When the phase is close to the center of a frame, this method is less sensitive. Therefore the two methods—the second frequency and phase comparison methods—are complimentary and should be used to augment each other.

Since the second frequency detection methods are based on a similar principle, their performances will be compared. Two different input conditions will be used for the test: one and two frames of data. For the one-frame test, two frames of data will be used. The first frame contains no phase transition and is used to test false detection. The second frame contains one phase transition, which is randomly located in the frame.

For the two-frame test, four frames of data will be used and the phase transition is introduced in the third frame with random location, as shown in Figure 12.13(a). The first two frames are used to test false detection and frames 2 to 3 and 3 to 4 are used to detect phase transition.

Thresholds will be set for all cases. For the FFT operation, the threshold is set based on a CW signal at the sensitivity level. This idea is discussed in Section 12.6. For the eigenvalue approach, the threshold is set by noise input. This idea is presented in Section 12.7. Once a preferred method is selected, it will be used for further study. The input condition shown in Figure 12.13(b) will not be used for the present study.

12.11 The FFT Method on One Frame of Data

First the amplitude of the FFT output will be used as a detection method, which is similar to the concept in Section 12.5. This approach compares the spectrum outputs of a CW signal and a BPSK signal. For the one-frame test, one frame of input is used to determine the threshold. Before the threshold setting, the number of frequency bins will be selected for the evaluation. For a long input signal, Figure 12.6(d) indicates that nine frequency bins are a reasonable choice. For the one-frame data, the input frequency and initial are randomly selected without noise. A Blackman window of 128 points is multiplied to the input data, then an FFT operation follows. The highest frequency bin with four neighboring bins is selected. A sine wave with the same input frequency and initial phase is used for comparison. The results are shown in Figure 12.15. Figure 12.15(a) shows the output of the CW signal. Figure 12.15(b) shows the phase transition is close to $\frac{1}{4}$ of the frame and Figure 12.15(c) shows the phase transition is close to the center of the frame.

The difference between Figure 12.15(a) and 12.15(b) is small, but Figure 12.15(c) can be easily distinguished from Figure 12.15(a). This simple result shows that for an FFT operation with 128 data points it might be difficult to differentiate a CW from a BPSK signal when the phase transition is near $\frac{1}{4}$ of the frame. From Figures 12.15(a) and 12.15(b), it is difficult to determine the number of frequency bins that should be used for the study.

The following method is used to determine the number of frequency bins. Both seven and nine frequency bins are evaluated for both a CW and a BPSK signal without noise. The phase transition location is randomly selected in the frame for the BPSK signal. Ten thousand runs

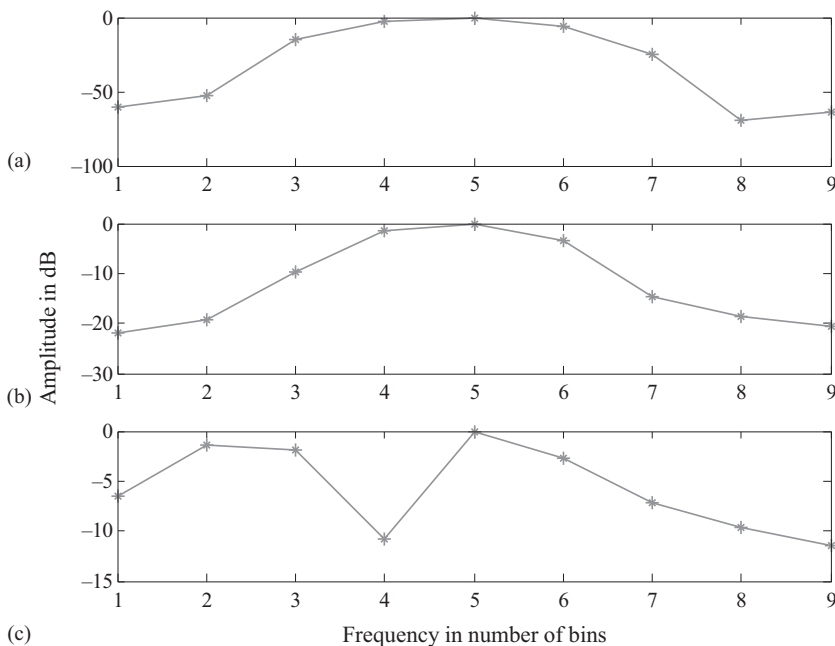


Figure 12.15 The highest frequency bin with four neighboring bins on each sides: (a) a CW signal, (b) a BPSK signal with phase transition close to $\frac{1}{4}$ of the frame, and (c) a BPSK signal with phase transition close to $\frac{1}{2}$ of the frame.

are performed. In each run, seven frequency bins are selected neighboring the maximum output and the minimum-to-maximum ratios are calculated for the CW and BPSK signals. The mean and standard deviation of these ratios are recorded. A similar operation is performed for the case of nine frequency bins. Equation (12.8) provides the selection criteria:

$$\begin{aligned} R_7 &= \frac{|m_{7c} - m_{7b}|}{\sqrt{s_{7c}^2 + s_{7b}^2}} \\ R_9 &= \frac{|m_{9c} - m_{9b}|}{\sqrt{s_{9c}^2 + s_{9b}^2}}, \end{aligned} \quad (12.8)$$

where m_{7c} and m_{9c} are the mean values of the ratios for 7 and 9 frequency bins on a CW signal, s_{7c} and s_{9c} are the standard deviations of the ratios for 7 and 9 frequency bins on a CW signal, m_{7b} and m_{9b} are the mean values of the ratios for 7 and 9 frequency bins on a BPSK signal, and s_{7b} and s_{9b} are the standard deviations of the ratios for 7 and 9 frequency bins on a BPSK signal.

The R values are the absolute value of the difference between the means divided by the mean square of the standard deviation for both the 7 and 9 frequency bin cases. If the difference between the CW and BPSK is larger, it is easier to differentiate them. For 10,000 runs, both R_7 and R_9 are close to 1, and for different runs, there are large variations. However, R_7 is always slightly larger than R_9 . Therefore it is easier to use seven frequency bins to identify a BPSK signal, so we use R_7 in this study.

The next step is to determine a threshold. A long BPSK signal can be detected at an S/N ratio of about -11.3 dB, and for one-frame detection, a CW signal can be detected at -1.5 dB (Table 6.6 in Tsui [8]). As discussed in Section 12.5, the threshold is set by using a CW signal at the sensitivity level to guarantee a 90% probability of detection. In this study the thresholds will be obtained at $S/N = -1.5$ dB. The threshold of 0.411 is obtained from a false alarm rate of 10^{-4} and the average of 3 of 100,000 runs. All the ratios are arranged in ascending order and the 10th value from the largest is used as the threshold. The probability of detection is presented in the next section.

12.12 Eigenvalue Method to Determine Phase Transition Using One Frame of Data and Comparison with the FFT Method

For the eigenvalue method, three eigenvalues are required to find a second frequency because the input signal is real. The lag value is selected based on the same argument in Section 12.7, and they are 0.08 and 0.2 of the total data length, thus the three values are 1, 10, and 26. First a threshold must be determined, which is obtained from noise input. The eigenvalues of the noise input are found with lags of 1, 10, and 26 and no window is applied to the input data. The largest eigenvalue is used for the threshold. A total of 100,000 runs are performed. The largest eigenvalue from each run is arranged in ascending order and the 99,990th value is used as the threshold; this value is 1.593. Since the largest eigenvalue is used as the threshold (not the ratio of the minimum to maximum eigenvalue), a CW is not required to generate the threshold.

For the detection test, two frames of data are generated. There is only one phase transition appearing in the second frame. The input frequency is randomly selected between 141 and 1140 MHz with the initial phase randomly selected within 0 and 2π . The location of the phase transition in the second frame is randomly selected. A total of 100,000 runs are performed with a constant input power level, which changes from $S/N = -3$ to $S/N = 26$ dB. For each run, two frames of data are processed. If a phase transition is detected in the first frame, it is considered a false detection because there is no phase transition in this frame. The detection in the second frame provides the probability of detection. The probability of detection results obtained with the FFT and eigenvalue methods are shown in Figures 12.16 and 12.17, respectively.

Figures 12.16(a) and 12.16(b) show the results from the FFT approach. At a low S/N value, there are usually a few false detections. When the S/N increases, there is no more false detection, as shown in Figure 12.16(a). In Figure 12.16(b), the phase transition occurs randomly in the second frame. It is interesting to note that when the S/N increases, the probability of detection decreases. The best probability of detection is close to 50 out of 1000 runs. When the phase transition in the second frame is near the center of the frame, which represents the best frequency spreading, the highest probability of detection is close to 15%. When the phase transition in the second frame is shifted toward the boundary of frames 1 and 2, the detection is reduced to zero from $S/N = -1$ to 26 dB. These results are not shown graphically. Thus the FFT operation on one frame of data is not suitable to detect phase transition.

Figure 12.17 shows the results of the eigenvalue method. Since the probability of false detection is about 10^{-4} , no false detection is measured in 1000 runs, as shown in

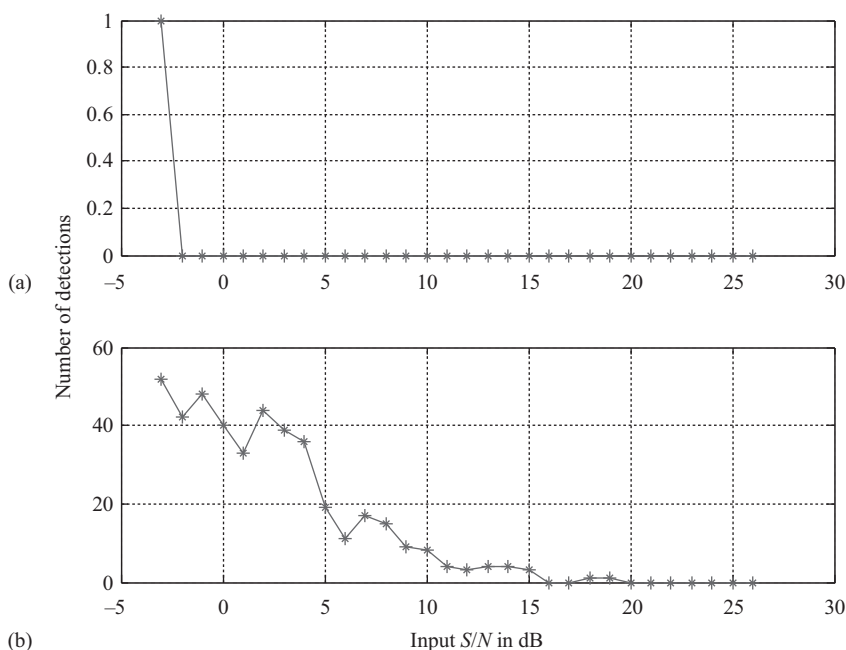


Figure 12.16 Number of detections of phase transition from the FFT method: (a) detection from frame 1 and (b) detection from frame 2.

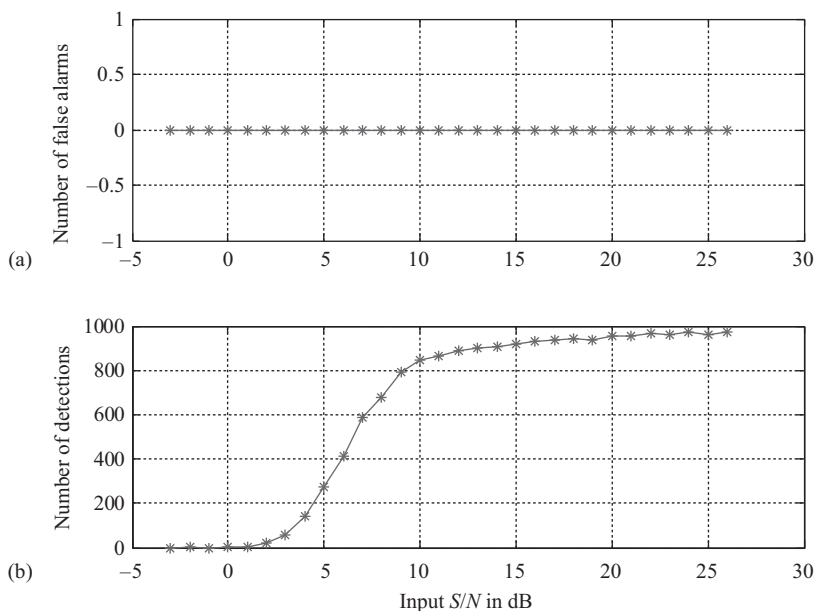


Figure 12.17 Number of detections of phase transition from the eigenvalue method using one frame of data: (a) detection from frame 1 and (b) detection from frame 2.

Figure 12.17(a). The probability of detection can be greater than 90% when the S/N is about 14 dB, as shown in Figure 12.17(b). However, even at higher S/N values the probability of detection cannot achieve 100% because the missed detection is caused by the location of the phase transition in a frame rather than signal strength.

When the phase transition is close to the boundary between frames 1 and 2 or at the end of frame 2, this detection method will not find the phase transition, which is the expected result. This result is illustrated in Figure 12.18. Figures 12.18(a) and 12.18(b) show the largest and the smallest eigenvalue versus the position of the phase transition. The starting point of the second frame is 129 and the ending point is 256. When the phase transition is close to the ends of the frame, the largest eigenvalue is relatively high and the smallest eigenvalue is relatively low. As shown in Figure 12.18, the eigenvalues cannot be used to detect the phase transition at the end of a frame. It can help explain the probability of detection shown in Figure 12.17.

From this study, it is concluded that the FFT output method is not suitable for detecting phase transition; the eigenvalue method is the preferred method. Even when the phase transition is close to the middle of the frame (the best scenario for the FFT method), the FFT method still does not produce a good probability of detection, therefore it will not be discussed further.

12.13 Eigenvalue Method to Determine Phase Transition Using Two Frames of Data

The eigenvalue method is very sensitive to detect phase transition in a certain part of a frame, therefore this method should be considered in receiver designs. However, when the phase

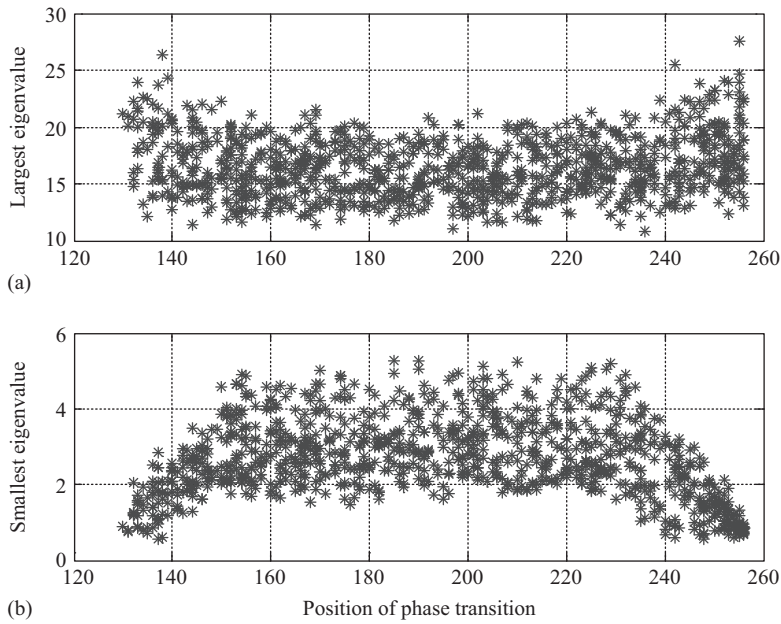


Figure 12.18 Eigenvalues versus the phase transition position: (a) largest eigenvalue and (b) smallest eigenvalue.

transition is close to the boundary, this method will miss detections. This section studies the same method but with two frames of data. In this study, four frames of data are generated and the phase transition occurs randomly in the third frame, as shown in Figure 12.13(a). Three sets of tests will be performed. The first set of data contains frames 1 and 2. Since there is no phase transition, it is used for the false detection test. The second and third sets of data contain data from frames 2 and 3 and frames 3 and 4. If detection is shown in either the second or the third set of data, or both of them, it is considered a phase transition is detected.

First, a threshold must be determined. The approach is similar to the approach in the previous section. With noise input and lag of 1, 20, and 52, a 3×3 correlation matrix is formed and the largest eigenvalue is found. One hundred thousand runs are performed and the 99,990th largest eigenvalue is used as the threshold, which is 1.29. The probability of false alarm is about 10^{-4} .

The results of the false and true detections are shown in Figure 12.19. These results are obtained from 1000 runs with random input frequency, initial phase, and a phase transition location in frame 3. Figure 12.19(a) shows no false detection, which is reasonable. Figure 12.19(b) shows that at a S/N ratio of about 6 dB, the detection probability can reach 90%. At higher S/N values, the probability of detection can achieve 100%, because when two frames of data are used, the worst phase transition location is close to $1/4$ and $3/4$ of a frame rather than at the frame boundary as for the single-frame case.

The eigenvalue method can determine whether there is a phase transition in a frame but it cannot determine the location of the phase transition. The following approach is considered to accomplish the location task. The idea is to measure the degree of spectrum spread in the data. It is expected that when the phase transition is close to the center of the data, the spectrum will have a wider spread than when the transition is close to the end. If one

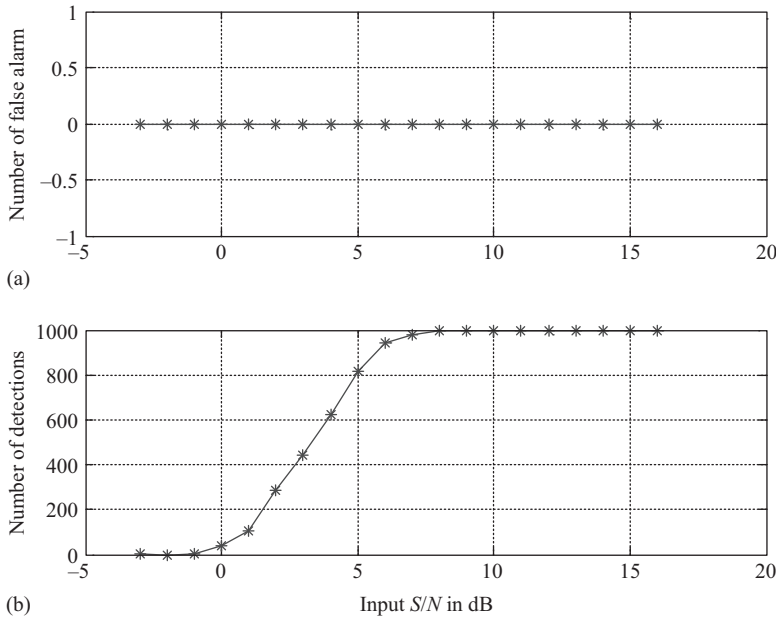


Figure 12.19 Number of detections of phase transition from the eigenvalue method using two frames of data: (a) detection from frames 1 and 2 and (b) detection from frames 2 to 4.

frame of data is used for the test, when the phase transition is close to the center of the frame, the third eigenvalue will be larger than the transition close to the end. However, using this method it still is difficult to differentiate, as explained in the following paragraph.

When two frames of data are used, the following approach is tried. Three frames of data are used for the test. A phase transition is introduced in the second frame. Two sets of eigenvalues are obtained and each set contains three eigenvalues. The first set uses data from frames 1 and 2, the second set uses data from frames 2 and 3. The eigenvalues are arranged in ascending order as e_{1i} , e_{2i} and e_{3i} , where $i = 1, 2$, referring to eigenvalues from sets 1 and 2:

$$\begin{aligned} e_{11n} &= \frac{e_{11}}{\sqrt{e_{21}^2 + e_{31}^2}} \\ e_{12n} &= \frac{e_{12}}{\sqrt{e_{22}^2 + e_{32}^2}} \\ e_d &= e_{11n} - e_{12n}, \end{aligned} \quad (12.9)$$

where e_{11n} can be loosely considered as the smallest eigenvalue normalized (subscript n) to the two large eigenvalues. Since one input will affect the amplitudes of two eigenvalues, their square root of the sum of the squared values is considered to be related to the signal amplitude, e_{12n} is for the second test. The difference $e_d (= e_{11n} - e_{12n})$ is plotted against the phase transition location in Figure 12.20. In these plots, 1000 runs are performed at two different input power levels. In Figure 12.20(a) the input S/N is at 10 dB and in Figure 12.20(b) the

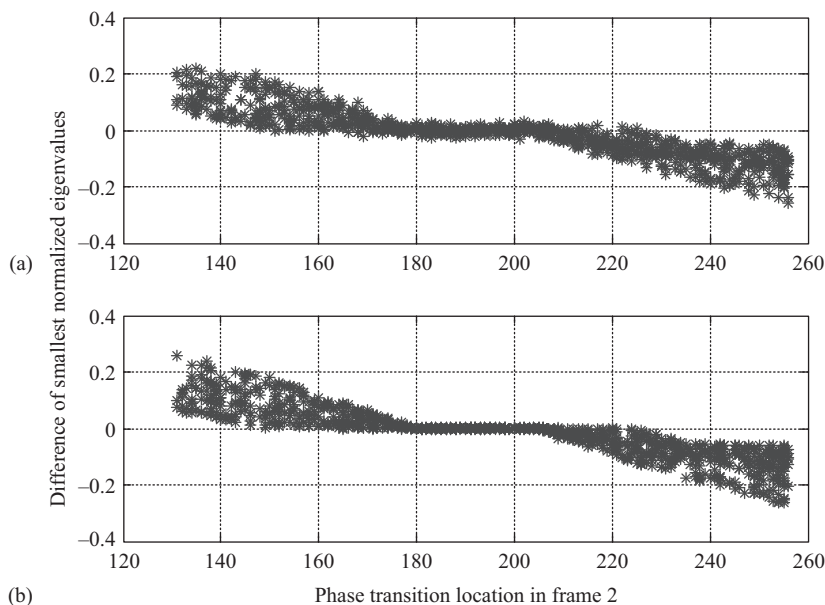


Figure 12.20 Phase transition location versus difference of normalized smallest eigenvalues: (a) $S/N = 10$ dB and (b) $S/N = 30$ dB.

S/N is at 30 dB. These curves have a similar pattern in that the difference is positive when the phase transition is close to the beginning of the frame and is negative when the phase transition is close to the end of the frame. However, these plots show that the value of e_d spread over a wide range, so it is difficult to accurately determine the phase location from the difference of the normalized smallest eigenvalues. We will need to find another approach to locate the phase transition in a frame.

12.14 A Phase Comparison Method to Determine Phase Transition

As described previously, the eigenvalue method has difficulty detecting a BPSK signal when the phase transition happens near the frame edge and a receiver is unable to predict the location of phase transitions. To solve this issue, we investigate the effectiveness of the phase measuring method for detecting a BPSK signal. For this study, four frames of data are used in order to generate three difference phases. The phase transition is introduced in the third frame and its location is close to the center of the frame. With this arrangement between frames 1 and 2 there is no phase transition between frames 2 and 3, while between frames 3 and 4 there is a phase transition. The input S/N is 100 dB and the frequency is randomly selected between 141 and 1140 MHz with random initial phase between 0 and 2π . Each frame of data is multiplied with a Blackman window and followed by the FFT operation. The frequency bin with the highest output in the first frame is selected. The phases at this

frequency bin from four frames are designated as θ_1 , θ_2 , θ_3 , and θ_4 . The differences and double differences can be written as

$$\begin{aligned}\theta_{21} &= \theta_2 - \theta_1 \\ \theta_{32} &= \theta_3 - \theta_2 \\ \theta_{43} &= \theta_4 - \theta_3 \\ \Delta\theta_{21} &= \theta_{32} - \theta_{21} \\ \Delta\theta_{32} &= \theta_{43} - \theta_{32}.\end{aligned}\tag{12.10}$$

Let us refer to $\Delta\theta_{21}$ and $\Delta\theta_{32}$ as the double difference phase angles. Since there is no phase change in either frame 1 or 2, the phase θ_{21} can be considered as obtained from a CW signal and can be used to obtain fine frequency. If the phase transition occurs near the boundary of frames 3 and 4, one can consider that θ_{32} is also obtained from a CW signal and θ_{43} has a phase transition near their boundary. Under this condition, $\Delta\theta_{21}$ is close to zero degrees (or 2π) and $\Delta\theta_{32}$ is close to $\pm\pi$ and these results are shown in Figures 12.21(a) and 12.21(b). These results are obtained from 100 runs and match the predications.

The results can be explained as follows. When the input is a CW signal the phase difference at a specific frequency between two consecutive frames is constant. The double

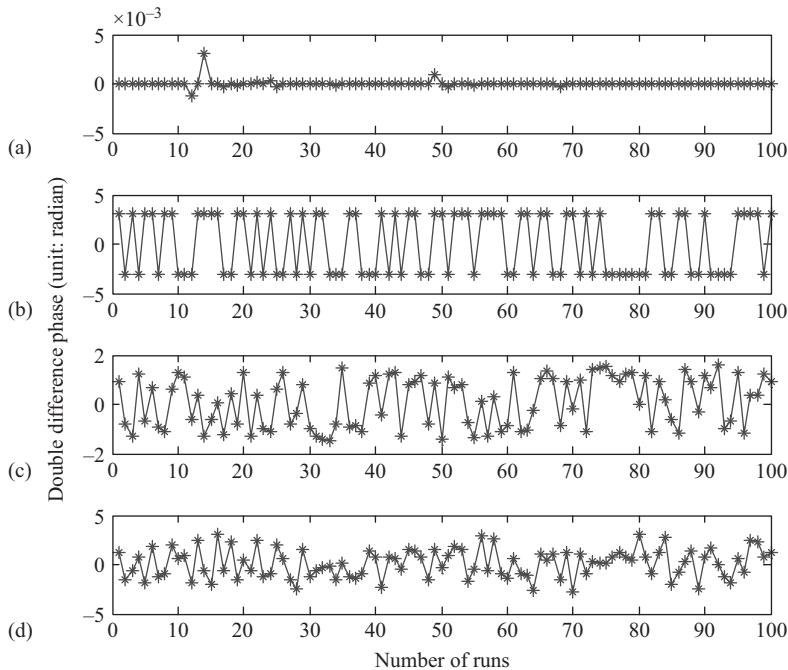


Figure 12.21 Double difference phase plots for $\Delta\theta_{21}$ and $\Delta\theta_{32}$: (a) $\Delta\theta_{21}$ for phase transition near the boundary of frames 3 and 4, (b) $\Delta\theta_{32}$ for phase transition near the boundary of frames 3 and 4, (c) $\Delta\theta_{21}$ for phase transition near the middle of frame 3, and (d) $\Delta\theta_{32}$ for phase transition near the middle of frame 3.

difference between them is close to 0 deg. If there is a phase transition, close to the boundary between two consecutive frames, the double difference between them is close to $\pm\pi$.

This approach can be applied as follows. First, the phase differences between two consecutive frames are calculated. The double difference phases from consecutive outputs are then calculated and three frames are required to calculate one double difference. If there are no phase transitions in these three frames, the double difference is close to 0, otherwise it is close to $\pm\pi$.

When the phase transition occurs close to the center of frame 3, the results are shown in Figures 12.21(c) and 12.21(d). In these figures the input conditions are similar to the results generated in Figure 12.21(a). The only difference is that the phase transition occurs close to the center portion of frame 3 instead of at the end. Under this condition there is no phase change at θ_{21} , but there are phase changes at θ_{32} and θ_{43} . These two plots show that although the phase transitions are shown, they are not as clear as in Figure 12.21(b). Therefore it is concluded that when the phase transition is near the center of a frame, the phase comparison method is not sensitive. This phenomenon is opposite to the one-frame eigenvalue method, which is sensitive when the phase transition is close to the center of a frame. Thus these two methods complement each other.

The next step is to run a test on the probability of false and true detection. For this simulation, five frames of data are required to generate four difference phases and three double difference phases. The signal frequency bin is found in the first frame by locating the bin with the maximum amplitude. The phases at the frequency bin from five frames are labeled as $\theta_1, \theta_2, \theta_3, \theta_4$, and θ_5 . The four difference and three double difference phase differences are defined in the same format as (12.10). For example, $\theta_{54} = \theta_5 - \theta_4$ and $\Delta\theta_{43} = \theta_{54} - \theta_{43}$. The phase transition is in frame 4 at a random location in the frame. The double phase difference can change from -2π to 2π and the absolute value can be used to determine the phase change. Under this condition, if there is no phase transition, the double phase difference value is close to zero or 2π . When there is a phase transition, the double phase difference is close to π . The threshold can be set between $\pi/2$ and $3\pi/2$. If the double phase difference is between these two values a phase transition is detected, otherwise there is no phase transition.

If a phase transition is detected in $\Delta\theta_{21}$, it is considered a false detection. If either $\Delta\theta_{32}$ or $\Delta\theta_{43}$ (or both) detects a phase transition, a phase transition is found. One thousand runs are performed at a constant S/N value. The input frequency is randomly selected between 141 and 1140 MHz with random initial phase. The results are shown in Figure 12.22. Figure 12.22(a) shows the false detection. When the input signal is strong, the number of false detections is almost zero. Figure 12.22(b) shows the detections of phase transition. The number is above 940 but less than 980 for the entire range of S/N values in this plot. One hundred percent detection cannot be achieved because the phase comparison method is insensitive when the phase transition is close to the middle of a frame. When the phase transition is close to either end of frame 4, the probability is 100% and the results are not shown.

For the phase comparison method, each double phase difference contains three frames of data. For example, $\Delta\theta_{21}$ contains data in frames 1, 2, and 3; $\Delta\theta_{32}$ in frames 2, 3, and 4; and $\Delta\theta_{43}$ in frames 3, 4, and 5. Thus if a phase transition is detected in a certain double difference frame, it is difficult to determine which frame of data contains the phase transition. If this method is used in a continuous mode, the phase transition can be found. For example, if there is no phase transition in $\Delta\theta_{21}$ it means there is no phase transition in frames 1, 2, and 3.

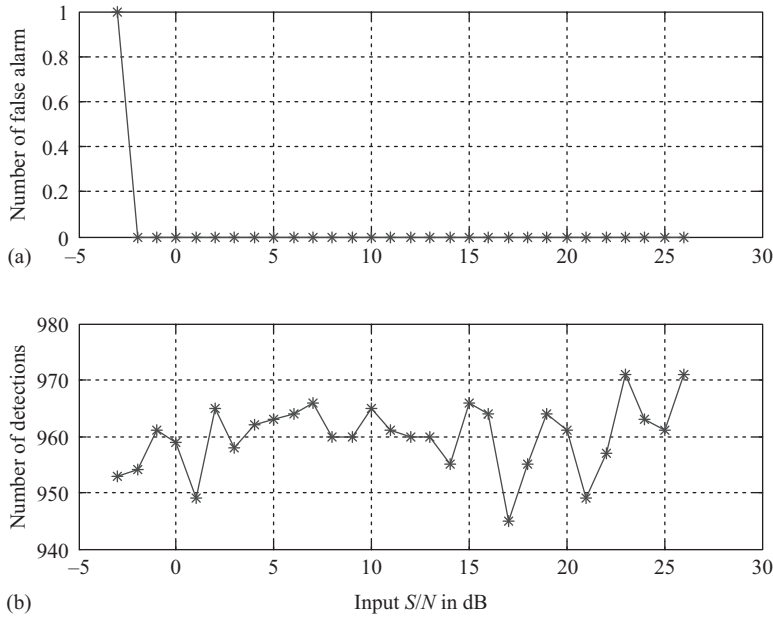


Figure 12.22 Results from the phase comparison method with 1000 random input frequencies and initial phases: (a) false detection and (b) number of detections.

If a phase transition is detected in $\Delta\theta_{32}$, then frame 4 has a phase transition. However, this phase transition may also cause a phase transition in $\Delta\theta_{43}$.

One frame of data is used in three consecutive double phase difference calculations. For example, a phase transition in frame 3 will appear in $\Delta\theta_{32}$, $\Delta\theta_{43}$ and $\Delta\theta_{54}$. Thus using the double phase difference method it might be difficult to determine the frame with the phase transition. This problem can be solved with a constant phase term θ_c . This phase term can be obtained from two frames of data without phase transition. In other words, this is the phase difference between frames caused by a CW signal. Using this concept, one must assume that the phase transition occurs occasionally in a frame and most of the phase differences between two consecutive frames are constant and caused by a CW signal. By observing the phase differences from several frames of data, one can determine the θ_c , which might be obtained from the average of several frames. Once this value is determined, the phase difference can be modified as follows:

$$\begin{aligned}\theta_{43} &= \theta_4 - \theta_3 - \theta_c \\ \theta_{54} &= \theta_5 - \theta_4 - \theta_c,\end{aligned}\tag{12.11}$$

where θ_c is the phase difference between adjacent frames when a CW signal (i.e., no phase transition) is present.

In this equation, it is assumed that there are five frames of data and the phase transition occurs in frame 4. It will take a minimum of two phase differences to confirm the constant phase, which is shown in the first two equations of the above equation. This constant phase

will be subtracted from all following phase differences as shown in the third and fourth equations. If a phase transition is detected in θ_{43} , the phase transition will occur in frame 4 because frame 3 does not contain a phase transition. If a phase transition is detected in θ_{54} , the phase transition will occur in frame 4 or 5. Under this condition, one can arbitrarily choose that the phase transition is in the first frame (frame 4 in this case).

12.15 Using Both the One-Frame Eigenvalue and the Phase Comparison Methods

From Sections 12.12 and 12.14, it appears that the single frame eigenvalues method and the phase comparison method provide reasonably good results. Both methods can produce a high probability of detection, but not 100%. The one-frame eigenvalue method can detect phase transitions near the center portion of a frame, but is insensitive to phase transitions near the end of a frame. The phase comparison method has the opposite property; it is sensitive to detect phase transitions near the end of a frame but insensitive to phase transitions close to the center of the frame. Thus these two methods are complementary to each other. If both of them are used in a receiver design they should be able identify most of the frames with phase transitions. The one-frame eigenvalue method can determine the frame with the phase transition more easily. Once a phase transition is found through the eigenvalue method, it is assumed that in the frame there is a phase transition. If the center of a frame is the designed phase transition location, the maximum error should be close to a half frame when the phase transition occurs near the end of a frame.

When the eigenvalue method cannot detect a phase transition in two consecutive frames but the phase comparison method detects it, this means the phase transition is close to the boundary of the two frames. As discussed in the previous section, under this condition the phase transition is assigned in the first frame. Since the phase transition is close to the boundary, the error can be close to a half frame.

12.16 Input Data with Two Phase Transitions in Consecutive Frames

The combination of phase comparison and eigenvalue shows great promise in detecting phase transitions in a BPSK signal. In this section we investigate how this combination might function in scenarios when two phase transitions happen around the frame boundary.

If two consecutive frames both contain a phase transition, in general either the one-frame eigenvalue method or the phase comparison method can detect the phase transition because the probability of detection in both Figures 12.17 and 12.22 are rather high at a decent S/N value. However, if the transitions occur in two consecutive frames and on the boundary of the frames, such as in Figure 12.23(a), the eigenvalue method may not detect them or may detect only one of them. The phase comparison method should detect them in both the first and second tests, but cannot detect it in the third test.

In many cases one phase transition can be detected in two consecutive tests through the phase comparison method. Thus the phase transition detected in two consecutive tests does

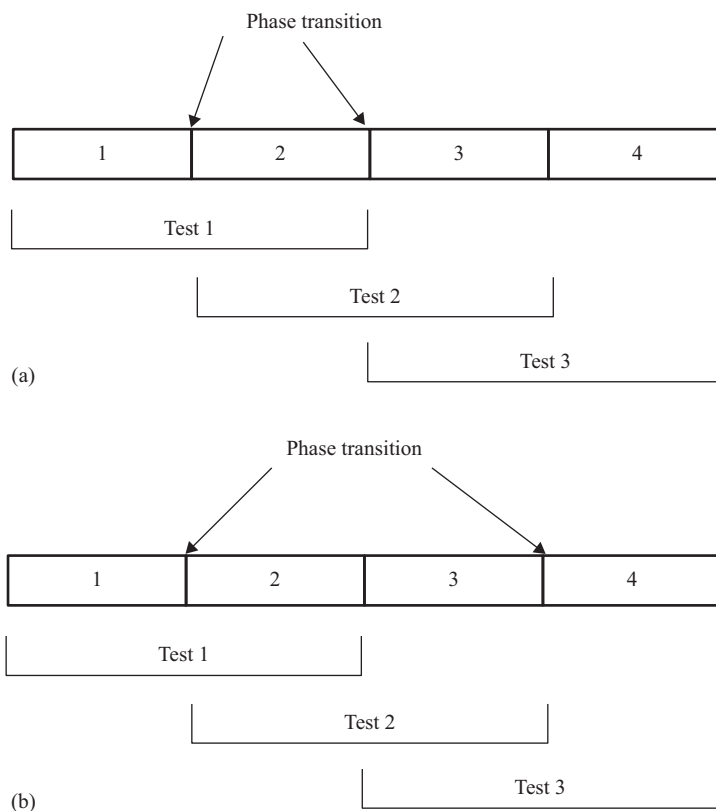


Figure 12.23 Phase transitions are close to frames 1 or 2: (a) phase transition separated by one frame and (b) phase transition separated by two frames.

not necessarily mean that there are two phase transitions in two consecutive frames, and the result from an eigenvalue can be used to determine the number of phase transitions. For the input condition shown in Figure 12.23(a), the following logic decision can be used.

1. If the eigenvalue method does not detect a phase transition in frames 1, 2, and 3, but the phase comparison method detects phase transitions in tests 1 and 2, two phase transitions can be assumed to occur in frames 1 and 2. This is the correct answer.
2. If the eigenvalues method detects a phase transition in frame 1 or frame 3 and the phase comparison method detects phase transitions in tests 1 and 2, two phase transitions can be assumed in frame 1 and 2 (for frame 1 detection) or frames 2 and 3 (for frame 3 detection). This also provides the correct answer.
3. If the eigenvalue method detects a phase transition in frame 2 and the phase comparison methods detect phase transition in frames 1 and 2, only one phase transition will be assumed in frame 2. Under this output condition, one phase transition will be missed. If the phase transitions separate by two frames, as shown in Figure 12.23(b), the eigenvalue method may not detect it but the phase comparison method should. The following conditions may occur and the following decision may be made.

4. If the eigenvalue method does not detect any phase transition, but tests 1 and 3 detect phase transition, the phase transitions are assigned to frames 1 and 3, which is the correct answer. If all three tests 1, 2, and 3 detect phase transitions, the phase transitions are also assigned to frames 1 and 3. The test 2 result will be ignored.
5. If the eigenvalue method detects phase transition in any of the four frames, the result from the phase comparison method including that frame will be ignored. Tests 1 and 3 will detect phase transition, but test 2 can be either way. The following four conditions will occur:

Eigenvalue detection is in frame 1—test 1 is ignored and test 3 will identify a phase transition in frame 3 independent of the test 2 result. The result is frames 1 and 3.

Eigenvalue detection is in frame 2—tests 1 and 2 will be ignored and test 3 will identify frame 3. The result is frames 2 and 3.

Eigenvalue detection is in frame 3—tests 2 and 3 will be ignored and test 1 will identify frame 1. The result is frames 1 and 3.

Eigenvalue detection is in frame 4—test 3 will be ignored and test 1 will identify frame 1 independent of the test 2 result. The result is frames 1 and 4.

All the outputs are considered correct. Thus, when two phase transitions are separated by two frames, the combined eigenvalue and phase comparison methods can obtain the correct results. The only possible problem occurs under the third condition, as under this condition the frame time and the chip time are approximately equal. One way to eliminate this problem is to increase the sampling frequency (or shorten the frame time) and separate the shortest chip time by more than 1 frame, such as 1.5 frames.

12.17 Conclusion

For a long BPSK signal containing many phase transitions, a windowed FFT approach is very effective to differentiate it from a CW signal through the spectrum width test. However, this method may not be a good approach in receiver design because special long FFT operation is required. If one frame of data is used to differentiate a CW and a BPSK signal, the FFT operation is not very effective since the spectra are similar due to the coarse frequency resolution with a short data length. If frames of data are used to differentiate CW and BPSK signals, the most effective approach is to use the eigenvalue method with a 3×3 matrix.

The eigenvalue method is very useful for a few frames of data. With one frame of data, the eigenvalue method can determine which frame contains the phase transition. However, when the phase transition is close to the ends of the frame, the signal is like a CW and the eigenvalue cannot detect it. If two frames of data are used, this problem can be eliminated. The deficiency of this approach is that it is difficult to determine which frame contains the phase transition.

The phase comparison method is an effective way to find a phase transition. It is more sensitive when the phase transition is close to the ends of a frame and less sensitive when the phase transition is close to the middle of the frame. Since two frames of data are required to compare the phase, it is difficult to determine which frame contains the phase transition.

If the eigenvalue method with one frame of data is combined with the phase comparison method, the phase transition can be detected very effectively because these two methods are complementary to each other. Properly combining the detections from both methods can

determine whether the frame contains the phase transition. Only in a very few cases will a phase transition be missed. If one can keep the frame time shorter than the chip time, the combined method is even more effective.

References

- [1] Haykin S. *Communication Systems*, 4th ed. New York: Wiley; 2001.
- [2] Gau J-Y. *Analysis of Low Probability of Intercept (LPI) Radar Signals Using the Wigner Distribution*. Master's thesis. Naval Postgraduate School, Monterey, CA; 2002.
- [3] Skolnik MI. *Introduction to Radar Systems*, 3rd ed. New York: McGraw-Hill; 2001.
- [4] Wen J. 'Parameter estimation in detection of BPSK radar signals'. *Proceedings of IET International Radar Conference*. Stevenage, UK: Institution of Engineering and Technology; 2009.
- [5] Borre K, Akos DM, Bertelsen N, Rinder P, Jensen SH. *A Software-Defined GPS and Galileo Receiver: A Single-Frequency Approach*. Boston: Birkhäuser; 2007.
- [6] Barker RH. 'Group synchronizing of binary digital sequences', in Jackson W (ed.), *Communication Theory*. New York: Academic Press; 1953:273–287.
- [7] Borwein P, Kaltofen E, Mossinghoff MJ. 'Irreducible polynomials and barker sequences', *ACM Communications in Computer Algebra* 2007;**162**(4):118–121.
- [8] Tsui J. *Special Design Topics in Digital Wideband Receivers*. Boston: Artech House; 2010.
- [9] Wax M, Kailath T. 'Detection of signals by information theoretic criteria', *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1985;**33**(2):387–392.
- [10] Wong KM, Zhang Q, Reilly JP, Yip C. 'On information theoretic criteria for determining the number of signals in high resolution array processing', *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1990;**38**(11):1959–1971.
- [11] Zhang QT, Wong KM. 'Information theoretic criteria for the determination of the number of signals in spatially correlated noise', *IEEE Transactions on Signal Processing* 1993;**41**(4):1652–1663.
- [12] Zeng Y, Koh CL, Liang Y-C. 'Maximum Eigenvalue Detection: Theory and Application'. *Proceedings of 2008 IEEE International Conference on Communications*. New York: IEEE; 2008:4160–4164.
- [13] De Vito L, Napolitano DD, Rapuano S, Villanacci M. 'Eigenvalue-based signal detector for an automatic modulation classifier'. *Proceedings of 2010 IEEE Instrumentation and Measurement Technology Conference*. New York: IEEE; 2010:1131–1136.

Appendix 12.A

```
% JT Oct 5 2010
% 1 us 13 length Barker data with code and the RF are phase
%   locked
% clear all
% close all
function [x, barker, n]=barker13_rf_chip_syn_1us_fct(fs, f1,
    theta, snrdb)
```

```

% x: digitized output
% barker: barker code
% n: total number of output points
% fs: sampling frq in Hz
% f1: input frq in Hz
% theta: initial input RF phase
% snrb: input S/N
ts=1/fs;
t_sig_len_ns=1000; %signal length in ns
t_rf_percycle=1/f1; % RF time per cycle
t_total_sig=t_sig_len_ns*1e-9; %Total data time in us
n=2560;
nn=0:n-1;
no_of_rf_cycle=t_total_sig*f1; % Total rf cycles
rf_perchip=floor(no_of_rf_cycle/13); %RFcycles/chip for 13
    bit Barker
t_data_len=rf_perchip/f1*13;
t_data_pt=floor(t_data_len/ts);
dif_data_pt=2560-t_data_pt;
t_per_chip=t_rf_percycle*rf_perchip;
t_data1=theta/(2*pi)*t_rf_percycle; % time for the 1st data pt
t_chip1=t_rf_percycle-t_data1+ts; % Equivalent time for initial
    phase
floor(t_chip1/ts);
amp=sqrt(2)*10^(snrdb/20);
for ii=1:15; %find phase transition time
    t_chip(ii)=t_chip1+(ii-1)*t_per_chip;
end
for ii=1:2560;
    t_data(ii)=(ii-1)*ts;
end
t_chip(1);
t_data(1:5);
%Building the Barker code
for ii=1:15;
    if ii==1;
        tmp(ii)=length(find(0<t_data & t_data <=
            t_chip(ii)))+1;
    elseif ii==15;
        tmp(ii)=length(find(t_data>t_chip(ii-1)));
    else
        tmp(ii)=length(find(t_chip(ii-1)<t_data &
            t_data<=t_chip(ii)));
    end
end
end

```

```

barker=[];
for ii=1:15;
    if ii==7 || ii==8 || ii==11 || ii==13;
        barker=[barker,-ones(1, tmp(ii))];
    else
        barker=[barker,ones(1, tmp(ii))];
    end
end
noise=randn(1,n);
x1=sin(2*pi*f1*ts*nn+theta);
len_bar=length(barker);
x=barker.*x1;
x=amp*x+noise;

```

Detection of Frequency Modulated Signals

13.1 Introduction

The purpose of this chapter is to use the digital receiver outputs to detect a frequency modulated (FM, or chirp) signal. Although there are different types of chirp signals, in this discussion only a linear chirp signal is considered because of its wide applications. A pulsed linear chirp signal, $x(t)$, can be represented as

$$x(t) = \cos\left(2\pi\left(ft + \frac{k}{2}t^2\right) + \phi_0\right) \text{rect}\left(\frac{t}{PW}\right), \quad (13.1)$$

where k is the chirp rate, ϕ_0 is the initial phase, and instantaneous frequency can be presented as $f + kt$. The function of $\text{rect}(t/PW)$ represents a rectangular window function whose width is PW . When k equals zero, the linear chirp signal regenerates into a pulsed continuous wave (CW) signal. The value of the chirp rate can be either positive or negative and the signal detection method presented in this chapter will function in either case, although we only present simulation results about the detection of chirp signals with a positive chirp rate. As a result, the chirp signal with sawtooth frequency modulation (i.e., the signal frequency increases and then decreases periodically) is not considered. The main focus of this chapter is on detecting the existence of a chirp signal and its chirp rate. The duration of the signal under consideration is 1 μs , but the proposed algorithm can be applied to signals with a longer or shorter duration. Also, to simplify discussions and focus on fundamental principles, only an ideal pulsed chirp signal is considered. In other words, it is assumed that the signal magnitude remains constant during the signal's duration and the effects of falling/leading edges are ignored.

To successfully detect chirp signals, two goals need to be achieved: 1) detecting the existence of a chirp signal and 2) finding the chirp rate or the initial and ending frequencies and the pulse width (PW). Although different techniques have been proposed for chirp signal detection [1–7], the major focus of this chapter is the FFT-based method because of its processing speed. Some discussion about using eigenvalue for FM signal detection will also be presented and compared with the FFT-based method.

For the FFT operation, amplitude and phase comparison methods can provide better frequency estimation for pulsed CW signals. From preliminary study (presented in this

chapter), it appears that the amplitude comparison method derived for detecting CW signal can still provide decent results for chirp signal detection, even when the chirp rate is very fast. These methods will be further studied with FM signal in this chapter.

We have demonstrated that the phase comparison method is used to find the fine frequency of a CW signal. The approach is to compare the phase of two consecutive time frames to obtain desired information. For a CW signal, the frequency may change from one frame to the next if the input signal is close to the boundary of two frequency bins. Under this condition, the maximum FFT outputs in the frequency domain stay at a fixed frequency bin or occasionally change to a neighboring bin. In the case of a chirp signal, the significant frequency shift between FFT frames might happen, therefore the maximum outputs from two consecutive frames may separate several frequency bins. Under this condition, the phase may wrap around several times; however, the fine frequency may still be usable. From simulation results, it has been shown that when the frequency changes fast, it appears that the phase comparison method can still be used to find the fine frequency.

A chirp signal is usually relatively long to take advantage of the large time–bandwidth product [8, 9]. The time–bandwidth product is defined as the product of the chirp signal's PW and frequency range (the difference between its starting and ending frequencies) and is referred to as the processing gain. For an electronic warfare (EW) receiver, it is desirable to process longer signals to improve detection sensitivity and generate an accurate estimate of signal characteristics. The desired information—detection of signal and chirp rate—will be obtained from the many frames of data. This chapter concentrates on this subject.

13.2 Potential Problems with FM Signal Detection

In a chirp signal, when the frequency changes fast, it may cause two potential problems for the intercept receiver. The first is that when the frequency change is large in one frame, there might be several frequency bins of comparable amplitude. For example, if the sampling frequency is 2.56 GHz and one frame contains 128 points, the time per frame is 50 ns and the corresponding frequency resolution is 20 MHz. Let us arbitrarily assume an FM signal with a maximum frequency change of 2000 MHz in 1 μ s (or chirp rate of 2000 MHz/ μ s). This can be considered a very fast chirp signal. Since the maximum bandwidth of the receiver is assumed to be 1000 MHz, the maximum PW processed by the receiver is limited to 0.5 μ s. There are 10 frames of data in 0.5 μ s and each frame is 50 ns long. Therefore the maximum frequency change within one frame is 100 MHz, which is five frequency bins. As a result, a unique maximum amplitude might be difficult to obtain.

Figure 13.1 shows the FFT output of an FM signal with a 100-MHz frequency change in 50 ns with random starting frequency and random phase. In this figure only one frame of data is examined. The outputs depend on the starting frequency of the chirp signal. As expected the spectrum has multiple peaks and a peak is difficult to identify.

When the FFT operation is used to find the input frequency, a window is often needed to reduce the sidelobes in the frequency domain. A Blackman window is used to multiply the input chirp signal, and the FFT is then conducted upon the windowed signal. The resulting spectra are shown in Figure 13.2. As shown in this figure, once the chirp signal multiplies a window, the maximum amplitude in the frequency domain, which corresponds to the signal's center frequency in the frame, can be easily identified.

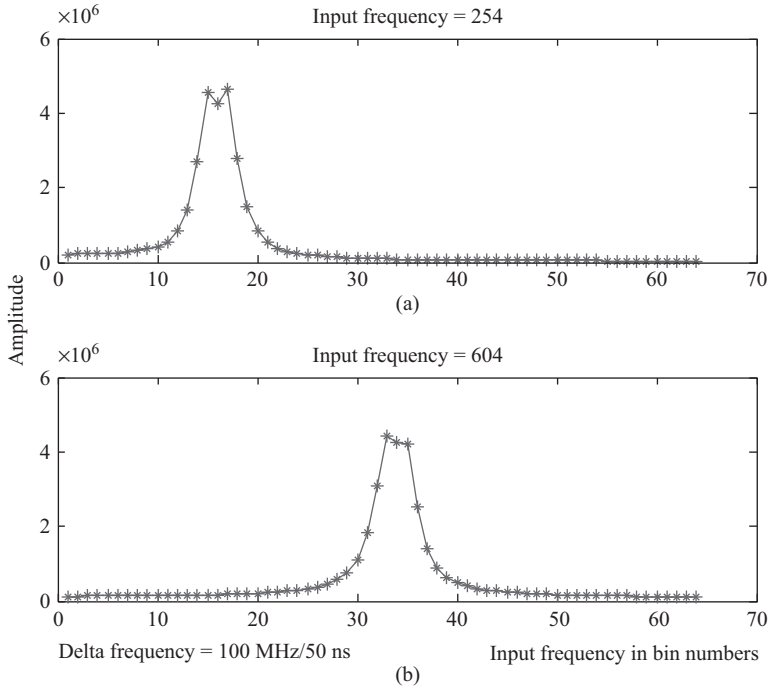


Figure 13.1 Amplitude of FFT outputs of an FM signal with 100-MHz delta frequency in 50 ns without noise: (a) with a starting frequency at 254 MHz and (b) with a starting frequency at 604 MHz (input frequency indicated in bin numbers).

13.3 Amplitude and Phase Comparison Approaches

Amplitude and phase comparison can be used to generate fine frequency estimation for CW signals, and readers can find the detailed treatment in Sections 11.6 and 6.11, respectively, of Tsui [10]. A brief introduction about both techniques is provided below.

13.3.1 Amplitude Comparison

After conducting a FFT upon one frame of the data, the frequency of the FFT peak (assuming the peak magnitudes are above the threshold) can be used as the coarse frequency estimation, say f . Assume X_0 is the maximum amplitude value of the FFT result and X_1 is the larger amplitude of its two adjacent frequency bins. The ratio between these two magnitudes, $r (= X_1/X_0)$, can be used to determine a frequency adjustment, f_{fine} , which is added to the coarse frequency estimation to improve the estimation accuracy. When the Blackman window is used, we set the signal frequency at one of the FFT bins and then vary the signal frequency from the FFT frequency bin and calculate the resulting r over the bandwidth of half of the FFT frequency resolution. In this way we can get a plot of r versus f_{fine} (the difference between signal frequency and the closest FFT frequency bin). The curve fitting can then be applied to determine a second-order polynomial in the format of $f_{fine} = a_2 r^2 + a_1 r + a_0$.

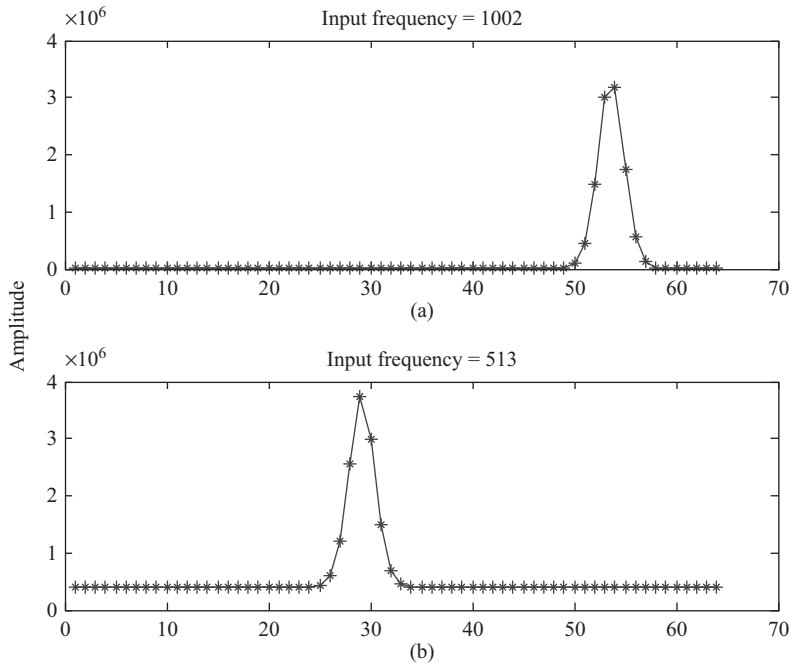


Figure 13.2 Amplitude of FFT outputs of an FM signal with 100 MHz in 50 ns and a Blackman window without noise: (a) with a starting frequency at 1002 MHz and (b) with a starting frequency at 513 MHz.

13.3.2 Phase Comparison

If two frames of FFT results are available and peak frequency bins of these two frames are the same, the phase difference between these two peaks, ϕ , is calculated. The fine frequency adjustment can then be determined by dividing the phase difference by the product of 2π and the time duration of one FFT frame. This operation can be found in (13.3) which will be explained in details later.

The amplitude and phase comparison techniques are developed for CW signal. Whether these methods are suitable for an FM signal is discussed in the next section. It appears that the results depend on the signal's starting frequency and delta frequency, which is defined as the difference between the signal's starting and ending frequencies. A Monte Carlo approach will be used in our investigation of amplitude and phase comparison. In our simulation, the sampling rate is set as 2.56 GHz and the frame size is 128 points (time duration 50 ns). In each simulation, two frames of chirp signal data are generated. The delta frequency and the starting frequency will be randomly selected. For two frames of data, the maximum delta frequency is 200 MHz, as stated in Section 13.2, and the minimum value can be zero, representing a CW signal. Once the delta frequency is randomly selected, the starting frequency can be arbitrarily chosen between 141 and 1140 MHz with a random initial phase. The following two equations can be used to choose the delta and the starting frequencies:

$$\begin{aligned} f_1 &= (999 - \Delta f)rand + 141 \\ f_2 &= f_1 + \Delta f, \end{aligned} \quad (13.2)$$

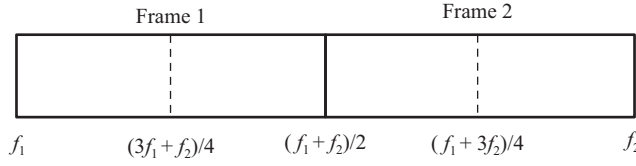


Figure 13.3 Frequencies of two frames of data.

where f_1 and f_2 are the starting and ending frequencies in MHz, $rand$ is a uniformly distributed random number whose range is (0, 1), and Δf is referred to as the delta frequency in the following discussions.

The two frames of data are shown in Figure 13.3. For the starting frequency of f_1 and the ending frequency of f_2 , the frequency at the edge of the two frames is $(f_1 + f_2)/2$. The frequencies at the centers of the two frames are $(3f_1 + f_2)/4$ and $(f_1 + 3f_2)/4$, respectively. When the amplitude comparison method is used to measure the frequency of frame 1 and frame 2, the expected frequencies are $(3f_1 + f_2)/4$ and $(f_1 + 3f_2)/4$, respectively. When the phase comparison method is used to measure the frequency, the expected result should be $(f_1 + f_2)/2$ since the time corresponds to the middle of the two frames. These are the criteria to compare the performance.

The phase comparison is similar to the one discussed in Section 6.11 in Tsui [10]. The angle difference ϕ between the two frames is calculated as

$$\begin{aligned}\phi &= \text{angle}(y_1 * y_2') \\ f_{fine} &= \frac{\phi}{2\pi \times 128t_s} \\ f_o &= \left[\frac{k_1 + k_2}{2} - 1 \right] \times 20 \times 10^6 + f_{fine} \quad \text{or} \\ f_o &= (k_1 - 1) \times 20 \times 10^6 + f_{fine} \quad \text{or} \\ f_o &= (k_2 - 1) \times 20 \times 10^6 + f_{fine},\end{aligned}\tag{13.3}$$

where angle is a MATLAB[®] (MathWorks, Natick, MA) function, y_1 is the maximum FFT output at frequency bin k_1 of frame 1 and y_2 is the maximum FFT output at frequency bin k_2 of frame 2, $'$ is the complex conjugate operator, t_s is the sampling period, f_{fine} is the fine frequency determined with the phase comparison method and f_o is the output frequency, which can be calculated as one of the frequencies at three different positions shown in Figure 13.3. It needs to be pointed out that the fine frequency can also be determined with the amplitude comparison method.

In this equation, k_1 and k_2 are two independent values. Although using the average of k_1 and k_2 to determine the frequency, as in the third equation in (13.3), is a reasonable estimate, there are some issues when a threshold must be determined to detect weak signals, as discussed in Section 13.10. Let us refer to this approach as the averaged index method. The final frequency can also be determined by either the fourth or the fifth equations in (13.3) and the frequency obtained should be close to the center of frame 1 or 2 rather than between the two frames. This will be referred to as the single index method. The distributions of the

difference frequency measured with noise input between two consecutive frames from these formulas are different. As a result, the thresholds determined by these two approaches—the average and the single index—are different. This issue is further discussed in Section 13.10.

For one chirp signal, three sets of frequencies are measured. First, the maximum frequency bin is used to determine the input frequency in each frame, which is referred to as the coarse frequency. This method generates two coarse frequencies at the middle of the two frames. The amplitude comparison also generates two frequencies at the middle of the two frames. The phase comparison method generates only one frequency at the boundary of the two bins. Their performance comparison is presented in the next section.

13.4 Frequency Measured from the Three Methods

In this section we compare the frequencies estimated from the three methods presented in the previous section: coarse frequency estimation with FFT, amplitude comparison, and phase comparison. In order to make a meaningful comparison, the input signal frequency at the frame boundary will be used as the reference. Referring to Figure 13.3, if the frequencies $(3f_1 + f_2)/4$ and $(f_1 + 3f_2)/4$ are measured by either coarse frequency estimation or amplitude comparison, their average will be obtained. The average frequency should be close to $(f_1 + f_2)/2$, which is the same input frequency as obtained by the phase measurement method. Stated differently, the frequency at the boundary of two frames is measured through three methods based on the same two frames of data. The estimated frequencies are then compared with expected values of $(f_1 + f_2)/2$ and the differences are referred to as the error frequency.

First, a CW input signal without noise is used as the input. The signal has a random frequency from 141 to 1140 MHz with a random initial phase. Although the CW signal has been studied before, it will be presented here as a quick reference. One thousand runs are carried out and the results are shown in Figure 13.4. Figure 13.4(a) shows the result of the coarse frequency error. Since the frequency bin width is 20 MHz, the maximum frequency error is close to 10 MHz. The mean error is close to 114 kHz and the standard deviation is about 5.7 MHz.

Figure 13.4(b) shows the results obtained from amplitude comparison, which compared with the coarse frequency measurement, are improved by two orders of magnitudes. The mean and standard deviation of the frequency errors are close to 800 Hz and 12 kHz, respectively. Figure 13.4(c) shows the results of the phase measurement. The results are even better than the amplitude comparison method. The mean and standard deviation of the frequency errors are close to 1 and 33 Hz, respectively.

Figure 13.5 shows the results of a chirp signal without noise over 1000 runs. The maximum chirp rate is set at 2000 MHz/ μ s. The delta frequency, the frequency change over two frames (0.1 μ s), is randomly selected between 0 and 200 MHz with a random initial phase. Figure 13.5(a), showing the frequency estimation error of the coarse frequency estimation method, is comparable to Figure 13.4(a), but slightly better. This phenomenon will be explained in later paragraphs. Figure 13.5(b) shows that the frequency estimation for a chirp signal with amplitude comparison is about two orders of magnitude worse than the CW signal frequency estimation results shown in Figure 13.4(b). For phase comparison, Figure 13.5(c) shows that its frequency estimation is about three orders of magnitude worse than the CW signal frequency estimation shown in Figure 13.4(c). These results are expected because the amplitude and phase comparison methods are derived based on CW signals.

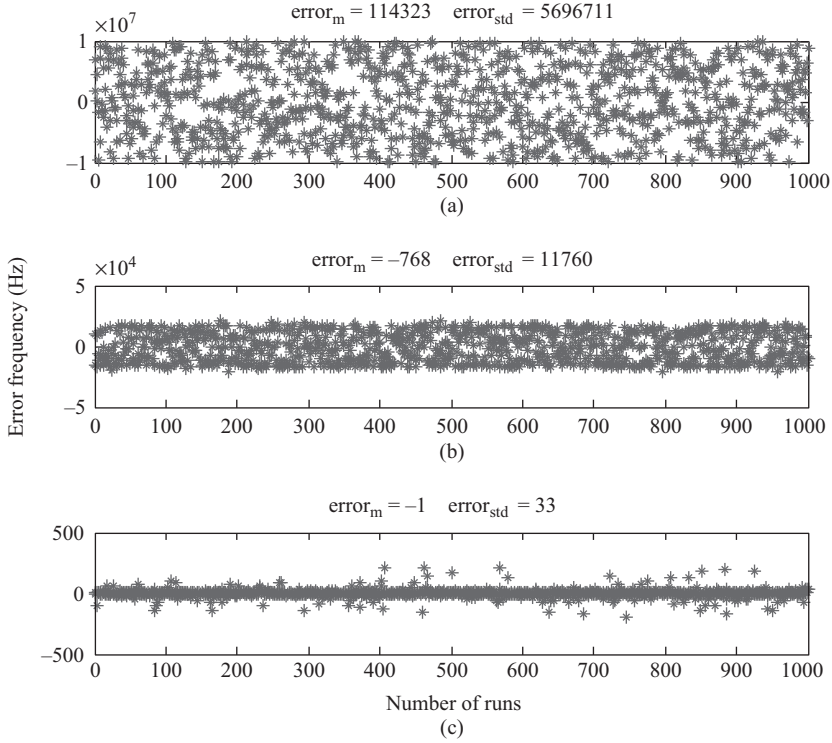


Figure 13.4 Frequency estimation of a CW signal with random input frequency without noise: (a) coarse frequency, (b) amplitude comparison, and (c) phase comparison.

The important message is that by comparing Figures 13.5(b) and 13.5(c) to 13.5(a), we can conclude that the amplitude and phase methods generate better frequency estimates. For the coarse frequency, the error standard deviation is close to 4.1 MHz; for the amplitude and phase comparison methods, these values are about 633 kHz and 63 kHz, respectively. These simple illustrations demonstrate that amplitude and phase comparisons improve the frequency estimation for chirp signal significantly. It is a pleasant surprise because both these methods were originally developed for CW signals.

Before further discussion on the amplitude and phase comparison method, let us explain the improvement of coarse frequency estimation on chirp signals over CW signals (comparing Figures 13.4(a) and 13.5(a)). Figure 13.6 shows the histogram of frequency estimation errors displayed in Figures 13.4(a) and 13.5(a). For a CW signal, since the input frequency is randomly selected, the error should be uniformly distributed, as shown in Figure 13.6(a). On the other hand, for a chirp signal, as shown in Figure 13.6(b), frequency error distribution is somewhat triangle shaped, which has a smaller standard deviation. This distribution can be explained by the following equation:

$$\begin{aligned}\Delta f_{cw} &= f_1 - f_b \\ \Delta f_{ch} &= \frac{f_1 + f_2}{2} - f_b,\end{aligned}\tag{13.4}$$

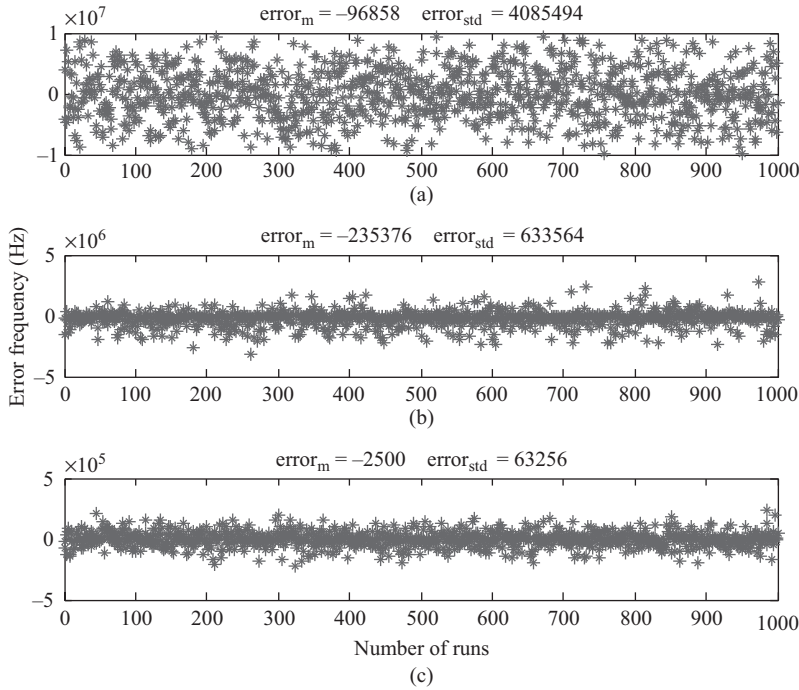


Figure 13.5 Frequency estimations of chirp signals of random starting and ending frequencies without noise: (a) coarse frequency, (b) amplitude comparison, and (c) phase comparison.

where Δf_{cw} and Δf_{ch} are the frequency estimation error for CW and chirp signals, respectively, f_1 is the frequency of the CW signal in the first equation, f_1 and f_2 are the initial and ending frequencies for chirp signal in the second equation, and f_b is the estimated signal frequency at the boundary of two FFT frames. Both f_1 and f_2 are randomly selected, thus their distribution is uniform over 141 to 1400 MHz. The probability density function of the sum of two uniformly distributed functions is triangle shaped, which matches the result in Figure 13.6(b).

Although it appears that the phase comparison method delivers the best performance based on results presented in Figure 13.5, it has hidden fundamental issues. When the number of runs is increased to 10,000 times, sometimes large frequency errors caused by the phase comparison method might happen even when there is no noise added to chirp signals. The cause of the problem is the ambiguity phase error. When the phase is off by 2π , the phase comparison method can generate significant frequency estimation error. In this case the frequency will be off by approximately 20 MHz, the resolution of the frequency bin. Figure 13.7 shows the simulation results of 10,000 runs. In this figure there are four large errors. It is amazing that large errors rarely occur. In some cases even with 10,000 runs, only one or zero large errors occur. The large error occurrence depends on the starting and the delta frequency. From limited observations, it appears that when the two highest frequency bins in one frame have similar amplitudes, large frequency error will occur.

The results presented in Figure 13.7 are simulated in a noise-free setting. To compare these frequency estimation approaches in a more realistic setting, noise is added in the chirp

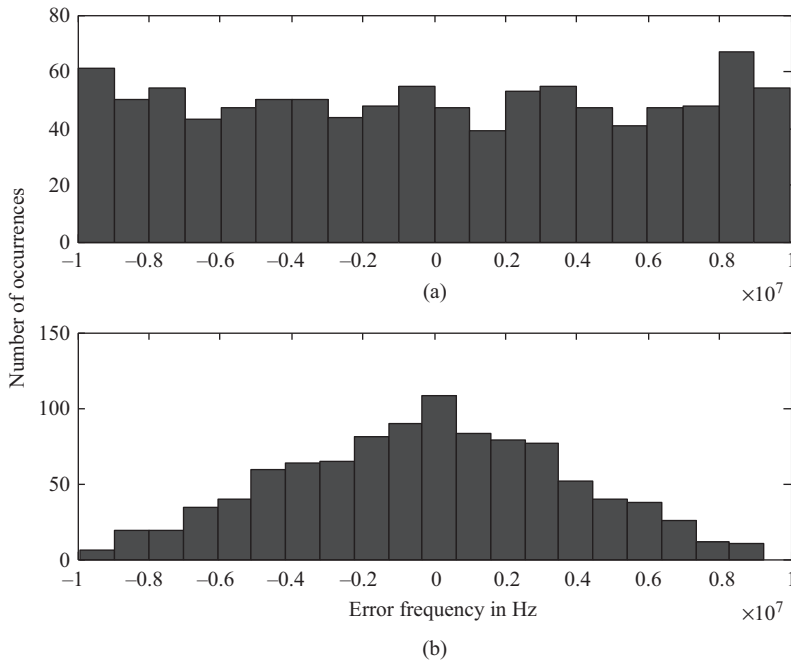


Figure 13.6 Histogram of the frequency error using the coarse frequency estimation method: (a) CW signal and (b) chirp signal.

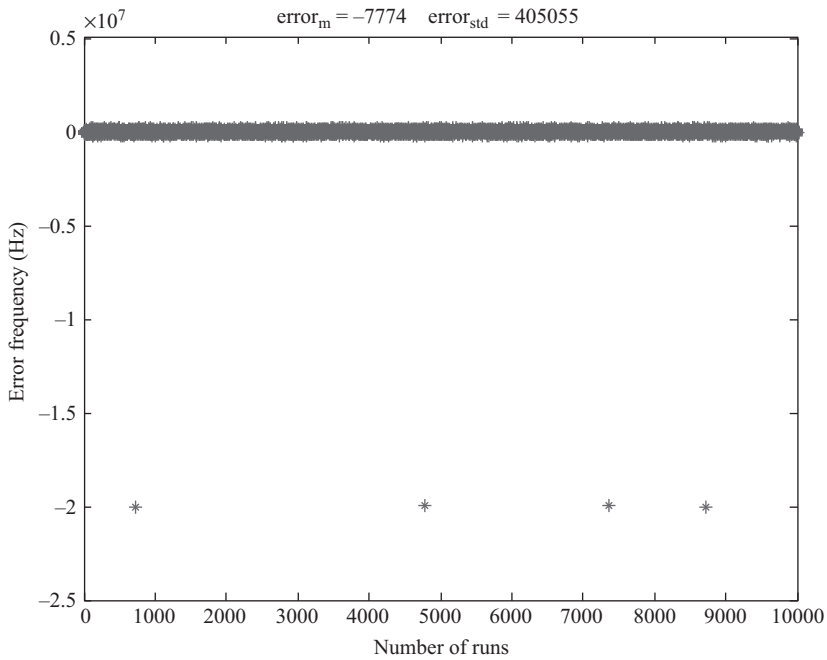


Figure 13.7 Large frequency error produced by the phase comparison method on 10,000 runs of a chirp signal without noise.

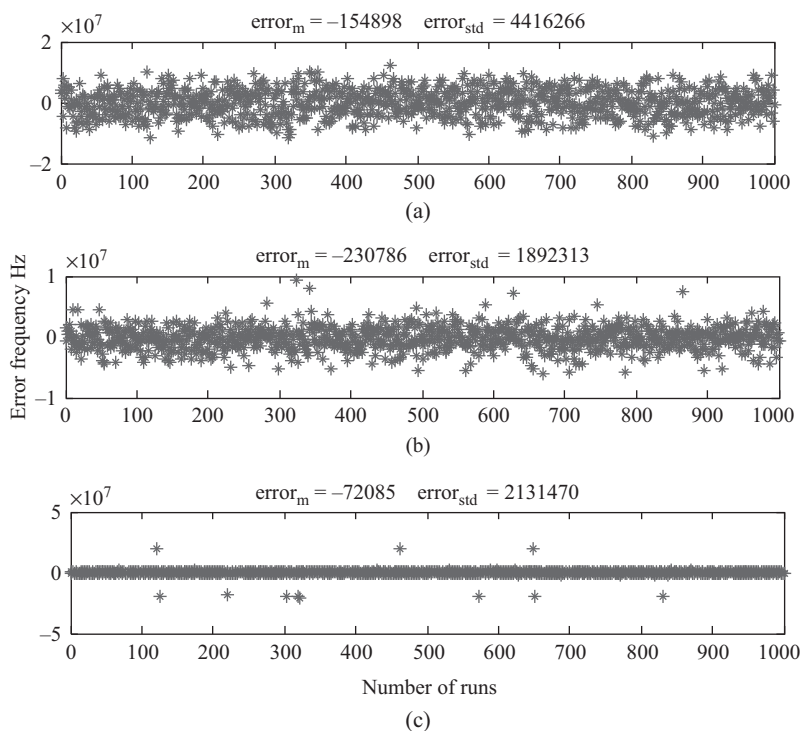


Figure 13.8 Frequency measured with a chirp signal of random starting and ending frequencies at $S/N = 0$ dB for 1000 runs: (a) coarse frequency, (b) amplitude comparison, and (c) phase comparison.

signal with the S/N ratio arbitrarily chosen as 0 dB and the simulation is repeated. The results are shown in Figure 13.8. Under this condition, the amplitude comparison has a slight improvement over the coarse frequency estimation. The phase comparison method has several large errors. If these large frequency errors are ignored, the results are better than the amplitude comparison system.

From these simple tests it appears that although the phase comparison method can improve the frequency readings, there are ambiguity problems under certain conditions. When the signal is weak, the phase ambiguity occurrence is more frequent. The amplitude comparison method improves the frequency measurement slightly under noisy input conditions, but the approach is rather robust and no large frequency errors are generated. The occasional large frequency error of phase estimation methods will be addressed in the next section.

13.5 Phase Comparison Method Aided by the Amplitude Comparison Method

In the previous section, the two center frequencies of two FFT frames measured with the amplitude comparison method are combined into one in order to compare with the frequency

measured with the phase comparison method. In this section, the estimated frequencies at the centers of frames 1 and 2 generated with amplitude comparison are used to alleviate the ambiguity problem accompanying the phase comparison method. The basic concept is that the frequency estimated by the phase comparison method (the frequency between two frames) should be between the two frequencies estimated through the amplitude comparison method (frequencies at the two frame centers). Therefore the frequencies estimated with amplitude comparison can be used to check the correctness of frequencies obtained through the phase comparison method.

This approach uses (13.3) to obtain the frequency through the phase comparison method. Two more frequencies will be generated from this frequency. They are the measured frequency plus and minus 20 MHz, which is the frequency resolution of the FFT operation, because if there is a phase ambiguity, the frequency difference will be equal to the frequency resolution generated by the FFT operation. These three frequencies are compared with the average frequency of two frames measured with the amplitude comparison method and the closest one is selected as the desired result. This method will be referred to as the amplitude-aided phase comparison method.

During the simulation test, the chirp signal's starting and ending frequencies are accidentally chosen to be 200 and 400 MHz and the signal's PW is $0.5\ \mu\text{s}$ (10 FFT frames) with a very high S/N ratio of 100 dB. The resulting signal frequencies estimated with phase comparison and amplitude-aided phase comparison over the signal duration are shown in Figure 13.9.

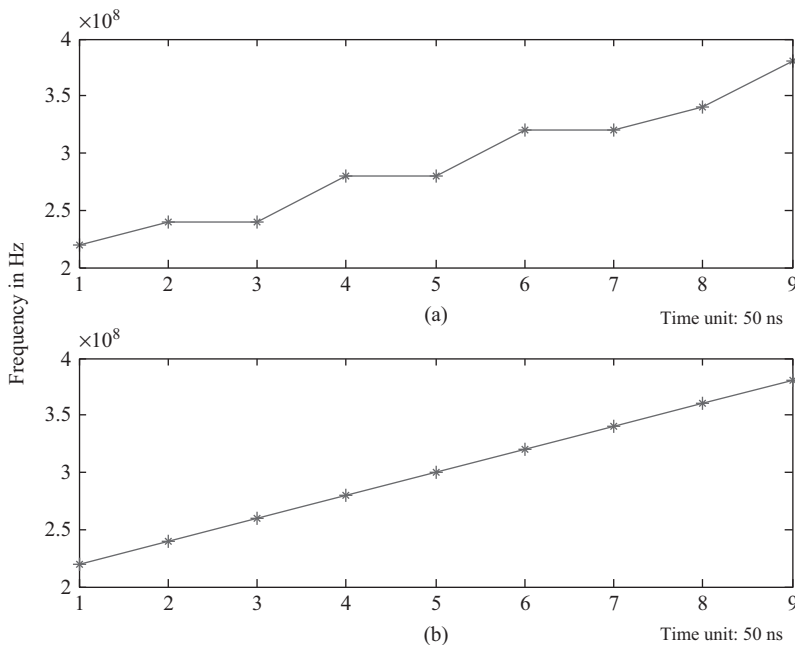


Figure 13.9 Input signal with starting and ending frequencies of 200 to 400 MHz with $S/N = 100$ dB: (a) signal frequency estimated with the phase comparison versus time and (b) signal frequency estimated with amplitude-aided phase comparison versus time.

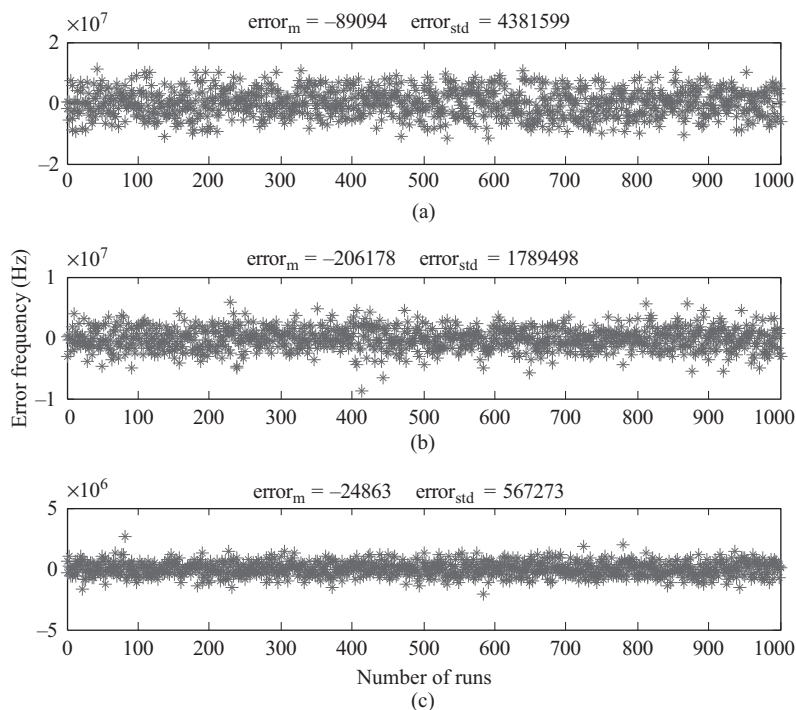


Figure 13.10 Frequency measured with a chirp signal of random starting and ending frequencies at $S/N = 0$ dB: (a) coarse frequency, (b) amplitude comparison, and (c) amplitude-aided phase comparison.

The phase comparison method generates an error three times, as shown in Figure 13.9(a). When the result of the amplitude comparison method is used to correct the phase information, these errors are corrected and the estimated output frequency is a straight line reflecting the linear chirp signal's characteristic, as expected.

The results of 1000 runs are shown in Figure 13.10. The input conditions are identical to the results in Figure 13.8 with the same S/N ratio of 0 dB. The only difference is between Figures 13.8(c) and 13.10(c). In Figure 13.10(c) the amplitude comparison method is used to check the frequency measured by the phase comparison method, and as the result, there are no large frequency errors, as in Figure 13.10(c). Comparing the standard deviation of the error between Figures 13.10(b) and 13.10(c), the improvement is about 3 times.

The phase comparison method is further studied to see whether the amplitude-aided comparison method causes any adverse effect at lower S/N ratios. The delta frequency is randomly selected between 0 and 200 MHz, with a PW of $0.1 \mu\text{s}$ and $S/N = -2$ dB. The simulations are run 1000 times and the frequency errors are shown in Figure 13.11. Figure 13.11(a) shows the results generated by the phase comparison method and Figure 13.11(b) shows the results of the amplitude-aided phase comparison method. The advantage of the aided approach is clearly shown, as many large errors are eliminated and the error's standard deviation is also lower. Figure 13.12 shows similar simulations with $S/N = -5$ dB. Figure 13.12(a) shows the phase comparison results and Figure 13.12(b) shows

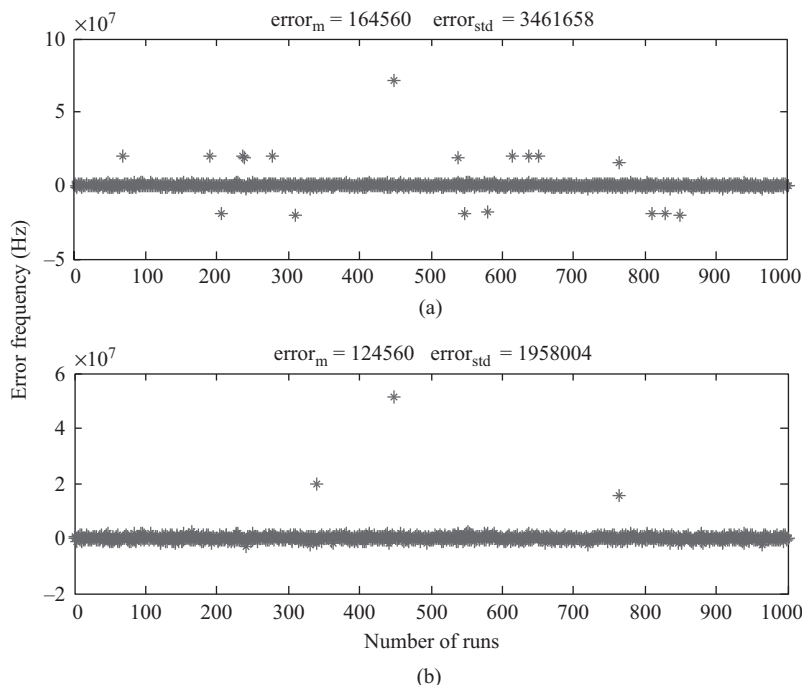


Figure 13.11 Error frequency at $S/N = -2$ dB with the delta frequency randomly selected between 0 and 200 MHz and with a PW of $0.1 \mu\text{s}$ in 1000 runs: (a) phase comparison method and (b) amplitude-aided phase measurement.

the amplitude-aided phase comparison approach. The two results are very close. This simple simulation illustrates that amplitude-aided phase comparison improves the performance and does not produce a negative effect at low signal strength. However, in Figure 13.12, it appears that when the signal is weak, such as at $S/N = -5$ dB, the amplitude-aided method does not improve the results of the phase comparison method, as the amplitude comparison method also has large errors. Another phenomenon worthy of note is that the error frequencies occur at discrete values (Figure 13.11). This means that the error is a multiple of 2π . The error frequencies in Figure 13.12 appear random and are mainly caused by noise.

13.6 Chirp Signal Detection with Amplitude and Phase Comparison

In this section, the amplitude and phase comparison methods are used to detect a chirp signal. The input signal is chosen as a $1\text{-}\mu\text{s}$ PW with a maximum random delta frequency of 1000 MHz and random initial phase. This value is smaller than the $2000 \text{ MHz}/\mu\text{s}$ discussed in Section 13.2. The $1\text{-}\mu\text{s}$ PW is selected because the purpose here is to study the long data effect and $1 \mu\text{s}$ equals the duration of 20 frames. In our study we set the bandwidth of the receiver as 1000 MHz, which limits the PW as well as the chirp rate. If the starting and

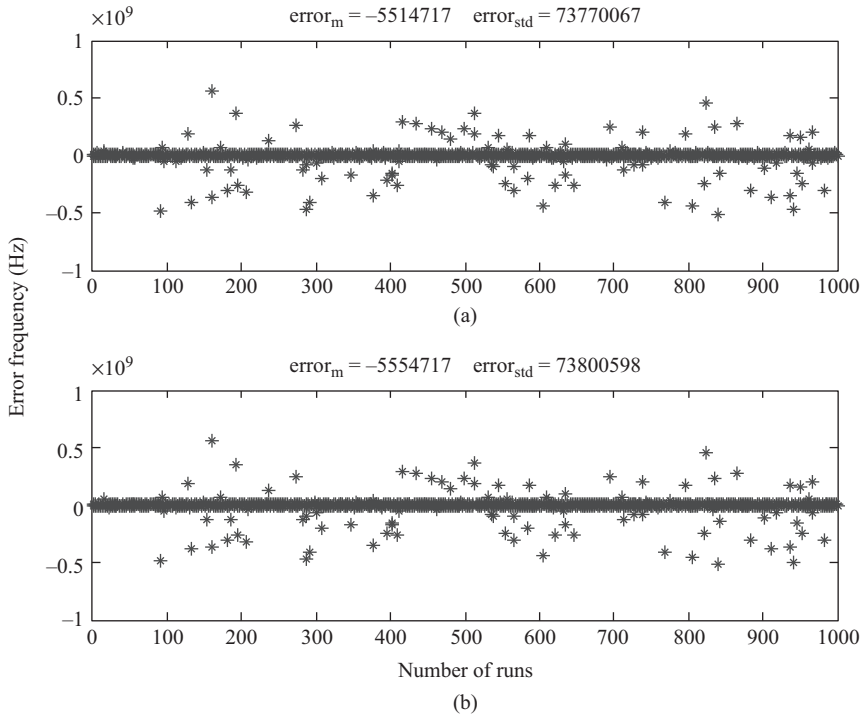


Figure 13.12 Error frequency at $S/N = -5$ dB with delta frequency randomly selected between 0 and 200 MHz and with PW of 0.1 μ s in 1000 runs: (a) phase comparison method and (b) amplitude-aided phase measurement.

ending frequencies are f_1 and f_2 , respectively, the frequency difference between the beginning and the ending of one frame df_f is

$$df_f = \frac{f_2 - f_1}{20}, \quad (13.5)$$

where 20 is the total number of frames. Figure 13.13 shows these frequencies associated with frames 1, 2, and k . In general, if there are M frames, the three frequencies shown in the k section of Figure 13.13 can be written as

$$\begin{aligned} f_{1k} &= \frac{(M+1-k)f_1 + (k-1)f_2}{M} \\ f_{2k} &= \frac{(M-k)f_1 + kf_2}{M} \\ f_{3k} &= \frac{(2M+1-2k)f_1 + (2k-1)f_2}{2M}, \end{aligned} \quad (13.6)$$

where f_{1k} , f_{2k} , and f_{3k} are the beginning, ending, and middle frequencies of the k th frame, respectively. In this special case, $M = 20$ and the frequencies in the first and last frames are labeled accordingly.

Since the amplitude comparison method can produce a frequency representing the center frequency of a frame and the phase comparison method can producing a frequency

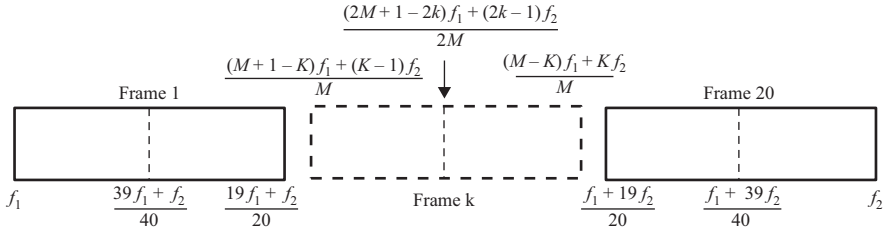


Figure 13.13 Frequencies of a chirp signal in a certain frame.

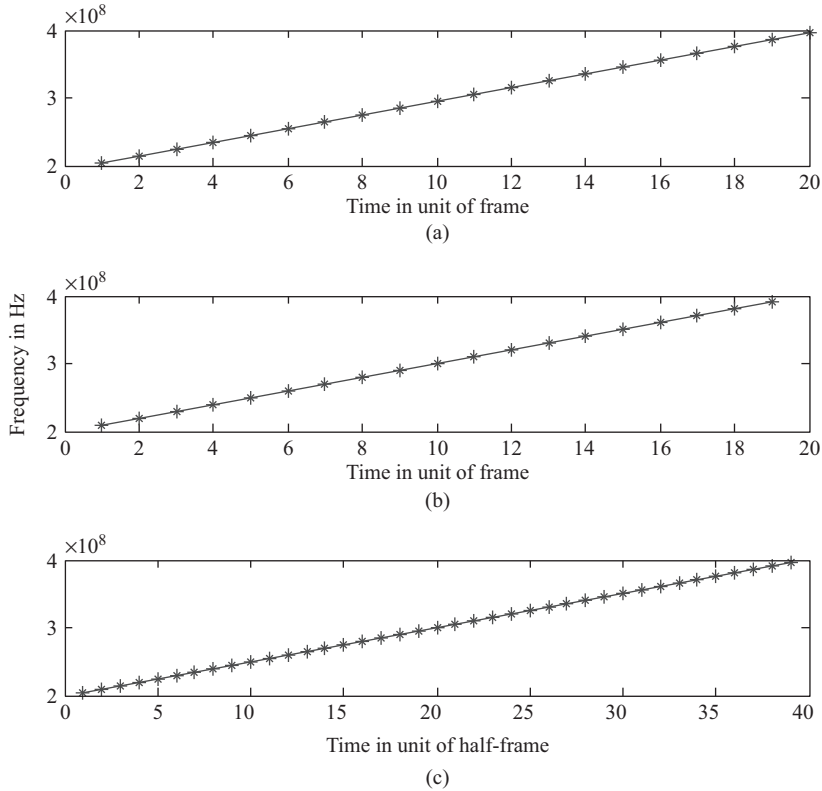


Figure 13.14 Frequency of an input signal with starting and ending frequencies at 200 and 400 MHz without noise: (a) amplitude comparison method, (b) amplitude-aided phase comparison method, and (c) combined amplitude and amplitude-aided phase comparison methods.

representing either a beginning or an ending of a frame, they can be put together in an orderly form to show the actual frequency of the chirp signal.

Figure 13.14 shows the results of the amplitude comparison, amplitude-aided phase comparison, and a combination of these two methods. The starting frequency f_1 and ending frequency f_2 are arbitrarily chosen as 200 and 400 MHz and there is no noise in the simulation. Figure 13.14(a) shows the frequency generated from the amplitude comparison method and there are 20 data points. Figure 13.14(b) shows the results from the amplitude-aided phase

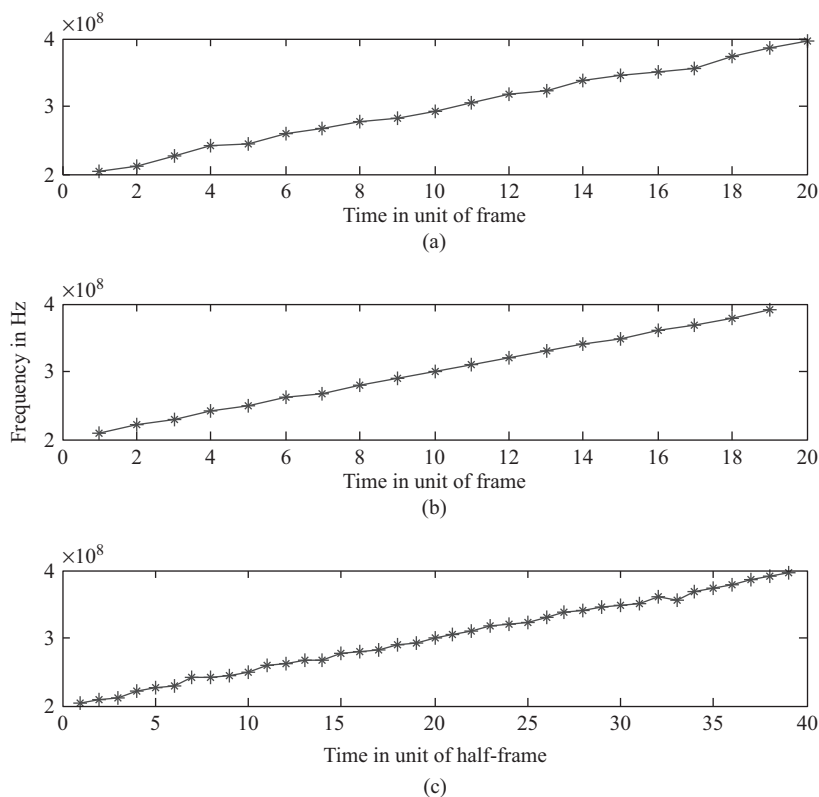


Figure 13.15 Frequency of an input signal with starting and ending frequencies at 200 and 400 MHz and $S/N = -5$ dB: (a) amplitude comparison method, (b) amplitude-aided phase comparison method, and (c) combined amplitude and amplitude-aided phase comparison methods.

comparison method and there are 19 data points. Figure 13.14(c) shows the combined results from the amplitude comparison and amplitude-aided phase comparison methods and there are 39 data points. Since there is no noise in the simulation, all three approaches provide accurate results.

The results when the signal is at $S/N = -5$ dB are shown in Figure 13.15. From observation of the three plots in Figure 13.15, it appears that the amplitude-aided phase comparison method provides the best results, which is closest to a straight line. It should also be noted that whenever there is a large error in the amplitude comparison method it will cause a large error in the amplitude-aided phase comparison method. This is expected because the correction is based on amplitude comparison.

13.7 Find the Chirp Rate of the Measured Signal

In order to detect the chirp signal using frequency estimation results one needs to fit all the estimated frequency points on a straight line. Since both the initial frequency and the

slope of the line are unknown, it is tedious to find a matching line. Since we only consider linear chirp signals, the initial frequency requirement can be eliminated if the difference frequency between frames is plotted, which should be a horizontal line with a constant value. The difference frequencies in the previous simulation are shown in Figure 13.16. These values should be the chirp rate per frame. For the input changes from 200 to 400 MHz, with 20 frames the average chirp rate is 10 MHz per frame (200 MHz/20 frames). It is expected that the results should be close to a horizontal line at 10 MHz per frame. Since the frequency estimation is based on approximation methods, some variation is expected.

Figure 13.16(a) shows the results from the amplitude comparison method; the average chirp rate per frame is 9,998,347 Hz, which is off from the expected value of 10 MHz by about 17 KHz. Figure 13.16(b) shows the results of the amplitude-aided phase comparison method; the average value is equal to the expected chirp rate per frame. In some other simulations the differences are about 1 or 2 Hz, which is better than the amplitude comparison method. Figure 13.16(c) shows a combination of the amplitude and phase comparison methods; the average chirp rate per half-frame is 4,999,174 Hz and the expected value should be 5 MHz (when combining results from two methods, we have two frequency estimations per frame). The error is about 800 Hz. From the average chirp rate per frame, it appears that the amplitude-aided phase comparison method provides the best result. However, when the signal is weak, such as when $S/N = -6$ dB, both the amplitude and the amplitude-aided phase methods produce relatively large errors; the results are shown in Figure 13.17.

13.8 Four Approaches to Detect a Long Weak Signal [8, 9]

So far we discussed different FFT-based approaches to estimate chirp frequency and chirp rate. In this section we discuss how to detect the existence of a chirp signal, which is usually a relatively long but weak signal. Four approaches will be discussed. In this study the signal duration is assumed to be 1 μ s and the maximum delta frequency is 1000 MHz, to accommodate the 1000 MHz bandwidth of the receiver. The signal lasts 20 frames.

The first approach is similar to the detection of a CW signal through noncoherent integration. With a 2.56-GHz sampling rate, 128-point FFT operation and 1000 MHz bandwidth there will be 53 frequency bins under consideration per frame, as listed in Table 6.1 of Tsui [10]. Since there are 20 frames, all 20 amplitudes of the 53 frequency bins are added. The highest of the 53 outputs are compared with a predetermined threshold for detection. For a chirp signal, a similar approach is applied. The difference between chirp signal detection and CW signal detection is shown in Figure 13.18.

For CW signal detection, in all 20 frames the amplitudes of the frequency bins with the same bin number are added together. In Figure 13.18(a), the straight line indicates all the bins with the same number added together. For a chirp signal, the chirp rate must be found first, and the slope can be used to align the bins added together. In Figure 13.18(b), the tilted line indicates the number of bins added together. This method will be referred to as the summation method. If the slope estimation is wrong, the summation should be less than the threshold. Thus this method also depends on the accuracy of the frequency estimation.

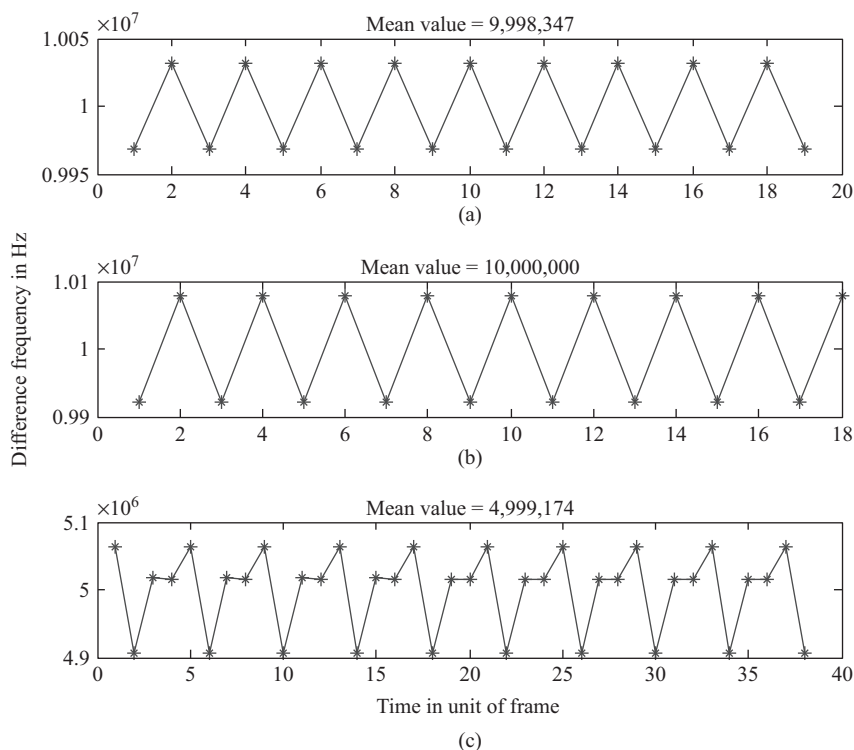


Figure 13.16 Difference frequency of an input signal with starting and ending frequencies at 200 and 400 MHz without noise: (a) amplitude comparison method, (b) amplitude-aided phase comparison method, and (c) combined amplitude and amplitude-aided phase comparison methods.

Although this approach may improve the sensitivity on a long signal, the result is expected to be inferior to a CW signal. The reason is that the amplitude of a CW signal in the frequency domain is higher than a chirp signal where the energy is spread.

The second method, suggested by David Lin of the Air Force Research Laboratory (AFRL) [11], is quite similar to the summation method just described. The only difference is that instead of summing all the frequency bins, we only sum the frequency bin with maximum amplitude in each of 20 frames. This approach reduces the number of summations from 53 bins to 1 bin. Once the summation is above a certain threshold, a signal is detected. The frequency will be found from the peak bins summed. This method will be referred to as the peak summation method.

The third approach, suggested by Lee Liou of AFRL [12], is to find the number of difference frequencies within a certain difference frequency range. In Figure 13.17, most of the difference frequencies are clustered near a certain value, the slope of the chirp signal. For a 1- μ s signal there are 20 frames and 19 difference frequencies, as shown in Figure 13.17(a) and 13.17(b). If a threshold is set around the difference frequency, a detection criterion can be obtained. Let us refer to the number of difference frequencies, 19, as N . If L of the N difference frequencies are within the boundary of the threshold, it is considered that a signal is

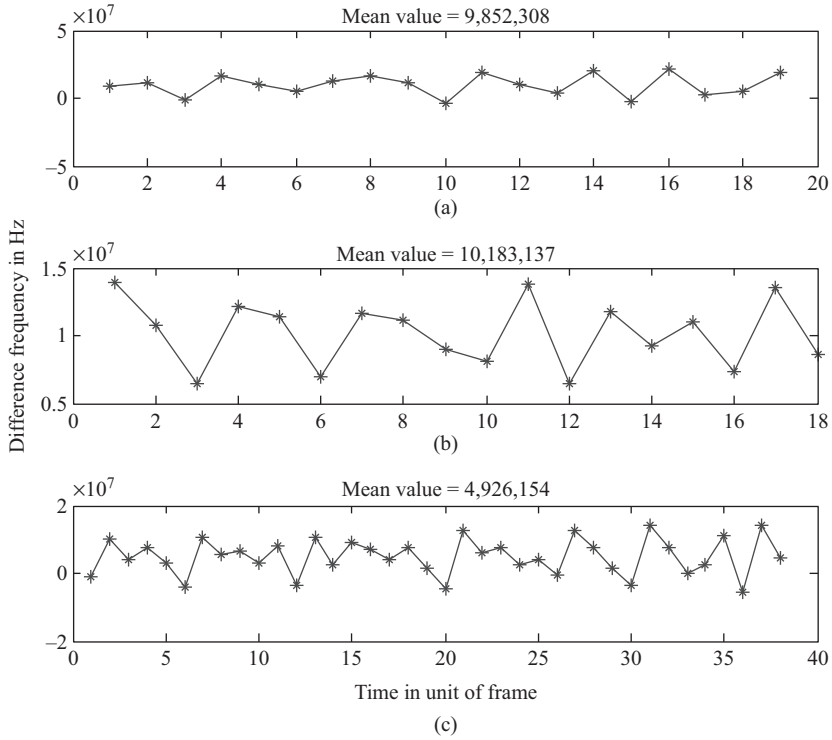


Figure 13.17 Difference frequency of an input signal with starting and ending frequencies at 200 and 400 MHz with $S/N = -6$ dB: (a) amplitude comparison method, (b) amplitude-aided phase comparison method, (c) combined amplitude and amplitude-aided phase comparison methods.

detected. Let us loosely define this method as L -out-of- N detection. This same name was used in Section 7.13; however, these two L -out-of- N methods are similar but not identical.

The last method is to use the eigenvalue method discussed in Chapter 8 of Tsui [10]. The eigenvalues are calculated from 20 frames and the results are summed together. In order to keep the calculation simple, a 3×3 matrix will be considered. If there is only one CW signal, it will affect only two eigenvalues. If the input is a chirp signal, it will affect all three eigenvalues. In this approach the third eigenvalue (largest one) is used for detection of whether a signal exists and the first eigenvalue (the smallest one) is used for detection of an FM signal.

The following three sections are dedicated to the determination of a threshold for the four methods discussed in this section.

13.9 Threshold Setting for Both the Summation and Maximum Summation Methods

For the detection approaches mentioned in the previous section, a threshold must be determined first. The probability of false alarm of 10^{-7} is used to determine the threshold.

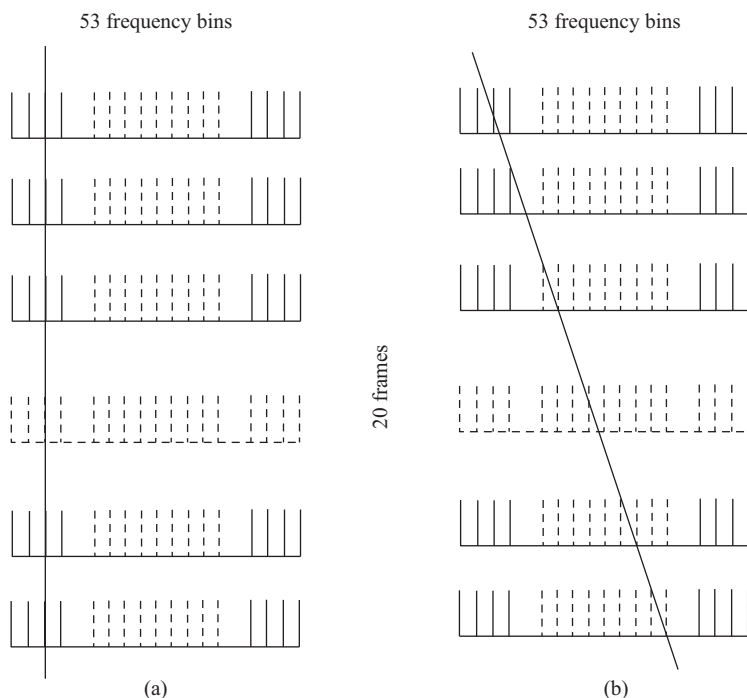


Figure 13.18 Summation of frequency bin amplitudes: (a) for a CW signal and (b) for a chirp signal.

For the summation method, the threshold is identical to the noncoherent method discussed in Sections 6.6 through 6.8 in Tsui [10]. The threshold can be obtained from numerical calculation of a convolution function. This threshold is used because when the delta frequency of the chirp signal is small, the signal is close to a CW signal, so the threshold determination and signal detection methods developed for the CW signal can still be applied in this scenario. There are 20 frames and the threshold is obtained from the summation of 20 frequency bins.

Instead of obtaining the threshold from 20 convolutions of the Rayleigh function, it is obtained through an approximation. A noise of 2560 data points is generated as the input, which equals 20 of 128 point frames. A Blackman window of 128 points is applied to each frame of data and then an FFT is performed on the windowed data. In the frequency domain, 53 frequency bins (6 to 58) are selected for this study. The amplitudes of all 20 frames are summed together. This process runs 10,000 times to obtain $53 \times 10,000$ data points. A threshold can be estimated from the histogram of these 530,000 data points assuming the amplitude's probability density function is a Gaussian distribution. However, this threshold is slightly lower than a threshold obtained from 20 convolutions of a Rayleigh function. A fine adjustment can be applied based on the results in Table 6.4 of Tsui [10]. It is estimated that when 20 summations are performed, the difference between the convolution method and the Gaussian distribution approximation is about 4%. The desired threshold will be the

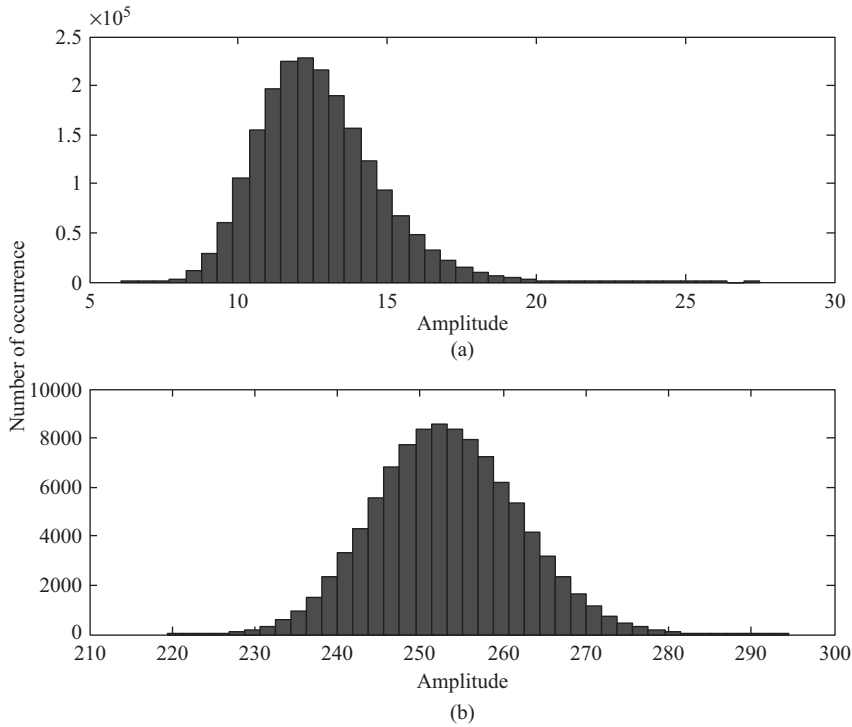


Figure 13.19 Histogram of peak noise outputs: (a) peak frequency bins and (b) summation of 20 peak frequency bins.

threshold obtained from the Gaussian distribution approximation divided by 0.96; the final value is about 185.

For the peak summation method, the threshold is set by an approximation method because either the distribution of the maximum values or the distribution of the summed result is unknown. A Monte Carlo approach is used to find both distributions. Ten thousand runs are performed and the results are shown in Figure 13.19. Figure 13.19(a) shows the distribution of the amplitude of peak frequency bins, which is different from the Rayleigh distribution. If this distribution is convolved with itself 20 times, it is expected that the result is close to a Gaussian distribution. The result is shown in Figure 13.19(b). The threshold value is obtained from a Gaussian distribution and is about 299.

13.10 Threshold for the L -out-of- N Method

Before the threshold of the L -out-of- N method can be set, the frequency report from both the amplitude and phase comparison methods with only noise as input must be found. It is anticipated that the frequency obtained from the amplitude comparison method will be uniformly distributed. The frequency calculated from the phase comparison method either with or without amplitude aid may not be uniform since two frames of information are required.

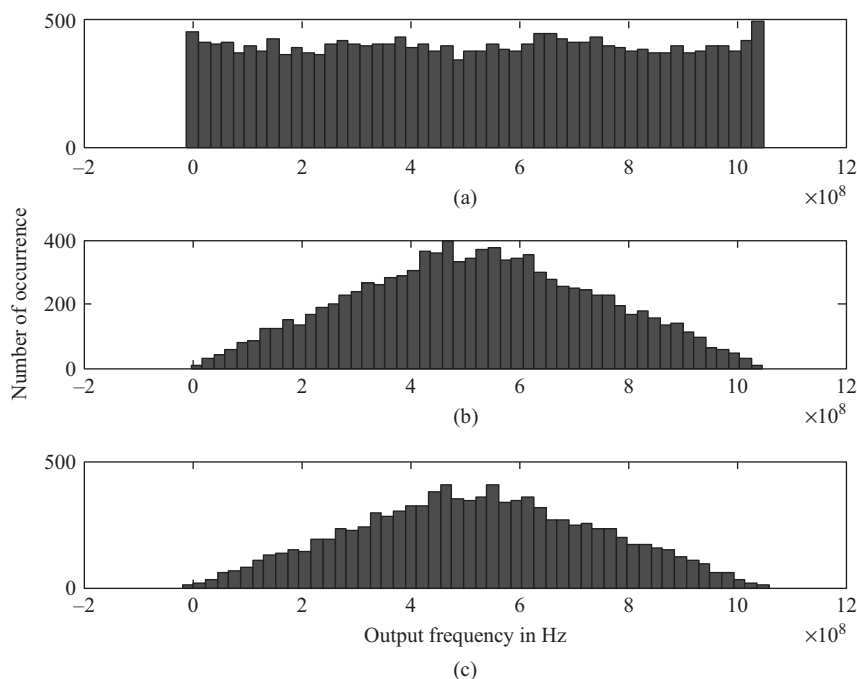


Figure 13.20 Histogram of frequency output with noise as the input: (a) amplitude comparison, (b) phase comparison, and (c) amplitude-aided phase comparison.

The results from 10,000 runs are shown in Figure 13.20. Figure 13.20(a) shows the results from the amplitude comparison method. The results can be considered as uniformly distributed. Figures 13.20(b) and 13.20(c) are the phase comparison method without and with amplitude aid, respectively. These results are obtained through the fourth equation in (13.3), or the single index method. The results can be considered as triangle shaped. This frequency calculated from the phase method is obtained from the phase difference between two consecutive frames. If the distribution of peak frequency in each frame is uniform, as shown in Figure 13.20(a), the distribution of their difference should be a triangle, which is the convolution of two rectangular probability density functions. The frequency range in Figure 13.20 is slightly wider than the input bandwidth of 1000 MHz because, in obtaining the random frequency, 53 frequency bins are selected, which is slightly wider than 1000 MHz.

Figure 13.21 shows the histogram of 100,000 random inputs uniformly distributed from 0 to 1. The frequency difference between two consecutive frames is referred to as the difference frequency and the double difference is defined as the difference of the difference frequency. The histogram of the difference and the double difference frequencies of this signal are shown in Figure 13.21(a) and 13.21(b), respectively. These results can be obtained from the convolution of the input distribution. Figure 13.22(a) and 13.22(b) show the distribution of the difference frequency and double difference frequency obtained through convolution methods, and these results match Figure 13.21 rather well. Figure 13.22(a) is

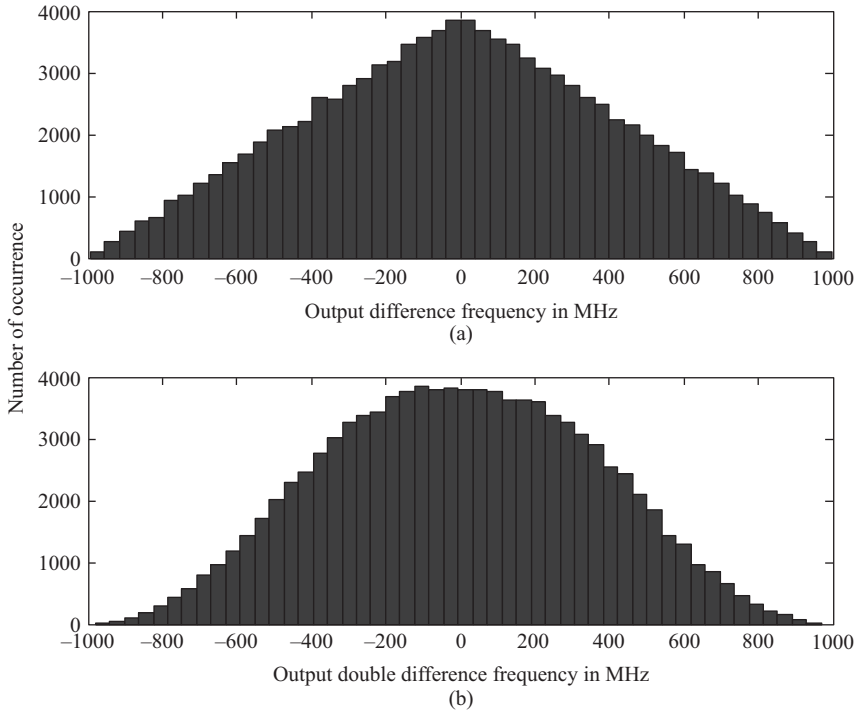


Figure 13.21 Histogram obtained from 100,000 runs of a uniform distribution: (a) difference frequency between two consecutive frames and (b) difference frequency of two consecutive difference frequencies.

obtained from two uniform distributions and Figure 13.22(b) is obtained from the convolution of two triangle-shaped distributions.

It takes two steps to find the desired threshold. The first step is to find the probability of occurrence from the curves in Figure 13.22. Once the probability of occurrence is found, the second step is to find the occurrence of L -out-of- N .

First let us find the frequency calculated from the amplitude comparison method. The distribution of the difference frequency is shown in Figure 13.22(a). Let us use half the figure, whose range of frequency is from 0 to 1000 MHz, to find the probability in the range from frequency = 0 to x , as shown in the figure. The probability can be found from the areas. The total area A from frequency = 0 to 1000 is 500. The area from frequency = 0 to x in the trapezoidal is

$$\begin{aligned}
 A &= 1 \times 1000/2 = 500 \\
 \text{Area} &= \left(1 + \left(1 - \frac{x}{1000}\right)\right) \frac{x}{2} = \left(2 - \frac{x}{1000}\right) \frac{x}{2} \quad \text{thus} \\
 p_a &= \frac{\text{Area}}{A} = \left(2 - \frac{x}{1000}\right) \frac{x}{1000},
 \end{aligned} \tag{13.7}$$

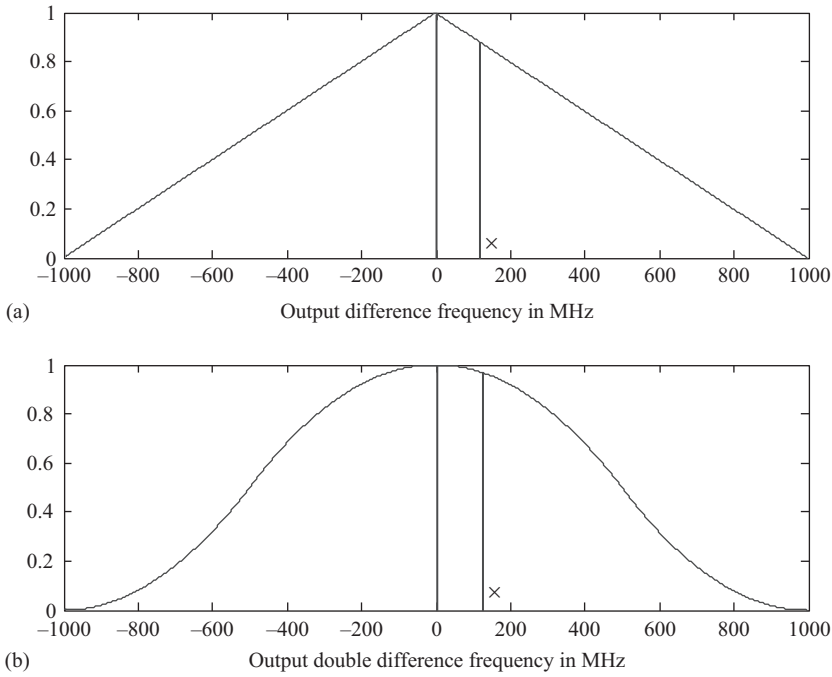


Figure 13.22 Histogram obtained from convolution of a uniform distribution: (a) difference frequency and (b) double difference frequency.

where p_a is the probability of a frequency in the trapezoidal area. Since in this case there are 19 difference frequencies from 20 frames of data, $L = 10$ is arbitrarily chosen because it is close to $N/2$, or $19/2$. When L is close to half of N , a better sensitivity can be obtained. When there are 10 or more frequencies within this threshold, a false detection happens. Thus the threshold can be found from the following equation:

$$p_{fal} = \sum_{i=10}^{19} \frac{19!}{(19-i)!i!} p_a^i (1-p_a)^{19-i} = 10^{-7}, \quad (13.8)$$

where p_{fal} is the probability of false alarm. A threshold can be obtained from this equation through trial and error. The result is shown in Figure 13.23(a). The bandwidth required is about ± 34.24 MHz at a probability of false alarm of 10^{-7} .

For the distribution of Figure 13.22(b), the results are obtained from (13.9):

$$y = 1 - \frac{x^2}{5 \times 10^5} \quad \text{for } x = 0 - 500$$

$$y = \frac{(x - 10^3)^2}{5 \times 10^5} = \frac{x^2 - 10^3 x + 10^6}{5 \times 10^5} \quad \text{for } x = 501 - 1000. \quad (13.9)$$

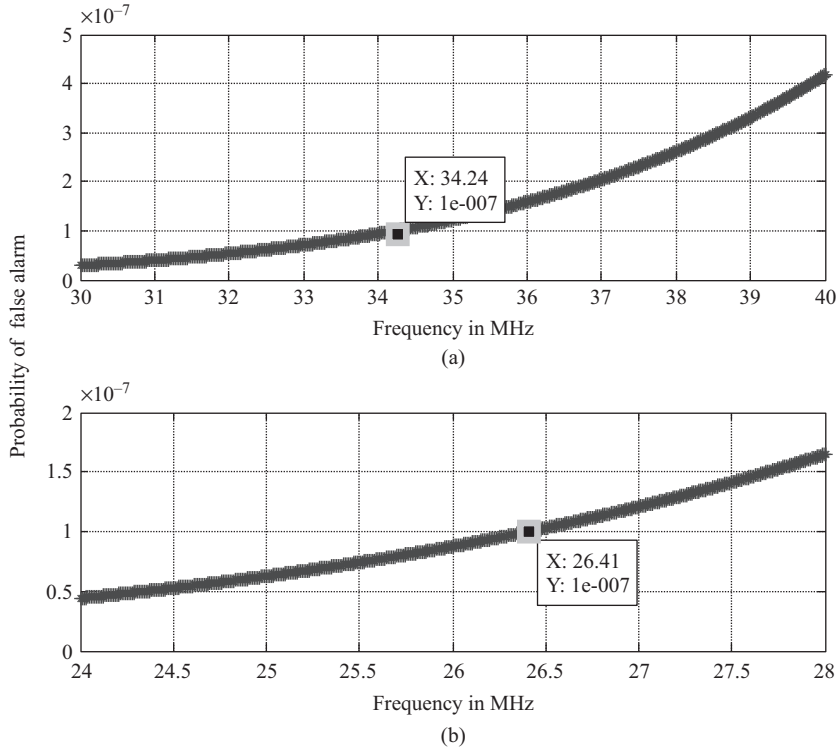


Figure 13.23 Threshold obtained through the difference frequency method: (a) amplitude comparison and (b) phase comparison with a single index.

To find half of the total area A in this figure, the area from double difference frequency = 0 to x , and the probability of double difference frequency reading in $(0, x)$, the following equation can be used:

$$\begin{aligned}
 A &= \int_0^{500} y dx + \int_{500}^{1000} y dx = \left(x - \frac{x^3}{15 \times 10^5} \right) \Big|_0^{500} + \frac{\frac{x^3}{3} - 10^3 x^2 + 10^6 x}{5 \times 10^5} \Big|_{500}^{1000} \\
 &= 500 - \frac{500^3}{15 \times 10^5} + \frac{\frac{10^9}{3} - 10^3 \times 10^6 + 10^9}{5 \times 10^5} - \frac{\frac{500^3}{3} - 10^3 500^2 + 10^6 \times 500}{5 \times 10^5} \\
 &= 500 - 83.33 + 666.67 - 583.33 = 500 \\
 Area &= x - \frac{x^3}{15 \times 10^5} \\
 p_a &= \frac{Area}{A} = \frac{x - \frac{x^3}{15 \times 10^5}}{A}.
 \end{aligned} \tag{13.10}$$

The total area obtained is the same as that obtained in (13.7), which is a reasonable result. The probability of the double difference frequency reading in $(0, x)$ can be found from the

first part of the integration of A since the frequency range should be less than 500 MHz. The probability p_a of a frequency reading in the region $(0, x)$ can be used to find the threshold. An equation similar to (13.8) can be used to find the threshold value. The only difference is that when phase is used to find the output frequency there are only 19 frequencies and 18 difference frequencies. Under this condition, nine difference frequencies within the boundary will be classified as detection. The threshold will be obtained as

$$p_{fal} = \sum_{i=9}^{18} \frac{18!}{(18-i)!i!} p_a^i (1-p_a)^{18-i} = 10^{-7}. \quad (13.11)$$

The threshold values are shown in Figure 13.23. Figure 13.23(a) shows the threshold for the frequency-obtained amplitude comparison method, and the value is ± 34.24 MHz. Figure 13.23(b) shows the threshold for the frequency-obtained with phase comparison method, and the value is ± 26.41 MHz. The big difference is due to the measured frequency distribution, as shown in Figure 13.20.

For the frequency obtained through the averaged index method, the third equation of (13.3), the distribution is also a triangle, as shown in Figures 13.20(b), 13.20(c), 13.21(a), and 13.22(a). However, the difference frequency range is from -500 MHz to 500 MHz rather than from -1000 MHz to 1000 MHz. This phenomenon can be explained through a numerical example. Assume that the indexes of four consecutive frames are 1, 53, 1, and 1. Using the single index approach, the three frequencies measured can be near 0, 1000, and 1000 MHz, thus the difference frequencies can be close to -1000 and 0 MHz. Using the averaged index method, the frequency can be near 500, 1000, and 500 MHz, and the difference frequency will be close to -500 and 500 MHz. Using the same approach to find the threshold, the value should be half of the value obtained in Figure 13.23(b).

13.11 Threshold for the Eigenvalues and Summary of Thresholds

The threshold for the eigenvalue method is obtained from 10,000 runs with noise as the input. Each run contains 20 frames of data and each frame has 128 points. A 3×3 matrix is formed from the 128 data points with lags of 1, 20, and 51 points. The largest eigenvalues are selected from all 20 frames and the results are summed. Thus each run generates one data point. The histogram of the 10,000 runs is plotted and the result is approximated by a Gaussian distribution. The threshold obtained with a 10^{-7} probability of false alarm is about 17.30. The threshold values obtained from Sections 13.9 through 13.11 are listed in Table 13.1.

13.12 Sensitivity Study of All Detection Methods

Once the threshold is obtained, the sensitivity can be obtained through simulation. One thousand inputs with the same S/N ratio but different noise are used as input. Each input signal is $1\text{-}\mu\text{s}$ long, or 20 frames of data. There are 20 frequencies measured by the amplitude comparison method and 19 frequencies by the phase comparison method. The difference frequencies are calculated and there are 19 values from the amplitude comparison method

Table 13.1 Threshold values determined for different methods.

	Detection method	Threshold value	Method used
1	Amplitude summation	185	Gaussian distribution adjusted
2	Maximum amplitude summation	299	Gaussian distribution
3	Frequency by amplitude comparison (L -out-of- N)	± 34.24 MHz	Convolution of rectangular shape
4	Frequency by phase comparison with a single index (L -out-of- N)	± 26.41 MHz	Convolution of triangle shape
5	Frequency by amplitude-aided phase comparison with a single index (L -out-of- N)	± 26.41 MHz	Convolution of triangle shape
6	Frequency by phase comparison with an averaged index (L -out-of- N)	± 13.21 MHz	Half of row 4
7	Frequency by amplitude-aided phase comparison with an averaged index (L -out-of- N)	± 13.21 MHz	Half of row 5
8	Eigenvalue threshold	17.30	Gaussian distribution

and 18 values from the phase measurement methods. Since the signal in this study is limited to linear upward chirp, the difference frequency should be close to a positive constant value, as shown in Figures 13.16 and 13.17. The mean value of these outputs is obtained.

Let us consider the L -out-of- N method first. From the mean value, the thresholds obtained in rows 3 through 7 are used as the detection criterion. For the amplitude comparison method, if there are 10 signals detected in the range, the signal is declared as detected. For the phase comparison method, if there are nine signals in the range, the signal is declared as detected. The S/N value changes from -12 to -2 dB. First let us use a CW signal (or delta frequency = 0) as the input signal. The input is repeated 1000 times with random input frequency from 141 to 1410 MHz.

Figure 13.24 shows the input S/N in decibels versus the number of detections. The sensitivities can be obtained from these curves at 90% probability of detection. Figure 13.24(a) shows the result of the frequency measured by the amplitude comparison method; the sensitivity is at about -6.7 dB. Figure 13.24(b) shows the result of the frequency obtained through phase comparison with a single index; the sensitivity is about -6.5 dB. If amplitude correction is added in the process, the sensitivity is about the same value (result not shown in this figure). Figure 13.24(c) shows the result of the frequency obtained through amplitude-aided phase comparison with an averaged index; the sensitivity is about -5.7 dB. If amplitude correction is not used, the sensitivity is about -5.5 dB (results not shown in this figure).

If the chirp rate is at 1000 MHz/ μ s, the highest value in this study, the sensitivity by the amplitude comparison method is about -6.0 dB. It should be noted that under this condition, the input chirp always starts from 141 MHz and ends at 1410 MHz and only different noise is used in each run. For the phase comparison method with a single index, the sensitivity is about -5.7 dB, and for the amplitude-aided phase comparison method with a single index it is also about -5.7 dB. The sensitivity of the averaged index is about -5.2 dB for both the phase comparison method and the amplitude-aided phase comparison method.

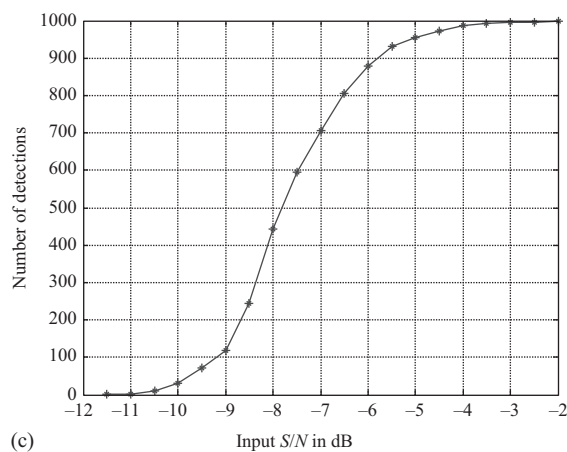
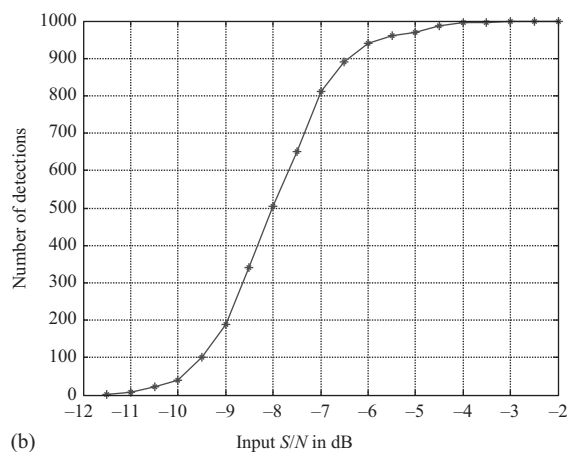
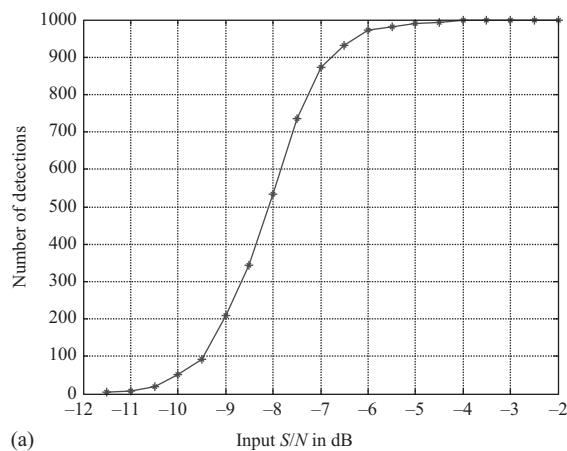


Figure 13.24 Sensitivity measured from the L -out-of- N method: (a) amplitude comparison, (b) phase comparison with a single index, and (c) amplitude-aided phase comparison with an averaged index.

The sensitivity of the receiver using the maximum amplitude summation method provides the best results. Using the threshold of 299 listed in Table 13.1, the sensitivity is about -8.3 dB for a CW signal. When the input signal chirp rate is 1000 MHz/ μ s, the sensitivity is close to -8.1 dB.

For the summation method, the chirp rate must be estimated first. The chirp rate is used to determine how to align the frequency bin from frame to frame. The chirp rate is measured through the amplitude comparison method. The threshold used in this operation is 185, as listed in Table 13.1. The sensitivity plot produced by this process is not smooth. The number of runs is increased to 5000 to smooth the plot and the sensitivity is about -6.8 dB. When the input chirp rate is 1000 MHz/ μ s, the sensitivity is at -5.5 dB. Since this method produces a large differential sensitivity between the CW and 1000 MHz/ μ s inputs, the method is tested against a known input frequency. In other words, the chirp rate is known rather than obtained from calculating the frequency of each frame. This is the well-known method for detecting a CW signal because the input chirp rate of zero is known. For a CW signal, the sensitivity is about -11.7 dB and for a 1000 MHz/ μ s chirp rate it is about -11.6 dB. The sensitivity obtained from a 128-point FFT with Blackman window is about -1.5 dB [10; Table 6.6]. For a 20-point noncoherent integration, the gain is about 10.2 dB [13] and the corresponding sensitivity is about -11.7 dB (-1.5 – -10.2). From this simple argument, it confirms that the method used is legitimate.

For the eigenvalue method, the sensitivity is about -7.5 dB for a CW signal and -7.2 dB for a chirp signal with a 1000 MHz/ μ s chirp rate. The sensitivity is obtained from reading the detection curves (probability of detection versus input S/N). Through the process, most of the data are obtained from more than one run. It appears that the results are rather consistent and the readings are usually within ± 0.1 dB. From this very limited experience it is estimated that the error should be less than ± 0.2 dB.

13.13 Summary of Sensitivity

In this section, the results of the measured sensitivity are discussed. The results measured from different methods with two chirp rates of 0 and 1000 MHz/ μ s are listed in Table 13.2.

The detection procedure is briefly summarized here according to the numerical value in the first column.

1. The frequency is measured from amplitude comparison. The difference frequency is used for detection. There are 19 difference frequencies and if 10 or more of the difference frequencies are within ± 34.24 MHz, a signal is detected.
2. The frequency is measured from phase comparison between two consecutive frames. The index of the maximum output from the first frame is used to calculate the frequency. There are 18 difference frequencies and if 9 or more of the difference frequencies are within the ± 26.41 MHz, a signal is detected.
3. This method is similar to method 2. The only difference is that amplitude comparison is used to aid the frequency measured by phase.
4. The frequency is measured from phase comparison between two consecutive frames. However, the average index of the maximum frequency output from both the first and the second frames is used to calculate the frequency. The operation changes the output noise distribution and the threshold decreases to ± 13.21 MHz. There are 18 difference frequencies and if 9 or more of the difference frequencies are within the threshold limit, a signal is detected.

Table 13.2 Sensitivity obtained from different detection methods.

Method	Sensitivity S/N (dB)	
	CW	1 GHz/ μ s
1. Frequency by amplitude comparison ± 34.24 MHz (L -out-of- N)	-6.7	-6.0
2. Frequency by phase comparison with a single index ± 26.41 MHz (L -out-of- N)	-6.5	-5.7
3. Frequency by amplitude-aided phase comparison with a single index ± 26.41 MHz (L -out-of- N)	-6.5	-5.7
4. Frequency by phase comparison with an averaged index ± 13.21 MHz (L -out-of- N)	-5.5	-5.2
5. Frequency by amplitude-aided phase comparison with an averaged index ± 13.21 MHz (L -out-of- N)	-5.7	-5.2
6. Maximum amplitude summation	-8.3	-8.1
7. Frequency information by amplitude summation of FFT outputs	-6.3	-5.5
8. Summation of FFT outputs with known input frequency	-11.7	-11.6
9. Eigenvalue method	-7.5	-7.2
10. Brute force approach*	-11.7	-11.6

*Discussed in Section 13.14.

5. This method is similar to method 4. The only difference is that amplitude comparison is used to aid the frequency measured by phase.
6. This method takes the maximum output from all 20 frames and adds them together. If the threshold (299) is crossed, a signal is detected.
7. The frequency is measured through the amplitude comparison method. From the frequency, the chirp rate is estimated. All 53 frequency bins from 20 frames are added together to compare with the threshold of 185.
8. Since all the sensitivity obtained from the above methods are relatively low against the CW signal, the method 7 is used to compare with a CW signal. In this approach the chirp rate is a known value. Under this condition the summation is used to determine the sensitivity and the result is very close to the expected value, the sensitivity for the CW signal.
9. Use noise as the input and a 3×3 matrix to generate three eigenvalues. The largest eigenvalues from all 20 frames are added together as a threshold. The sensitivity is found by comparing the summation of the largest eigenvalues to the threshold.

The above results show that the sensitivity is low when the receiver tries to detect a chirp signal with unknown chirp rate. The best result is obtained from the summation of maximum outputs. This result is about 3.4 dB ($11.7 - 8.3$) less sensitive than CW signal detection from methods 6 and 8.

Comparing the results of methods 2 and 3 and 4 and 5, it appears that both the phase comparison methods, whether with or without amplitude correction, produce similar results. This result agrees with the results from Figures 13.12(a) and 13.12(b), that is, at low S/N ,

amplitude correction does not provide better results. Only at higher a S/N value does the amplitude comparison produce better results.

From Figures 13.15(a) and 13.15(b), it appears that the frequency measured through the amplitude-aided phase comparison method is better than from amplitude comparison because the frequency versus time is closer to a straight line. However, from the sensitivity study, method 1 provides slightly better sensitivity than method 2. This phenomenon can be explained as follows. In this detection scheme, the difference frequency is used as the threshold. For the same probability of false alarm, the threshold values are different. The frequency measured with the amplitude comparison method has a looser threshold than that obtained from a phase comparison method. When a signal is detected, the difference is within about ± 34.24 MHz for the amplitude comparison method and ± 26.41 MHz for the phase comparison method. If the average index is used to calculate the frequency, the threshold is even tighter, which reflects on the lower sensitivity. In this type of detection, in addition to the probability of false alarm, one more parameter is introduced, the difference frequency range.

The summation of maximum outputs generate the best result (-8.3 dB) and the eigenvalue method produces the second best result (-7.5 dB). The difference between them is about 0.8 dB. For the highest chirp rate of 1000 MHz/ μ s, the sensitivity degrades about 0.2 to 0.8 dB in comparison with the CW signal. The degradation is expected because at a higher chirp rate the energy can spread into several frequency bins, thus the amplitude will decrease. Interestingly, the summation of maximum outputs is less sensitive to higher chirp rate.

Finally, it is interesting to find the standard deviation of the difference frequency versus input S/N . In this study the amplitude method is used to find the frequency. The difference frequency is plotted against the input S/N . The results are shown in Figure 13.25. In order to obtain a relatively smooth curve at each S/N level, the input is repeated 5000 times. Figure 13.25(a) and 13.25(b) show the results for a CW and 1000 MHz/ μ s signal, respectively. As expected, for the same input S/N value, the standard deviation of the CW signal is smaller than that of a 1000 MHz/ μ s signal.

13.14 Brute Force Approach [14]

Since the best sensitivity from the above study is about 3.4 dB lower than for a CW signal, it is interesting to determine whether increasing the processing load can improve the sensitivity. In this approach the summation is performed 53 times. Each summation consists of 20 frames. The approach is described as follows. In order to simplify the discussion, the first frequency bin in the first frame will be used as an example. The first summation is to add all the first frequency bins 20 times. This is the same as detecting a CW signal. The second summation includes the first half of the first bins and the second half of the second bins in all 20 frames. This approach matches a chirp signal equally divided into the first two frequency bins. The third summation includes the first $\frac{1}{3}$ from the first bins, the second $\frac{1}{3}$ of the second bins, and the last $\frac{1}{3}$ of the third bins. Since $\frac{1}{3}$ of 20 frames is not an integer, a round-off number should be used. This approach assumes the chirp is divided equally in three frequency bins. In other words, this approach checks all the chirp rates in about 20 MHz/ μ s steps.

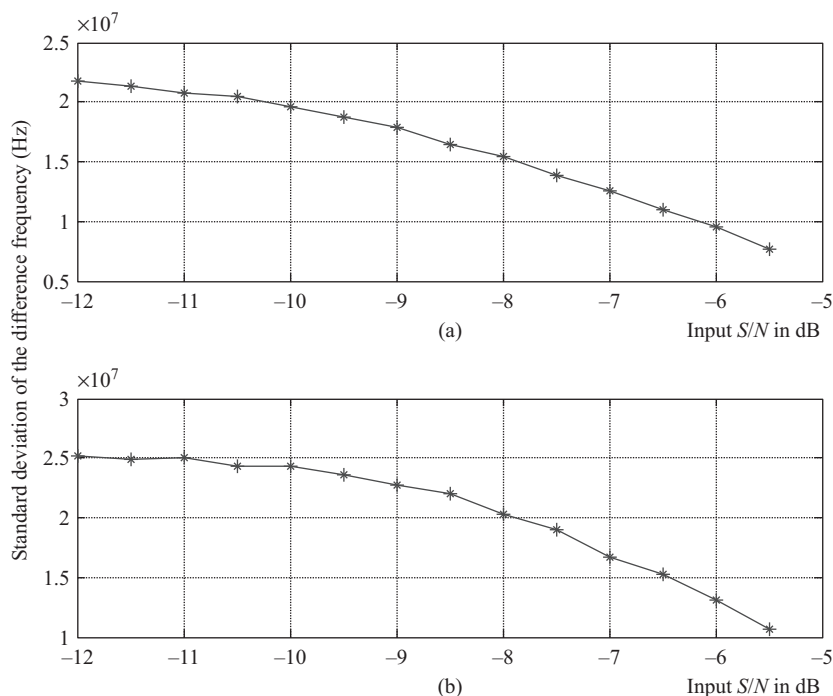


Figure 13.25 Standard deviation of the difference frequency versus input S/N: (a) CW signal and (b) 1000 MHz/ μ s signal.

The same pattern of operation is applied to frequency bins 2 through 53. After one set of summations there are 53 outputs representing different frequency bins and there are 53 summations, therefore there are a total of 2809 (53×53) summations and outputs. If the input signal includes downchirp, this total number of operations will be doubled. Since the summations are performed on frequency bins without picking the maximum, the threshold stays at 185. If any one of the 2809 outputs crosses the threshold, a signal is detected. The sensitivity is very close to the signal with known chirp rate [14]. The result is shown in row 10 of Table 13.2. Since in this approach the sensitivity measured is chirp rate dependent, the sensitivity is obtained from 1000 runs with random delta frequency from 0 to 1000 MHz. In this special example the chirp rate is tested every 20 MHz. If the chirp rate in every detection step is less than 20 MHz, the sensitivity should be better but with more operations. If the chirp rate step is more than 20 MHz, the sensitivity should be lower but with a smaller number of summations.

Comparing with the summation of maximum outputs, the number of operations is increased about 2809 times, but increased only 53 times in comparison with CW signal detection. The Table 13.2 shows that compared with the summation of maximum output method, the brute force approach improves the sensitivity by 3.4 dB for a CW signal and by 3.5 dB for a 1000 MHz/ μ s signal.

13.15 Using the Eigenvalue Method to Detect the Existence of Chirp Signal

In this section the eigenvalue method is used to detect the existence of a chirp signal. The approach is based on a frame-by-frame approach, which is suitable for receiver design. The assumption is that when every frame detects a wideband signal, the input is considered an FM signal. For 128 points of input data, the lags are 1, 10, and 26 and the threshold is 1.593 based on a 10^{-4} probability of false alarm (Section 12.12).

There are two parameters affecting the detection of chirp signal: the input power level and the chirp rate. In order to find the eigenvalue dependence on these two parameters, first the chirp effect will be studied. Keeping the input at a reasonable power level of $S/N = 20$ dB, the distribution of the smallest eigenvalue versus chirp rate from 10,000 runs is shown in Figure 13.26. Figure 13.26(a) is from a CW signal, and the eigenvalue is below the threshold of 1.593. Figure 13.26(b) shows the result with a chirp rate of 200 MHz/ μ s and some of the values are crossing the threshold. Figure 13.26(c) shows the eigenvalue with a chirp rate of 400 MHz/ μ s and a large number of eigenvalues are crossing the threshold.

The next step is to find the probability of detection of a chirp signal versus the chirp rate at different S/N levels. The power level changes from $S/N = 10$ dB to $S/N = 36$ dB in 2-dB steps. At each input power level the chirp rate changes from 60 MHz/50 ns (or 120 MHz/ μ s)

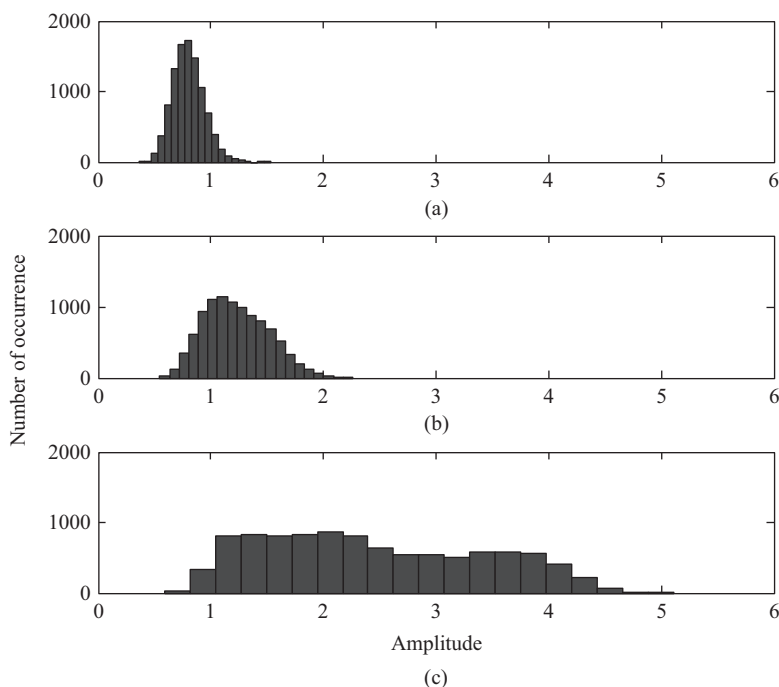


Figure 13.26 The smallest eigenvalues of a chirp signal at $S/N = 20$ dB: (a) CW signal, (b) 200 MHz/ μ s, and (c) 400 MHz/ μ s.

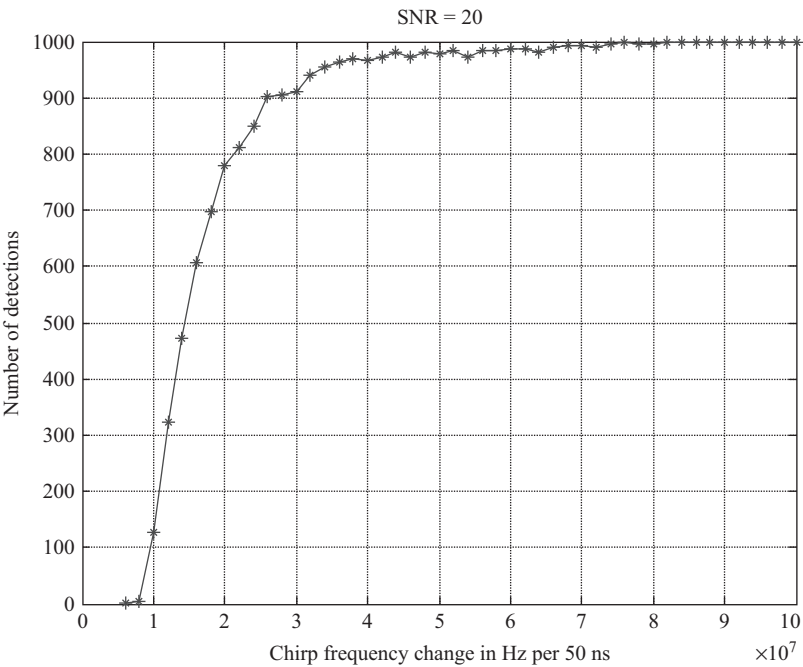


Figure 13.27 Probability of chirp signal detection versus chirp rate at $S/N = 20$ dB.

Table 13.3 Chirp rate versus S/N at the sensitivity level.

Chirp rate (MHz/ μ s)	2000	1600	1200	900	700	560	440	340	260	210	164	130	102	82
S/N (dB)	10	12	14	16	18	20	22	24	26	28	30	32	34	36

to 100 MHz/50 ns (2000 MHz/ μ s) in 2 MHz/50 ns (40 MHz/ μ s) steps. At each input condition there are 1000 runs with random initial phase and random starting and ending frequencies, but their difference equals the given chirp rate.

We conducted simulations at different S/N levels and generated 14 plots, but only the result of $S/N = 20$ dB is shown in Figure 13.27. It should be noted that the x -axis is in hertz per 50 ns. The 900 detection value (90%) is considered as the sensitivity level. Because the curve at this value is not smooth, the sensitivity is estimated at 28 MHz/50 ns.

The values read from the 14 plots (only one is shown in Figure 13.27) are listed in Table 13.3. These results are plotted in Figure 13.28. From this figure, one can conclude that the frame-by-frame approach with eigenvalue method is not very sensitive for detecting a chirp signal. Only at high S/N values and high chirp rates can a chirp signal be recognized.

13.16 Conclusion

Although there is no theoretical base, the frequency obtained from the amplitude and phase comparison methods derived for CW signals can improve the frequency accuracy on a chirp signal. These phenomena are shown through simulations.

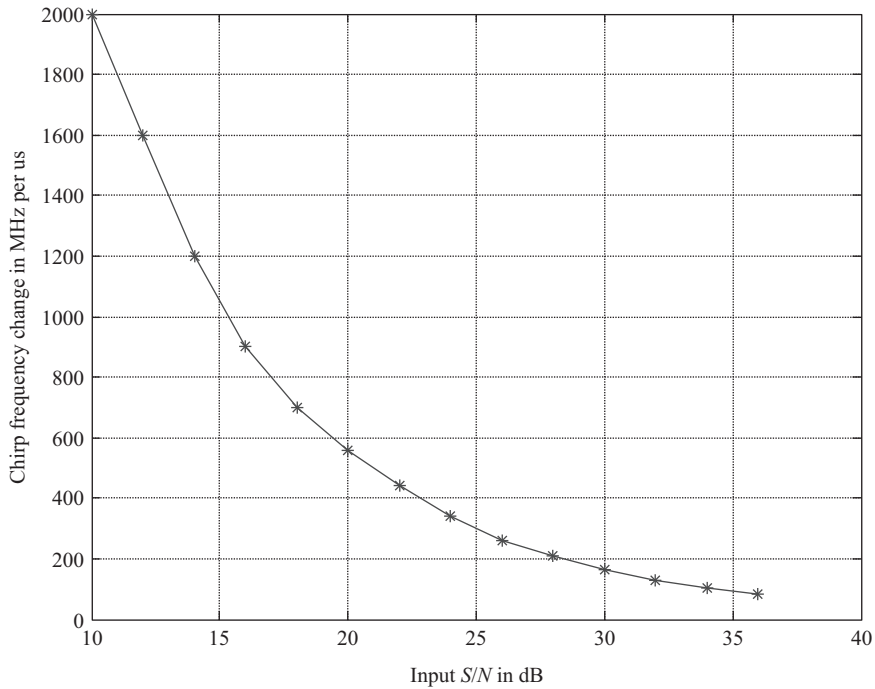


Figure 13.28 Sensitivity versus chirp rate (x-axis: S/N [in dB]; y-axis: chirp rate [in MHz/ μ s])

It is relatively difficult to detect a chirp signal without information on the chirp rate. It appears that the most efficient approach is to sum the maximum from every frame. The sensitivity obtained is about 3.4 dB worse than for a CW signal. The detection approaches through difference frequency produce lower sensitivity than the maximum summation method. However, once a signal is detected, the difference frequency will be within a certain range, and this can be considered as an additional requirement on the detection, decreasing the detection sensitivity.

The ideas of study can be used in receiver design. The actual receiver building is much more complicated than the study presented in this chapter. In this study the signal is known in the input data. In receiver design, the time of arrival (TOA) and the PW are unknown. Worse yet is the simultaneous signal problem. When the input PW is relatively long, the chance of simultaneous signals occurrence increases. All the methods except the brute force approach use the maximum output to calculate the input frequency from frame to frame. A relatively strong signal can mask the frequency measurement of a weak chirp signal. A general approach is to discard the strong output and its neighboring frequency bins of the FFT outputs to detect a second signal. In CW signal detection, this approach will not affect the detection of a second signal if they are separated far enough in frequency. However, in chirp signal detection, the separation of the two frequencies will not be a constant value. Sometimes the frequencies are close together and at other times they are widely separated. The two frequencies can even have the same value for a period of time. All these conditions must be carefully evaluated.

If a frame contains two signals and one signal is detected, the other frequency bins can be either ignored or pick up a second highest value. If the second highest peak is ignored, it will affect the total number of frames to detect the chirp signal. Under this condition, the pre-determined threshold will be affected. In order to limit the receiver study to a manageable size, it is suggested that only the case where one frame contains one pulsed CW and one chirp be considered. Even so, there will still be numerous input conditions. Choosing a set of signal conditions is the first step in studying receiver performance for detecting chirp signals.

References

- [1] Djuric PM, Kay SM. 'Parameter estimation of chirp signals'. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1990;**18**(12):2118–2126.
- [2] Wood JC, Barry DT. 'Tomographic time-frequency analysis and its application toward time-varying filtering and adaptive kernel design for multicomponent linear-FM signals'. *IEEE Transactions on Signal Processing* 1994;**42**(8):2094–2104.
- [3] Wang M, Chan AK, Chui CK. 'Linear frequency-modulated signal detection using radon-ambiguity transform'. *IEEE Transactions on Signal Processing* 1998;**46**(3):571–585.
- [4] Akay O, Boudreaux-Bartels GF. 'Fractional convolution and correlation via operator methods and an application to detection of linear FM signals'. *IEEE Transactions on Signal Processing* 2001;**49**(5):979–993.
- [5] Zhang X, Cai J, Wang W, Yang Y, Wen G. 'A new time-frequency transform algorithm for multi-LFM signals parameter estimation'. *Proceedings of the Ninth International Conference on Signal Processing*. New York: IEEE; 2008:116–119.
- [6] Wang P, Yang J, Du Y. 'A fast algorithm for parameter estimation of multi-component LFM signal at low SNR'. *Proceedings of International Conference on Communications, Circuits and Systems*. New York: IEEE; 2005:765–768.
- [7] Ikram MZ, Abed-Meraim K, Hua Y. 'Fast discrete quadratic phase transform for estimating the parameters of chirp signals'. *Proceedings of the Asilomar Conference on Signals, Systems and Computers*. New York: IEEE; 1997:798–802.
- [8] Klauder JR, Price AC, Darlington S, Albersheim WJ. 'The theory and design of chirp radars'. *Bell System Technical Journal* 1960;**39**(4):745–808.
- [9] Skolnik MI. *Introduction to Radar Systems*, 3rd ed. New York: McGraw-Hill; 2001.
- [10] Tsui J. *Special Design Topics in Digital Wideband Receivers*. Boston: Artech House; 2010.
- [11] Lin D. Air Force Research Laboratory. Private communication.
- [12] Liou L. Air Force Research Laboratory. Private communication.
- [13] Tsui J. *Fundamentals of Global Positioning Systems: A Software Approach*, 2nd ed. New York: John Wiley; 2005:chap. 10.
- [14] Broadstock M. Miami University, Oxford, Ohio. Private communication.

Concept of Analog-to-Information

14.1 Introduction [1]

In this chapter the concept of analog-to-information (A-to-I) is introduced. The name A-to-I comes from the Analog-to-Information Receiver Development concept presented in a Defense Advanced Research Projects Agency (DARPA) program. The goal of A-to-I is to use a low sampling frequency to cover a wide input frequency band beyond the Nyquist sampling rate allowed [2–5]. In this chapter we present the analysis results of one of the A-to-I technology development concepts by Northrop Grumman. We would like to acknowledge its contribution and the support of the Air Force Research Laboratory (AFRL).

The idea of A-to-I is that the input frequency band is sparsely occupied by input signals. An electronic warfare (EW) receiver is required to detect many signals but usually only a few concurrent signals, such as two to four signals including continuous wave (CW) are present over a broad bandwidth, and thus fitting this requirement. The A-to-I problem is generally studied from a theoretical point of view. The approach is to reconstruct the input signals from the digitized samples. If the original signals are reconstructed, the conventional fast Fourier transform (FFT) method can be used to find the parameters of interest, such as carrier frequency, pulse amplitude, etc. This approach is time consuming because the input signals must be reconstructed first. Since the goal of this study is to apply the idea to an EW receiver for which real-time processing is a crucial requirement, the parameters need to be obtained from the digitized data directly without reconstructing the original signal. The materials presented in this chapter are obtained from a Northrop Grumman DARPA project managed by the AFRL.

Several A-to-I approaches such as random sampling (or nonuniform sampling) have been proposed [4, 5]. The concept discussed in this chapter is the random modulation pre-integrator (RMPI) [6]. The received analog signal is modulated with a Gold code first and digitized data are collected after the modulation. In this study, the equivalent digitizing rate is less than 400 MHz (or 200 MHz Nyquist bandwidth) and the input bandwidth covered is over 2000 MHz, which is over 10 times folding.

This specific concept will be discussed in detail, including data collection and frequency measurement. A receiver designed with this concept will also be presented. The performance of the receiver, such as frequency reading and amplitude measurement, will be discussed. Especially important are sensitivity and the two-signal measurement capability. The low sampling rate will fold noise and multiple signals into the baseband. These problems will be evaluated through simulations.

14.2 Data Collection

First we discuss how to digitize analog signals for an A-to-I receiver. The digitized data are generated through the following procedure. An internal analog clock running at 5 GHz is used. This internal clock rate determines the receiver's maximum bandwidth, which is half of the clock rate, 2.5 GHz in this case. This clock is used to generate a series of Gold codes [7] that will be used to modulate the input signal. The input analog signal's polarity is varied based on the Gold code's value (+1 or -1), and it is done in the analog domain. The Gold code can be generated by bit-wise exclusive OR operation of two pseudo-random bit sequences (PRBS). If one of the PRBS sequences repeats after 63 bits and the other one repeats after 52 bits, the resulting Gold code has thus a total length of 3276 (63×52) and the code repeats after this length [8]. Four Gold codes are used as described in the next section. A longer Gold code can be generated to detect a longer signal. In order to simplify the discussion, the analog input signal modulated by a clock at 5 GHz will be called analog points, although the input data are never digitized into a point at this frequency. The analog points come out at 5 GHz (or 0.2 ns per analog point) and the integration time is 52 analog points, which means every 52 analog points a digital datum is generated. The equivalent sampling rate, f_{s1} , is about 96.154 MHz ($= 5000/52$) and the corresponding sampling time, t_{s1} , equals 10.4 ns ($1/f_{s1}$). With this relatively low sampling frequency, an analog-to-digital converter (ADC) of with a high number of bits can be used.

There are four channels in parallel, so the equivalent sampling rate f_s equals 384.62 MHz ($4 \times f_{s1}$) and the equivalent sampling time is 2.6 ns. The baseband width is $f_s/2 = 192.31$ MHz. The analog modulation is 5 GHz and the Nyquist input bandwidth is 2.5 GHz. Thus the input bandwidth is folded 13 times ($2500/192.31$) into the baseband. The data generated from each channel are not coming out at the same time but are staggered by 2.6 ns (13 analog points). One group of digital data consists of four points each, one of which comes from one channel. In the following receiver simulation we use seven or nine groups of data to find the input signal. For the receiver simulation we also assumed that the input frequency range of the receiver is from 100 to 2400 MHz to avoid frequencies close to the boundary of the Nyquist zone. This bandwidth leaves room for the skirt of the analog filter in front of the receiver. Thus with a sampling rate of 384.62 MHz, the receiver can cover 2300 MHz.

Since the Gold code is 3276 analog points long after 63 ($3276/52$) digitized points, the code will repeat again. The corresponding time is 655.2 ns (10.4×63). However, in order to take 63 groups of digitized data, three more digital data are needed and the total number of analog points is 3315 ($3276 + 39$). The reason why we need to add 39 analog points can be understood from Figure 14.1. The total analog points contains 63 periods, each of which has 52 analog points (which is 3276 analog points) plus an extra 39 analog points between the start point of digitized point 253 (code 1) and the start point of digitized point 256 (code 4) as demonstrated in Figure 14.1. The resulting equivalent time is 663 ns. The digital point 253 is collected with the same Gold code as digital point 1, as shown in Figure 14.1.

First, the number of groups must be determined to find the input signal. Since the shortest pulse of the anticipated detection is about 100 ns, nine groups of digital points generated from 507 analog points ($52 \times 9 + 39$), corresponding to 101.4 ns, is used. This choice closely matches the minimum pulse width. With this time resolution of about 100 ns, the corresponding frequency resolution is about 10 MHz.

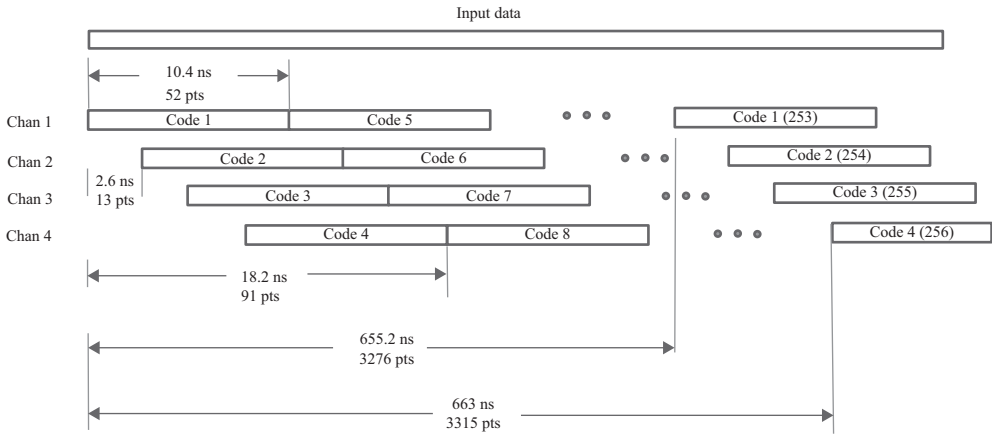


Figure 14.1 Architecture of digital data collection (points in this plot refer to analog points).

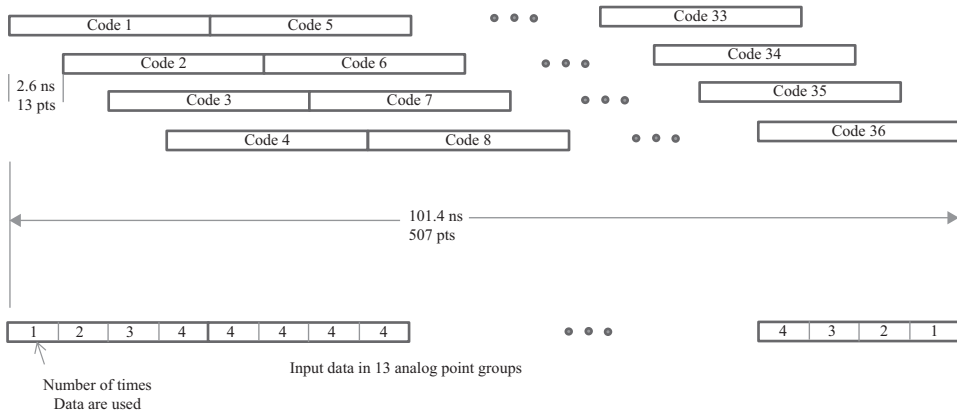


Figure 14.2 Data used in one group (points in this plot refer to analog points).

Figure 14.2 shows the data used in the following studies. The total data can be grouped into 39 thirteen-point groups ($39 \times 13 = 507$). As shown in Figure 14.2, to generate 36 digital points the first group of analog points is used only once, the second group is used twice, and the third group is used three times. Also shown in Figure 14.2 is that the last three groups of analog points are used three, two, and one time, respectively. The rest of the groups of analog points are used four times. This effect is somewhat like a window in the time domain.

14.3 Frequency Calculation

The first parameter to be found from the compressed data is the input signal frequency. Based on the data length of about 100 ns, the frequency resolution is selected to be 10 MHz. However, this choice will generate erroneous frequency readings. The next choice is to

use 5 MHz, and the frequency reading is acceptable, which means that under various tests the frequency error is within 10 MHz. Finer frequency resolutions will increase the computation time.

In order to find an input signal, digital points collected using the procedure outlined in Section 14.2 are correlated with plates, each of which corresponds to a specific frequency. This section describes the operation to generate these plates.

For a specific frequency ω , a plate P can be generated through the following equation [8]:

$$P = \Phi V_m (V_m^T \Phi^T \Phi V_m)^{-1} V_m^T \Phi^T \quad (14.1)$$

where

$$V_m = \left[\begin{array}{cc} 1 & 0 \\ \cos \omega t & \sin \omega t \\ \cos 2\omega t & \sin 2\omega t \\ \vdots & \vdots \\ \cos 506\omega t & \sin 506\omega t \end{array} \right] \quad (14.2)$$

$\underbrace{\hspace{10em}}_2 \quad \underbrace{\hspace{10em}}_{507}$

The size of the V_m matrix is 507×2 and t is the time resolution in the analog domain. Since the Gold code is operated at 5 GHz, the equivalent is 0.2 ns and ω is the angular frequency ($\omega = 2\pi f$). Its value depends on the frequency resolution of the plate. For a 5-MHz frequency resolution, the ω value changes accordingly.

The Φ matrix represents the Gold code and can be written as

$$\Phi = \left[\begin{array}{cccccccccc} \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & \dots & & & & & & \\ & & & \dots & & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \end{array} \right] \quad (14.3)$$

$\underbrace{\hspace{10em}}_{507} \quad \underbrace{\hspace{10em}}_{36}$

The matrix size is 36×507 . In every row the “...” in (14.3) represents 52 points of data containing only 1 and -1 according to the Gold code shown in Figure 14.2, and the rest are zeros. The locations of 1, -1 , and 0 can be located with reference to Figure 14.2. In (14.3), Channel 1’s Gold code is used in row 0, 4, 8, etc. Channel 2’s Gold code is used in row 1, 5, 9, etc. Channel 3’s Gold code is used in row 2, 6, 10, etc., and Channel 4’s Gold code is used in row 3, 7, 11, etc. For a specific row in this matrix, the starting point of its Gold code is 13 analog points to the right of the starting point of Gold code in the row above.

The matrix in (14.3) can be used to process 36 input data. To process a longer data in one operation, a larger matrix is necessary. For example, based on the arrangement shown in Figure 14.1, a 252×3315 matrix can be built in the same fashion. If one would like to process a long data frame by frame using a small frame size, a small Gold code matrix can be obtained from a big Gold code matrix. An example is presented in Section 14.8. In our study, a 312×4096 Gold code matrix provided by Northrup Grumman is used. This matrix is available upon a request via books@theiet.org.

After the operation of (14.1), the plate P , a 36×36 matrix, is obtained. This represents one plate at frequency f . Since the input frequency range of the receiver is from 100 to 2400 MHz and with 5-MHz frequency resolution, there are 460 frequency steps. For each frequency there is a plate, with a total of 460 plates, and each one is a 36×36 matrix.

In order to calculate the frequency, the digitized data y is operated with the plate to obtain an output Y as

$$Y = yPy', \quad (14.4)$$

where the input data y is a vector of 1×36 . This operation generates only one point. Since there are 460 different frequencies, there are also 460 Y outputs. The index with the highest output represents the input frequency. Although this Y value is related to the amplitude of the input signal, its value changes with input frequency and initial phase angle. The only use of this Y value is to find the input frequency. The signal amplitude will be found later.

14.4 Probability of False Alarm and Probability of Detection

Following the conventional approach of studying the sensitivity of a receiver, the probability of false alarm will be studied first. If the distribution of the output with noise as input is known, this is an easy task. If the distribution is unknown, it is a rather difficult task to obtain an accurate result. We will use a numerical approach for this study. Use noise as the input and, through the procedure mentioned in the previous section, 460 points in the frequency domain will be obtained each time. This operation is repeated 10,000 times and two sets of data are collected. The first set contains all the output data, and there are $460 \times 10,000$ points. The second set contains 10,000 points, and they are the maximum of every 460 outputs. The histograms of the two data sets are shown in Figures 14.3 and 14.4.

The theoretical distribution of the noise output shown in Figure 14.3 is unknown. It is also difficult to approximate the distribution. Under this condition, it is difficult to choose a threshold for a given probability of false alarm rate. The other approach is to select the maximum value in Figure 14.4. Theoretically one can claim that the threshold selected can provide a false alarm rate of 10^{-4} . However, the threshold obtained this way varies significantly. If the tenth largest value is the threshold, its variation is less and the corresponding probability of false alarm is about 10^{-3} . For example, four 10,000 runs are performed and the maximum values are 1931, 1869, 1697, and 1851, and their variation is relatively large. The tenth largest values are 1502, 1404, 1405, and 1471 with less variation. The average of these four values is used as the threshold, and this value is about 1446. This approach can only be considered as an approximation. In order to obtain an accurate result through simulation, a large number of runs will be used, which is, however, not practical to perform.

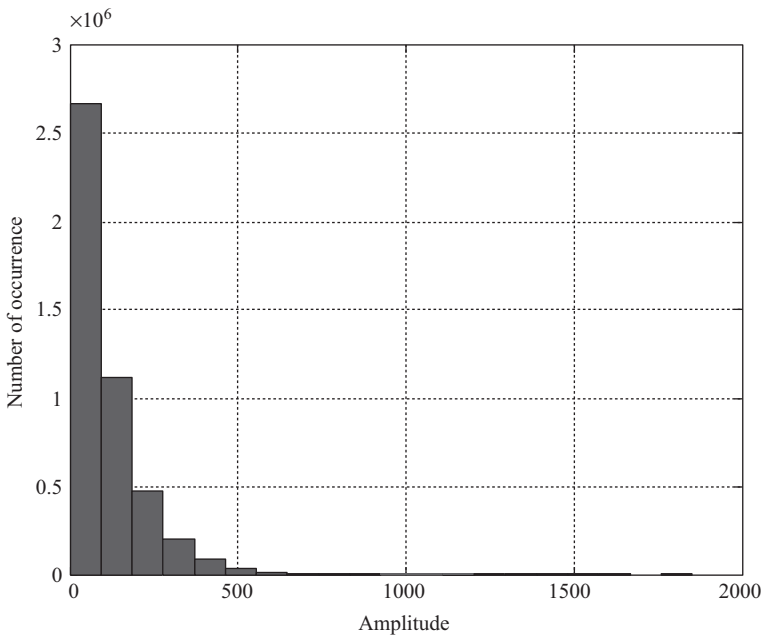


Figure 14.3 Amplitude in the frequency domain with noise as the input.

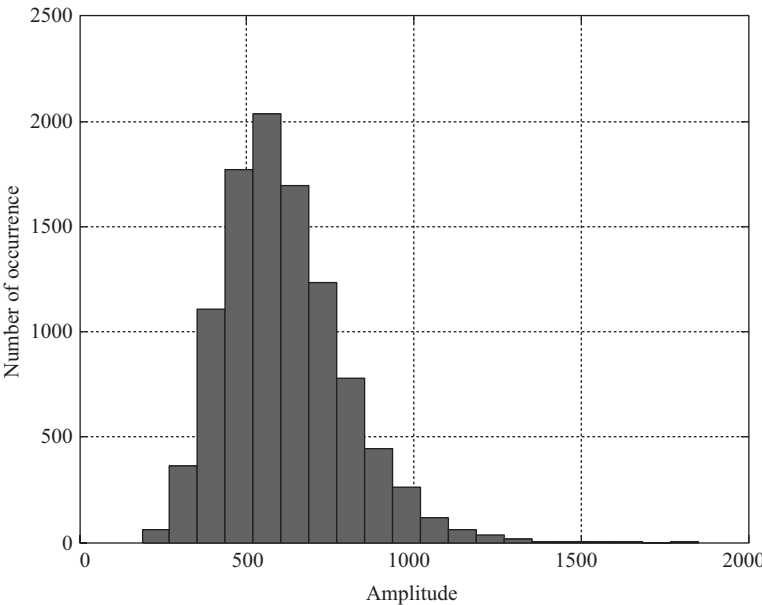


Figure 14.4 Maximum output in the frequency domain in 10,000 runs.

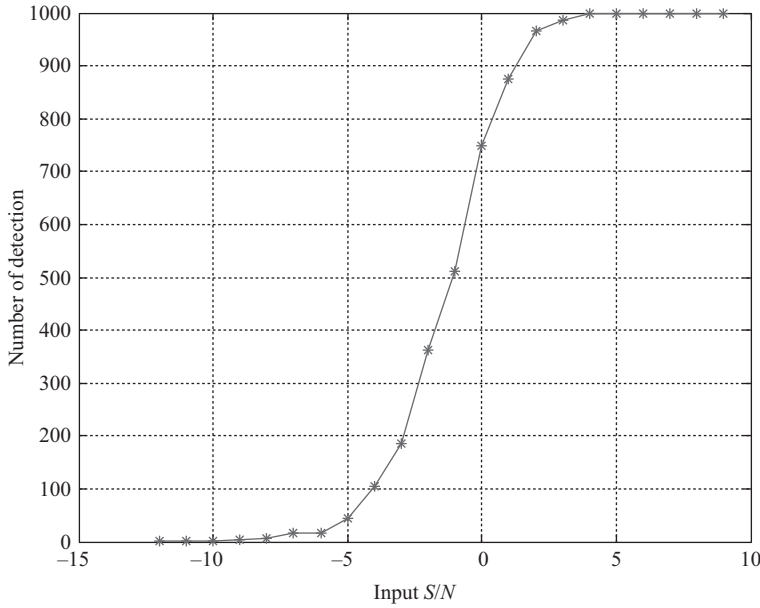


Figure 14.5 Number of detections (out of 1000) versus S/N ratio.

Once a threshold is determined, the probability of detection can be obtained. The approach to find the probability of detection is to check the amplitude of the output. The amplitude of the input signal-to-noise (S/N) ratio is changed from -12 to 9 dB in 1-dB steps. At each input level the signal is inputted 1000 times. The frequency and the initial phase of the signal are randomly selected from 100 to 2400 MHz and 0 to 2π , respectively. The maximum output in the frequency domain is monitored. If the output is higher than the threshold, the signal is considered detected. The results are shown in Figure 14.5. At 90% detection the S/N value is about 1 dB.

For a 100-ns pulse window using the FFT operation and a probability of detection of 90% with a false alarm rate of 10^{-7} , the S/N value is about -7.4 . If the false alarm rate is 10^{-3} , the S/N value is about 2.5 dB lower, or S/N is about -9.9 dB, and this value of 2.5 dB is obtained from chart reading [9]. For the A-to-I system, the bandwidth is 2500 MHz, although the input signal is from 100 to 2400 MHz. With a sampling frequency of 384.62 MHz, the noise is increased about 11.1 dB $\{10 \log[2500/(384.62/2)]\}$. Based on this argument, the sensitivity should be about 1.2 dB ($11.1 - 9.9$), very close to the simulation result.

14.5 Frequency Measurement Accuracy

In this study the accuracy of the measured frequency is evaluated. The frequency resolution is predetermined rather than obtained from the length of the input data, such as in the FFT operation. The frequency resolution is chosen to be 5 MHz, which is about half the resolution of an FFT operation with 100 ns. If a frequency resolution of 10 MHz is used, large

frequency errors will be produced. Two approaches are used to determine the frequency accuracy. In the first one, the frequency changes from 100 to 2400 MHz in 1-MHz steps with a random initial phase from 0 to 2π . In the second approach, the input frequency is randomly selected from 100 to 2400 MHz with a random initial phase. In both cases the input is applied 2300 times and the S/N ratio is set at 10 dB, which is well above the 1 dB S/N ratio requirement for a 90% probability of detection. The error frequency, which is defined as the measured frequency minus the input frequency, is plotted at each input frequency. The results from the sequential frequency are shown in Figure 14.6. In this figure the results are well behaved. The maximum frequency error is 5 MHz with a bias of -46 kHz and a standard deviation of 1.498 MHz.

The results for the random input frequency are shown in Figure 14.7. The maximum frequency error is less than 5 MHz, with a bias of -3.6 kHz and a standard deviation of 1.532 MHz. Comparing Figures 14.6 and 14.7, it should be noted that the errors in Figure 14.6 are somewhat structured, because the input frequency increases by 1-MHz steps.

Another test is to measure the frequency accuracy at the sensitivity level. In this test the input power level is set close to the sensitivity level with an S/N ratio around 1.2 dB, and 1000 runs are performed to obtain the error frequency. The input frequency and the initial phase are randomly selected. The signal output must be higher than the amplitude threshold of 1446 to be detected. A frequency threshold is arbitrarily chosen to be 10 MHz. For detected signals, if the frequency error is greater than the frequency threshold, it is considered as an erroneous frequency. The results are shown in Table 14.1.

The wrong frequency in percent is the number of wrong frequency readings divided by the number of detections. This simple illustration verifies that 90% of the input can be detected at an S/N close to 1.3 dB.

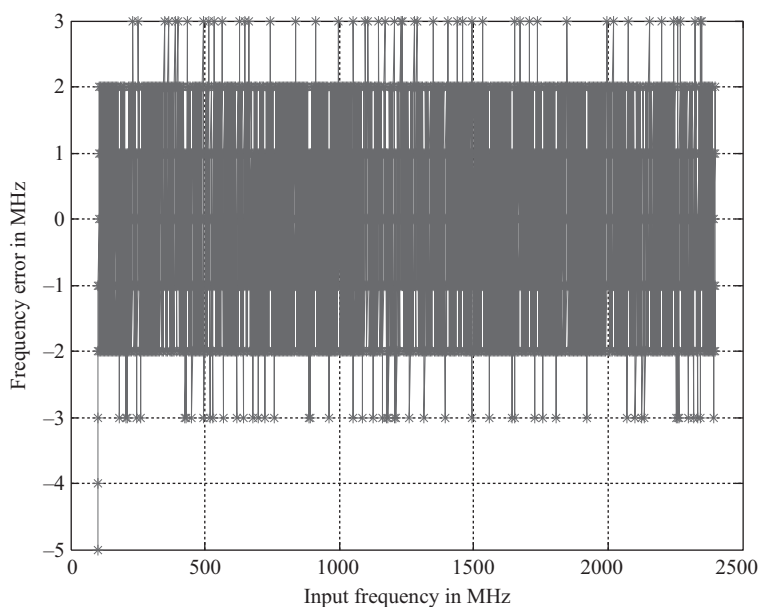


Figure 14.6 Frequency errors versus input frequency.

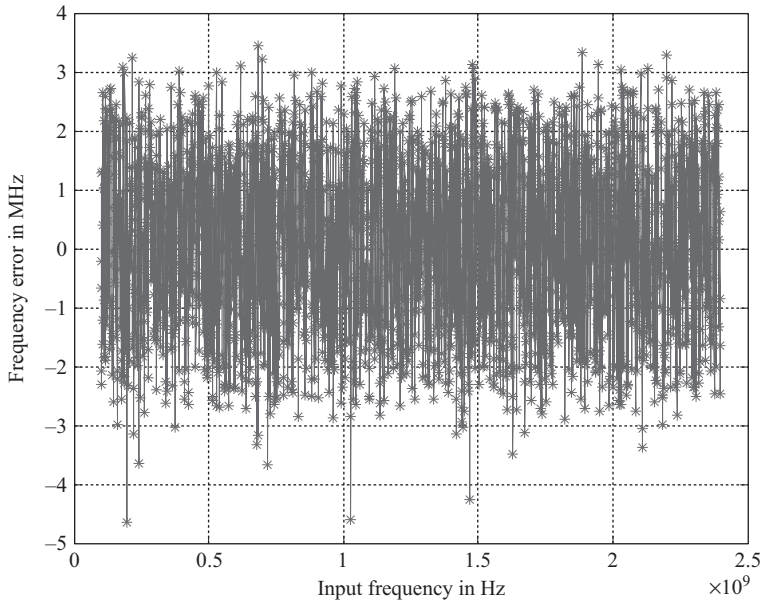


Figure 14.7 Frequency errors for random input frequency.

Table 14.1 Frequency measurement near the sensitivity level.

Input S/N (dB)	No. of detections	No. of wrong frequencies	Wrong frequency (%)
1	892	3	0.336
1.2	895	8	0.894
1.4	910	7	0.769
1.6	929	5	0.538

14.6 Amplitude Measurement

It is indicated in Section 14.3 that the Y value calculated from (14.4) does not represent the signal amplitude. This phenomenon can be illustrated as follows. In this study the input signal does not contain noise and the input frequency is arbitrarily selected as one value. The initial phase of the input signal is changed from 0 to 2π in 100 uniform steps. The output Y value is recorded in each step. The result is shown in Figure 14.8 and the Y value changes with the initial phase.

In this figure the minimum and maximum values of Y are 682 and 1096, respectively, and their ratio is 0.6. The signal amplitude at a certain frequency, ω , can be found from the following equation:

$$Q = (V_m^T \Phi^T \Phi V_m)^{-1} V_m^T \Phi^T. \quad (14.5)$$

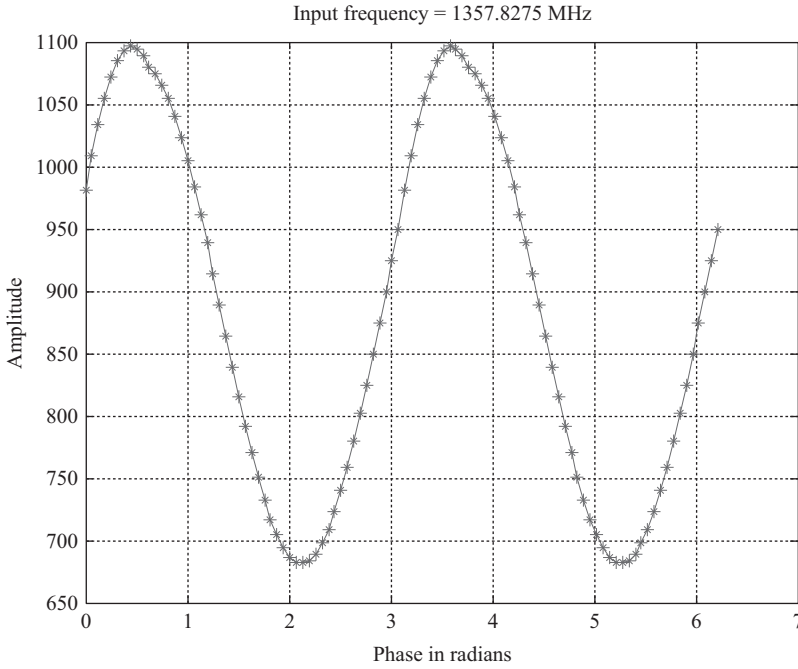


Figure 14.8 Output Y values versus initial phase angle.

This equation is similar to (14.1). The Q output is a matrix of 2×36 . Two values of a_1 and a_2 can be calculated from Q :

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = Qy, \quad (14.6)$$

where a_1 and a_2 are the values in the following equation:

$$s(t) = a \sin(2\pi ft + \theta) = a \sin(2\pi ft) \cos \theta + a \cos(2\pi ft) \sin \theta \equiv a_1 \sin(2\pi ft) + a_2 \cos(2\pi ft), \quad (14.7)$$

where

$$a_1 \equiv a \sin \theta, \quad a_2 \equiv a \cos \theta, \quad \text{and} \quad a = \sqrt{a_1^2 + a_2^2}.$$

The angle of θ is the signal's initial phase and y is the digitized input signal.

Using the identical input frequency and initial phase setting, we calculated the amplitude, with the results shown in Figure 14.9. The amplitude also changes with the initial phase. The minimum and maximum amplitudes are 0.827 and 1.019, respectively, and their ratio is 0.812.

The amplitude ratio values are expressed in decibels, such as

$$\text{dB} = 20 \log(R), \quad (14.8)$$

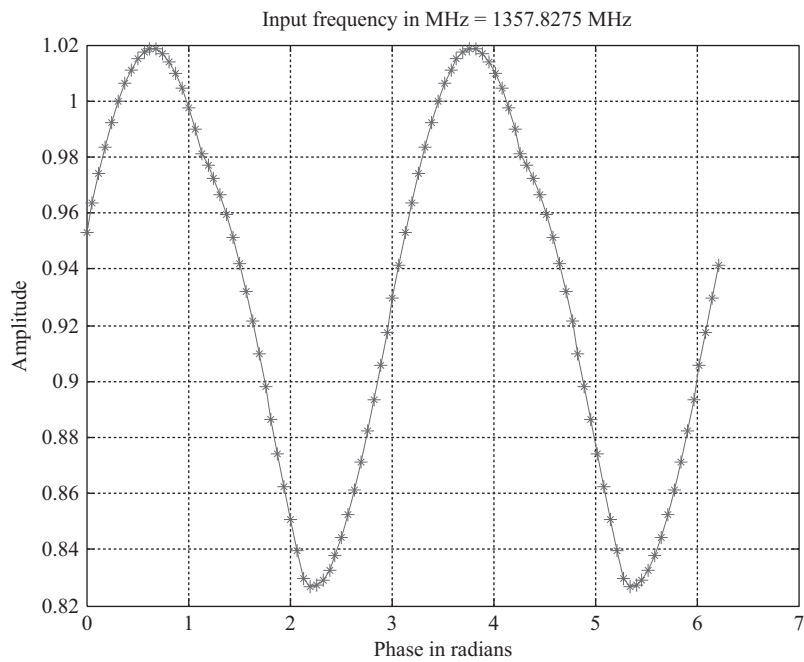


Figure 14.9 Output amplitude values versus initial phase angle.

Table 14.2 Outputs calculated from two methods.

Method	Runs	Maximum difference (dB)	Decibels calculated from (14.9)
Y output	1000	13.24	2.23
	5000	13.98	2.26
Amplitude output	1000	3.24	0.41
	5000	5.68	0.42

where R is the minimum-to-maximum ratio. Using the Y value approach in (14.4), this value is 4.1 dB. Using the signal amplitude in (14.5), this value is 1.81 dB, which is better than the Y value approach of 4.1 dB.

Identical input signals with random frequency and initial phase but no noise are used to compare the Y and amplitude outputs. The input is run 1000 times. The maximum, minimum, and standard deviation are recorded. The results are listed in Table 14.2.

The maximum difference in decibels is calculated by taking 20 times the logarithm of the maximum divided by the minimum outputs. This number will increase if more runs are performed. The decibel value calculated from the standard deviation is obtained from

$$\text{dB} = \left| 20 \log \left(\frac{\text{mean} - \text{std}}{\text{mean}} \right) \right|,$$

(14.9)

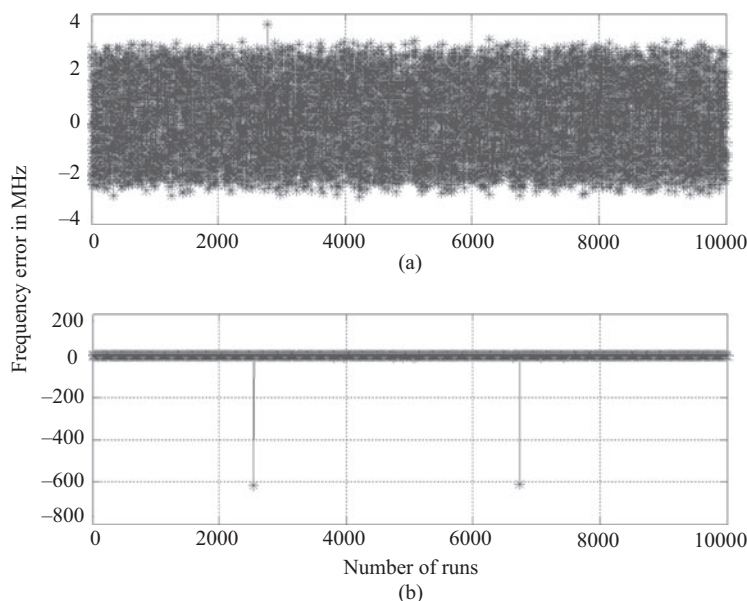


Figure 14.10 Frequency error measured by two methods: (a) Y output and (b) amplitude output.

where *mean* represents the mean value and *std* represents the standard deviation. When the number of runs increases from 1000 to 5000, the maximum difference increases, but the decibels calculated from the standard deviation stay about the same.

The next question is whether the signal amplitude can be used to determine the input frequency. The answer is no. Most of the time the amplitude approach will produce the correct frequency. With random frequency and initial phase but no noise, the amplitude approach will still generate a few erroneous frequencies (such as three to five in 10,000 runs). Therefore this approach is not acceptable for frequency measurement. Figure 14.10 shows the results with 10,000 runs. The input signals for the two approaches are identical. Figure 14.10(a) shows the frequency error calculated with the Y values and Figure 14.10(b) shows the frequency error with the amplitude method. In this specific case, there are two large frequency errors.

The approach to finding the frequency and amplitude of an input signal should proceed as follows. First, the frequency is found, and this provides the index of the Y output. Once this index is determined, the amplitude a can be found through (14.5) to (14.7). There are 460 amplitude values. Instead of using the maximum of 460 amplitude values, the amplitude value with the desired index will be used as the amplitude of the signal.

14.7 Study of Determining Two-Signal Frequencies

The approach to find two signals is to find one signal first and use the discrete prolate spheroidal sequences task, MATLAB[®] (MathWorks, Natick, MA) ‘dpss’ command, to filter the signal from the digitized data [10]. Then the same approach can be used to find the second signal in the remaining data. It is important to find out whether the correct frequency can be found under two signal conditions. The simulation setting to study the two-signal scenario is to

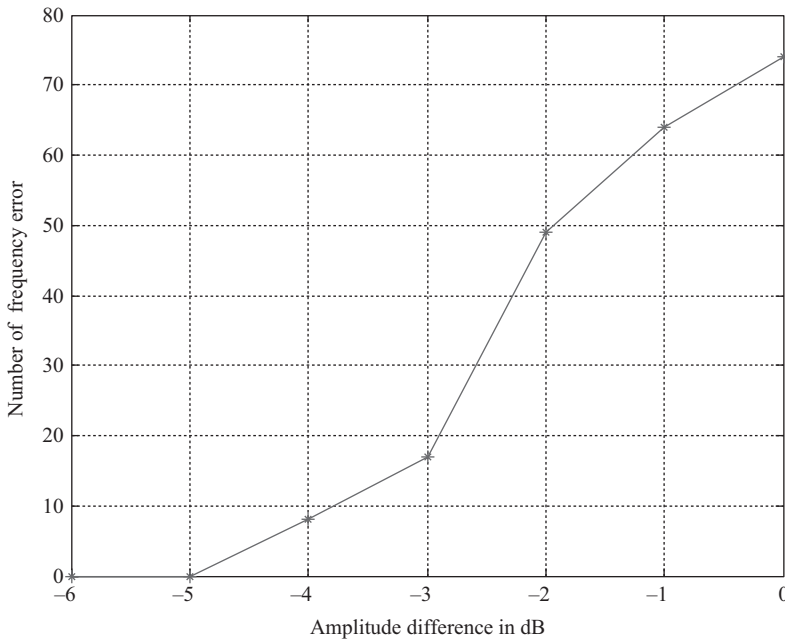


Figure 14.11 Number of error frequencies measured on two signals versus amplitude difference on 10,000 runs (x-axis: signal power difference in decibels).

use two signals of comparable amplitudes. Their frequencies and initial phases are randomly selected, but the difference in frequency must be greater than 20 MHz. There is no noise added in this test. The signal amplitude difference is expressed in decibels, and at each difference 10,000 signals will be generated. Only one frequency will be read. If the measured frequency is within 10 MHz of either input frequency, the result is considered correct. The result is shown in Figure 14.11.

In this figure, the number of error frequencies increases when the second signal's amplitude is approaching the first signal's amplitude, and it is about 70 in 10,000 runs when the two signals are of the same amplitude. When the amplitude difference between the two signals reaches 5 dB, no error frequency is detected in this special run. For an EW receiver, this is very undesirable. An erroneous frequency reading will create many problems downstream in the signal processing.

However, it is interesting to know if the first frequency is read wrong whether the second signal can be read correctly. For this test, two signals of the same amplitude with random input frequency and a difference greater than 20 MHz are used as inputs. The phases of the two signals are randomly selected between 0 and 2π deg, and there is no noise. The first frequency is measured and then compared with both input frequencies. The smaller error frequency between the two is considered as the first signal frequency error. A similar approach is applied for the second frequency measurement. For this test, 500 runs were conducted, and the frequency errors are shown in Figure 14.12.

In this specific case there are three large errors in both Figures 14.12(a) and 14.12(b). The first error in Figure 14.12(a) and the second error in Figure 14.12(b) are independent.

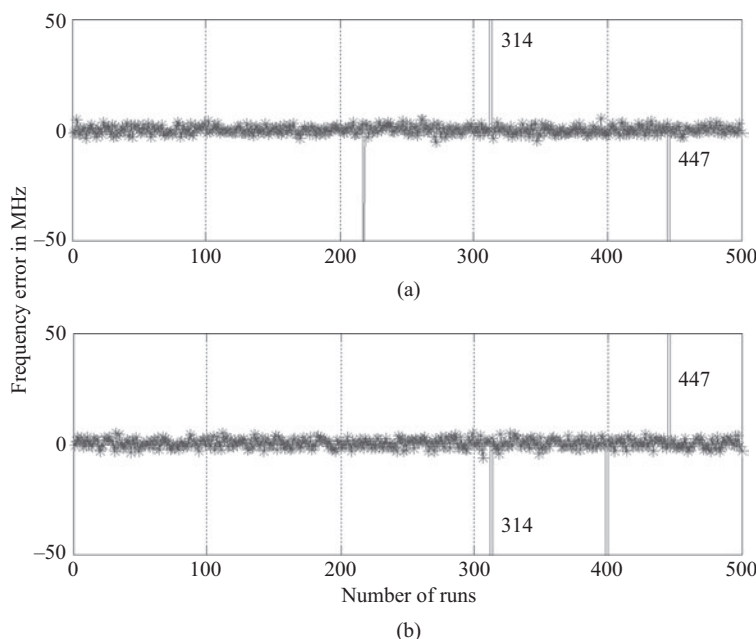


Figure 14.12 Frequency errors for two signals of the same amplitude: (a) first measured frequency and (b) second measured frequency.

This means that if the first signal is measured correctly, the second one could be wrong. Similarly, if the first signal is measured wrong, the second one could still be correct. It appears that the frequency errors are independent of each other. However, in measurements 314 and 447, both frequencies are measured wrong. For other measurements the same conclusion can be obtained; that is, when one frequency is measured wrong, the second frequency may also have a wrong result.

14.8 Two-Frame Approach for Detecting Two Signals

In conventional receiver design, one frame of data will not be solely used to determine whether there is a signal or not, even if the signal amplitude crosses a certain threshold. Usually two frames of results are used to confirm the existence of a signal. In this study we determine if the two-frame approach can reduce false detection. The frames can use the same Gold code or different ones. In the two-frame study, the signal will be generated continuously to cover the two frames, but data processing will be done on an individual frame. One frame of data is processed each time, which can be roughly considered as noncoherent processing.

The approach is to generate two frames of data using an input of about 200 ns. In our study, the size of Gold code matrix is 312×4069 . This Gold code matrix is provided by Northrop Grumman as mentioned in Section 14.3. To process a frame of 36 data points, a 36×507 Gold code matrix needs to be built from the big matrix. The first half of the input (or first frame) data will be processed once. The second half will be processed twice using different Gold codes. The first time uses the same Gold code as the first half, which is from row 1 to 36

and column 1 to 507 of the big Gold code matrix. The second time uses the Gold code from row 37 to 72 and column 469 to 975 of the big Gold code matrix.

In both the first and second frames, two frequencies are obtained, and they are written as f_1 , f_2 and f_1' , f_2' . These two pairs of frequencies are compared; e.g., f_1 is compared with f_1' and f_2' , and f_2 also is compared with f_1' and f_2' . If the difference frequencies are less than 10 MHz, it is considered the same frequency and a signal is detected. If both frequency differences are less than 10 MHz, two frequencies are detected. If only one difference frequency is less than 10 MHz, one frequency is detected. If both difference frequencies are greater than 10 MHz, no signal is detected. Once a frequency is obtained, it must be compared with one of the input frequencies to find the error frequencies. If two signals are detected, both error frequencies must be found. If one signal is detected, one error frequency will be generated.

Six runs of simulations, each consisting of 10,000 simulations, are performed. The results obtained from using the same Gold code arrangement in both frames are included in Table 14.3, while results obtained from using two different Gold code arrangements in each of two frames are listed in Table 14.4.

For the two frames with different Gold codes, the missing of one and two signals is slightly better, but the generation of an error frequency of greater than 10 MHz is greatly reduced.

The results of the third run in Tables 14.3 and 14.4 are shown in Figure 14.13. There are six plots in this figure. Figures 14.13(a) through 14.13(c) show the results obtained from using the same Gold code (the third run in Table 14.3). Figures 14.13(d) through 14.13(f)

Table 14.3 Missing detection and frequency errors for two frames using the *same* gold code.

Runs	Missing first signal	Missing second signal	First wrong	Second wrong
1	141	114	2	8
2	147	122	2	4
3	130	123	1	8
4	149	107	3	1
5	130	123	1	8
6	149	108	3	7
Average	141	116.17	2	6

Table 14.4 Missing detection and frequency errors for two frames using *different* gold codes.

Runs	Missing first signal	Missing second signal	First wrong	Second wrong
1	125	121	0	1
2	133	101	0	1
3	119	129	0	2
4	136	104	0	0
5	119	129	2	0
6	137	101	0	0
Average	128.17	114.17	0.33	0.67

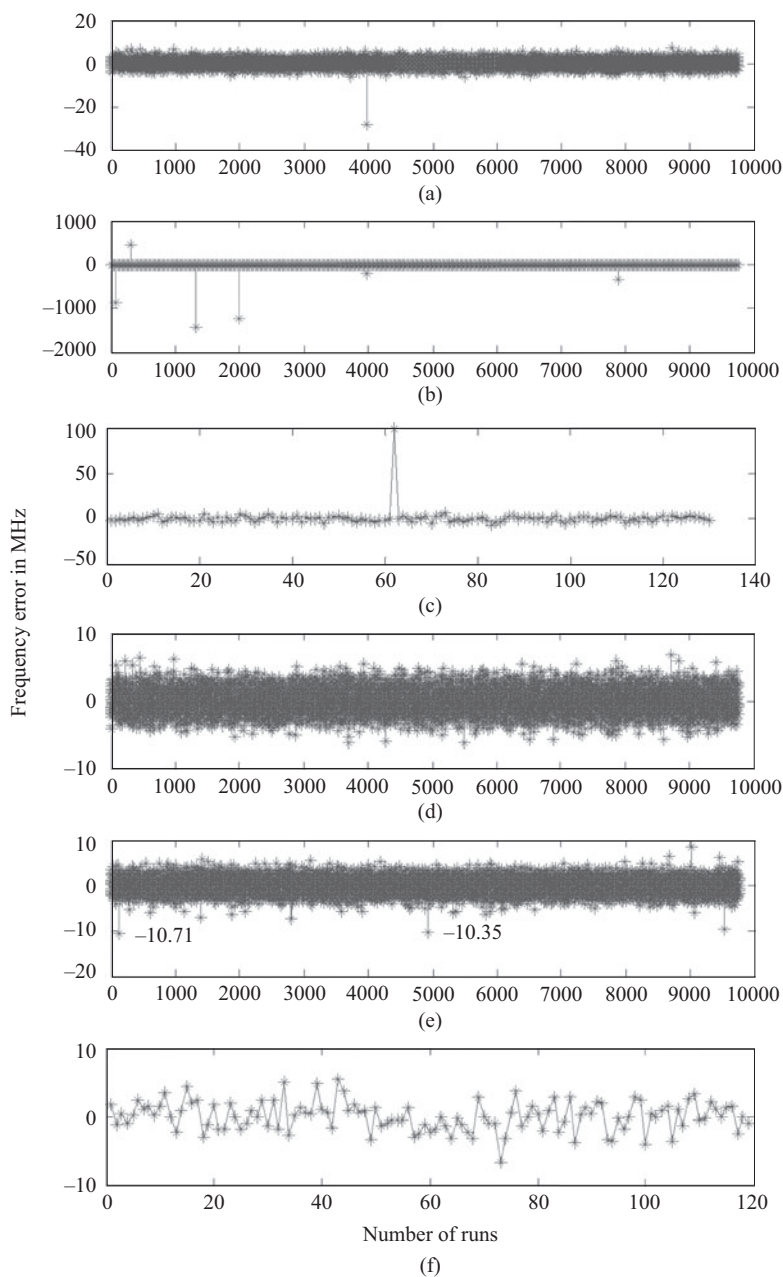


Figure 14.13 Frequency errors on two-frames signal detection with two signals of the same amplitude: (a) first signal error on two signals detected (same Gold code), (b) second signal error on two signals detected (same Gold code), (c) frequency error on one signal detected (same Gold code), (d) first signal error on two signals detected (different Gold codes), (e) second signal error on two signals detected (different Gold codes), and (f) frequency error on one signal detected (different Gold codes).

show the results obtained from using different Gold codes (the third run in Table 14.4). The reason why the data from this specific run are presented here is that it contains two second wrong frequencies when different Gold codes are used. Figures 14.13(a) and 14.13(b) show the frequency error of the first and second signals when two signals are detected and there are several large errors. In Table 14.3, 130 times when the first signal is missing, one signal is still detected. In these 130 detections, one frequency has a large error compared with the second signal. Figure 14.13(c) shows the frequency error when the first signal is missed and one signal is detected and one large error is shown.

Figures 14.13(d) through 14.13(f) show simulation results obtained from using two different Gold codes. In Table 14.4, run number 3, there are two frequency errors. The two errors are only slightly larger than 10 MHz, as shown in Figure 14.13(e). Strictly speaking, these two errors can be considered as correct frequency if the threshold is increased slightly from 10 MHz. This phenomenon occurs in other runs. Figure 14.13(f) shows the frequency error when one signal is detected and no large error is shown.

Basically, for two frames of data using different Gold codes the frequency reading does not have large errors in 10,000 runs. The only problem with the two-frames case is missing signals. The probability of missing one signal or both signals is greater than 1%. The chance of missing a signal by using the two-frame approach is about 1.25%.

14.9 Two Signals with Different Amplitudes without Noise

In Section 14.8, the two input signals used in the simulation are of the same amplitude. In this section, two signals of different amplitudes will be studied.

Since the two frames with different Gold codes produce better results than two frames with the same Gold code, only the case of different codes will be studied. In addition, the frequency measured by the two-frames approach is always correct with 10,000 runs if we allow a frequency error only slightly over 10 MHz, say 11 MHz. The only parameters measured are the number of signals detected (or missing).

In this study, there is no noise added to the input signal. The input frequencies are randomly selected and their minimum difference is 30 MHz. The instantaneous dynamic range is amplitude dependent. When two signals with the same amplitude are considered, 20 MHz is used as the minimum frequency separation. In this case the minimum frequency separation is arbitrarily increased to 30 MHz. The amplitudes of the two signals change in 2-dB steps and 1000 runs will be performed at each power level. The amplitude difference changes from 0 to 100 dB. The number of missing one and two signals will be recorded at each amplitude difference.

Figure 14.14 shows the results of missing one and two signals versus the amplitude differences. Figure 14.14(a) shows the number of one signals missing. From 0 to 92 dB amplitude differences, the number of missing one signals is between 1 and 17 in 1000 runs. When the amplitude difference is larger than 94 dB, the number of missing one signals increases. This might be caused by the spurious signal generated by the strong signal to shadow the weak one. From 0 to 92 dB amplitude difference, the average number of missing one signals is 6.9, which is about 0.7%.

Figure 14.14(b) shows the number of two signals missing. When the amplitude difference is less than 6 dB, the receiver may miss both signals. When the amplitude difference is greater than 6 dB, the receiver always receives at least one signal. These results agree with those shown in Figure 14.11. From this simple study, it appears that the receiver can receive at least one signal when the power of the second signal is less than 5 dB below the first one.

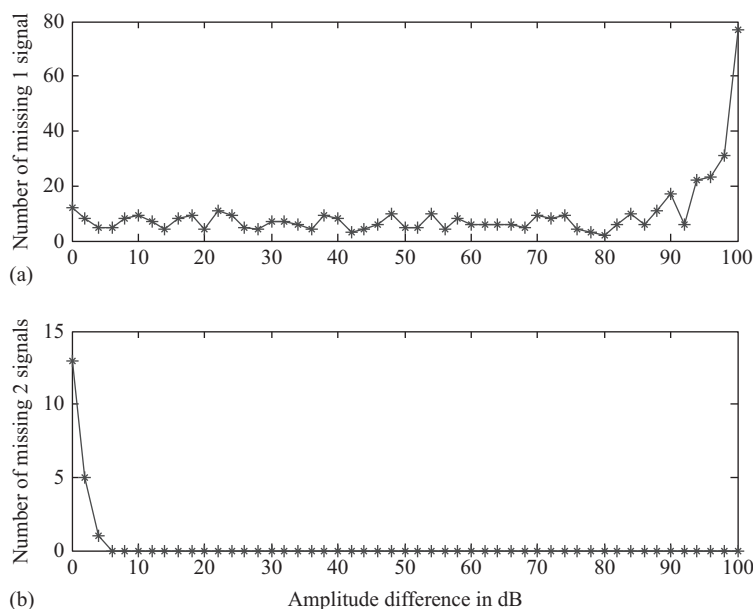


Figure 14.14 Number of missing signals versus two-signal amplitude difference among a total run of 1000: (a) number of missing one signals and (b) number of missing two signals.

However, the probability of missing the second signal increases when amplitude differences between two signals increase further. Since there is no noise added in this study, it can be considered as the limiting case.

14.10 Issues of Generating Thresholds for the Two-Frame Approach

Since no noise is added in the study presented in Section 14.9, it does not represent a realistic situation. In this section, noise is taken into account. First, a threshold must be determined. Intuitively the threshold can be determined from the threshold setting method for the FFT method, which uses noise as the input to generate the threshold. The threshold determines the probability of false alarm rate. The present detection scheme uses two frames of data; the false alarm will be generated from this approach. The second signal is obtained by subtracting the first signal. In order to perform this operation, the highest noise spike is considered as an input signal to subtract the first signal. Since the input is only noise, this operation will reduce the noise in finding the threshold of the second signal. As a result, the threshold for the second signal obtained through this approach will be lower. This conventional threshold setting method is demonstrated in this section.

Ten thousand runs with random noise as input are conducted for the threshold study. In each run, 460 outputs are generated. The distributions are shown in Figure 14.15. Figure 14.15(a) shows the outputs for the first signal and Figure 14.15(b) shows the outputs

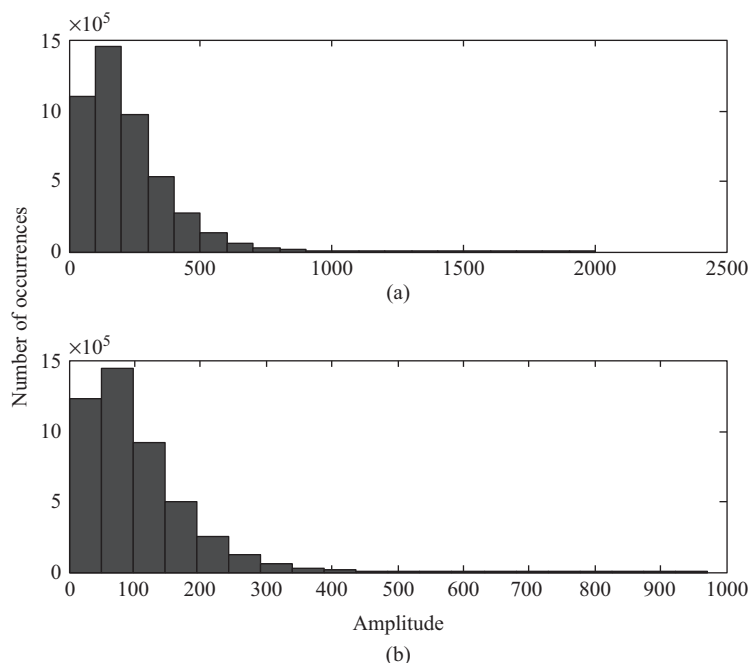


Figure 14.15 Outputs distribution from noise input: (a) for the first signal and (b) for the second signal.

for the second signal. It is difficult to fit these plots with known distributions. The thresholds are chosen from the maximum of the 460 outputs. There are 10,000 runs to generate the same number of maxima and the results are shown in Figure 14.16. Figure 14.16(a) shows the results for the first signal and Figure 14.16(b) shows the results for the second signal. The tenth highest point is chosen as the threshold since it has less variation as compared with the highest point. The tenth point theoretically produces a probability of false alarm rate of 10^{-3} . The two values obtained are 1726 and 837 for the first and second signals, respectively.

However, the threshold for the second signal determined with this conventional approach may not be correct. Theoretically the threshold for the second signal should be equal to the first signal. The argument is as follows: When there are signals in the input data, the subtracting operation will remove the signal rather than the noise. The noise should be the same as the first signal. If one signal is added in this process, the threshold should be generated by subtracting the signal and obtaining the correct result. However, the strength of the first signal and the efficiency of the signal-removing process are factors that may affect the result.

Using noise as both input signals, the number of false detections is evaluated. For 10,000 runs, the number of first and second detections is 10 and 4, respectively, which are close to the design goal of 10. With only one input signal at a moderate strength of 10 dB above noise, the first signal crosses the first threshold 10,000 times and the frequency error readings are within ± 5 MHz. The second signal crosses the second threshold 1839 times and all these are false alarms. This simple operation confirms that the first signal is not totally removed from the input. When the input signal is increased to 60 dB above noise, the first signal is detected

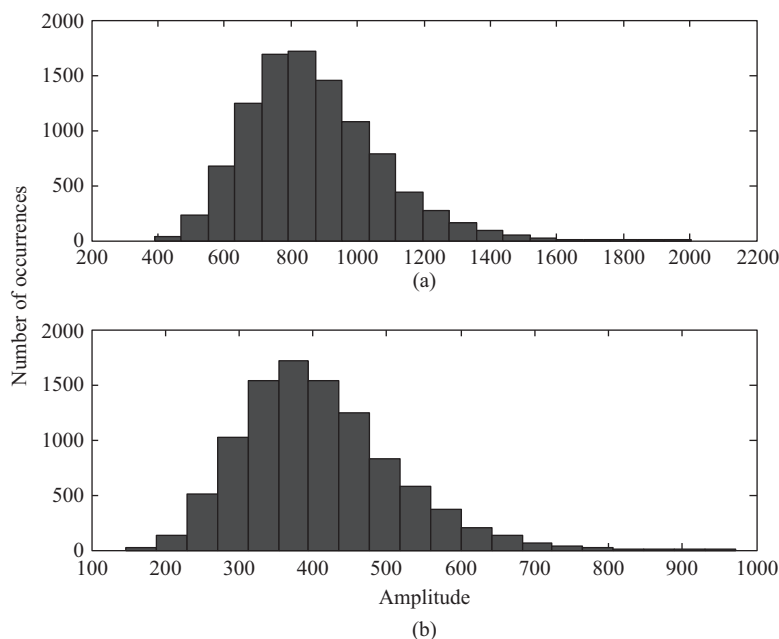


Figure 14.16 Maximum outputs distribution from noise input: (a) for the first signal and (b) for the second signal.

10,000 times with frequency error readings within ± 5 MHz. The false detection of the second signal is 1925, which is very close to the previous case of 1839. A systematic study will be used to find the probability of detection of the first signal and the probability of false detection of the second signal. The second threshold can be obtained from the above approach through evaluation of false detections. If one signal is in the input data, the first signal will be detected, but the second signal should not be detected. Any second signal detected will be a false detection.

14.11 Sensitivity and False Detection of a Second Detection of the Two-Frames Approach

The threshold for the second signal is too low, as discussed in the Section 14.10. The sensitivity and the false detection will be determined in order to obtain a better threshold for the second signal. In this study, noise is added and the two-frame approach is used. First, the frequency reading will be tested. Two strong signals of equal amplitude, 60 dB above the noise, with random input frequencies and initial phase angles are used for the test. The test contains 10,000 runs. The error frequencies are shown in Figure 14.17. Figure 14.17(a) shows the frequency errors of the first signal, and they are within ± 5 MHz. Figure 14.17(b) shows the frequency errors of the second signal, and their values are between 6 and -4 MHz. However, in one specific run of 10,000 there is a large frequency error in the first frequency measured. This result indicates that the frequency measurement can still have error when two signals are of equal amplitude.

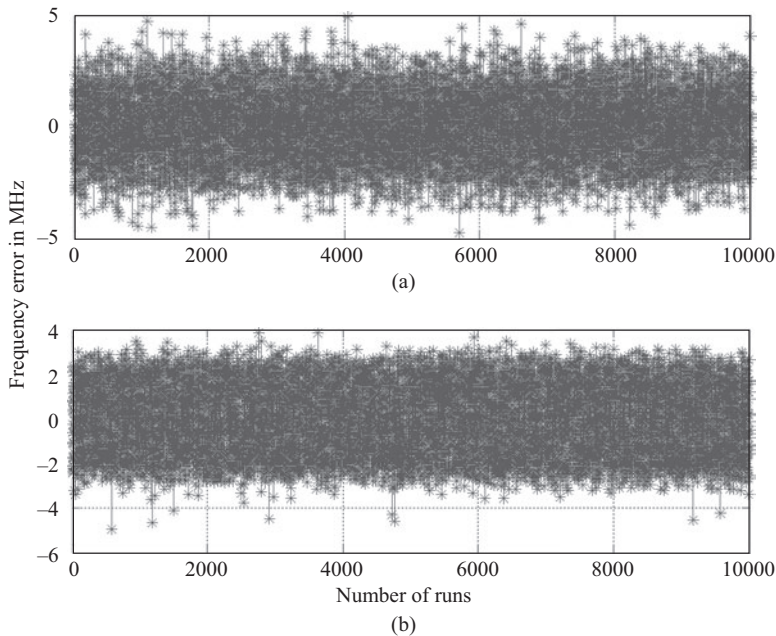


Figure 14.17 Frequency errors measured at two signals of the same amplitude with $S/N = 60$ dB: (a) first frequency and (b) second frequency.

In determining the sensitivity, only one input signal with random input frequency and initial phase but at a fixed input S/N is used. The sensitivity is determined by threshold crossing only. One thousand runs are performed at each S/N value, which is changed in 1-dB steps. Two signal outputs will be measured. Since there is only one input signal, the second signal measured will be considered a false alarm.

Figure 14.18 shows the probability of detection. The input S/N changes from -9 to 2 dB. At about -1.5 dB the first signal is detected about 900 times and the sensitivity can be considered as -1.5 dB. Compared with the sensitivity of 1.2 dB obtained from one frame of data, there is an improvement of 2.7 dB ($1.2 + 1.5$), which is the expected value when adding two frames of data to perform noncoherent detection.

Figure 14.19 shows the number of second signal detections. In order to cover a large input range for the first signal, the input S/N changes from -10 to 60 dB in 2-dB steps. When the input signal is weak, the number of false detections is small. However, when the input signal reaches about 0 dB, the number of false detections stays at a somewhat constant number close to 200. The explanation for this phenomenon is that in order to detect the second signal, the first signal is removed first. Only the remaining first signal affects the detection of the second signal or increases the false detections. These results show that the remaining signal is close to a constant when the S/N ratio is in the range of 0 to 60 dB. The explanation for this phenomenon is that when the first signal is removed, there is some residue signal left. The magnitude of the residue signal is somewhat a constant value, which causes the false detections to stay at a constant level. It is obvious from these results that the second threshold obtained should depend on the amplitude of the first signal.

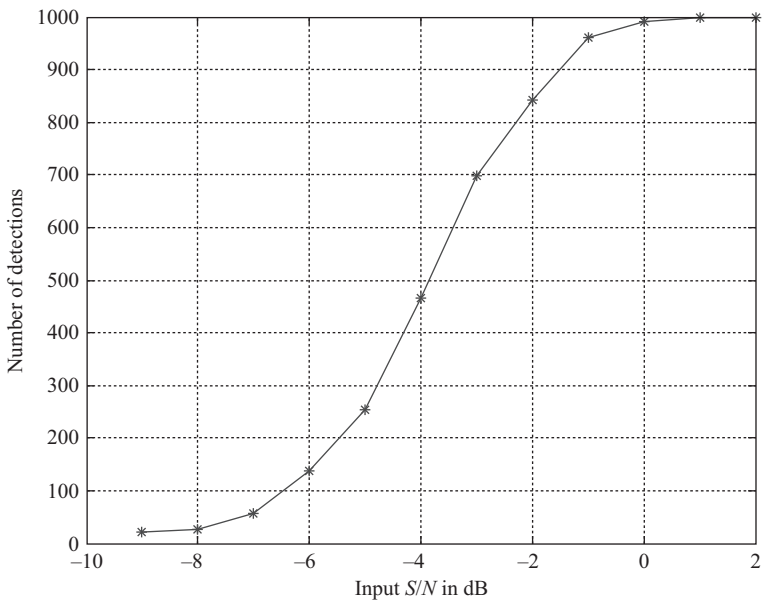


Figure 14.18 Number of detections on the first signal (x -axis: S/N ratio; y -axis: number of detections).

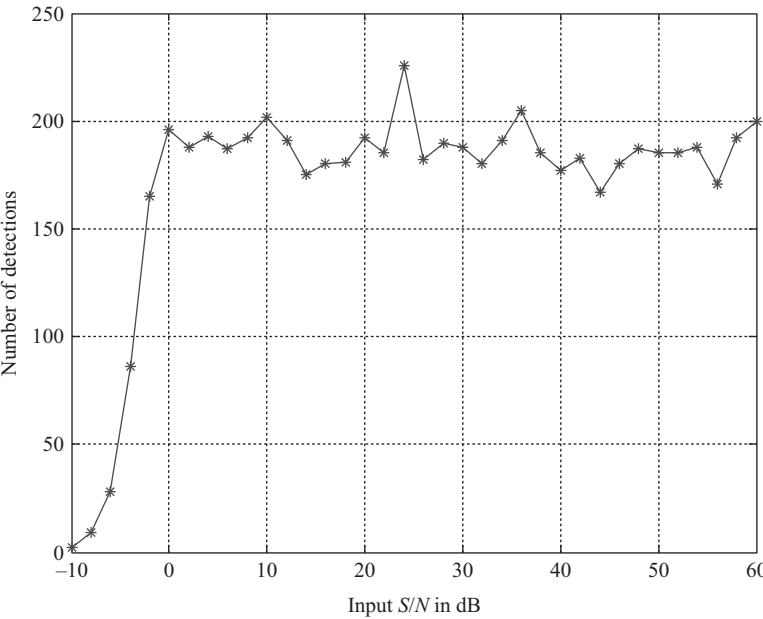


Figure 14.19 Number of false alarms on noise (second signal).

14.12 Second Threshold Obtained from the First Input Signal

From the previous two sections we can conclude that the second threshold should depend on the amplitude of the first signal. If a constant value is used, either the sensitivity is low when the first signal is weak or the false alarm rate is high when the first signal is strong. Therefore the second threshold is obtained through the following approach. One signal with random frequency and initial phase is at the input. The SNR of the input signal changes from -10 to 60 dB in 2 -dB step. At each input power level, $10,000$ runs are performed. At each run the highest amplitude is located in the frequency domain to determine the frequency of the first signal. The content of this signal is then subtracted from the time series using the prolate spheroidal sequences (MATLAB® ‘dpss’ command) [10]. The procedure to locate the signal is repeated to find the second signal from the resultant time series in which the first signal has been removed. The data on the maximum frequency output on the first signal are used to generate the amplitude of the first signal, although there are large amplitude variations, as discussed in Section 14.5. The value is obtained from the average of $10,000$ runs. The second threshold is determined as shown in Section 14.9. Among the $10,000$ outputs, the tenth largest value is used as the threshold, which produces a probability of false alarm of about 10^{-3} .

The results are shown in Figure 14.20. The x -axis presents the averaged maximum outputs for the first signal (in decibels) which is not the input signal SNR and the y -axis presents the threshold of the second signal. This figure is similar to Figure 14.19, and this is the desired result. The “*” symbol is the calculated data and the line with the “o” symbol is the approximated result. The variation appears in a range from about 29 to 40 dB. Beyond 40 dB, the value is somewhat flat; however, the data points have lots of variations beyond 35 dB, and it is difficult to obtain a good approximation.

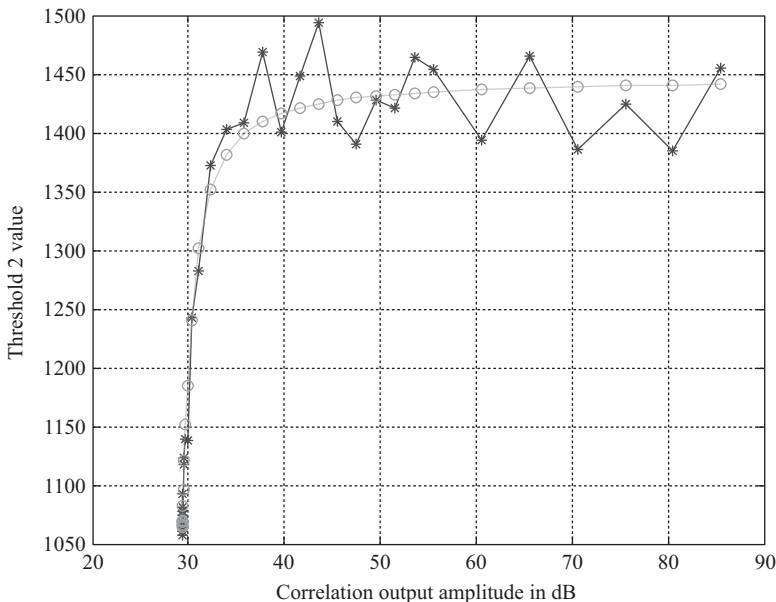


Figure 14.20 Threshold versus input correlation outputs.

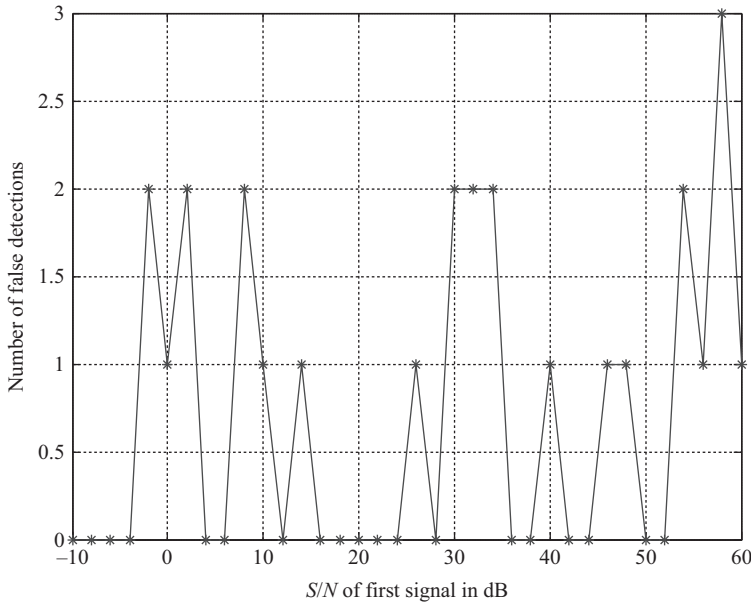


Figure 14.21 Detection of second signals (false alarm).

The approximation is obtained through the following equation determined by Dr. Liyeh Liou [1]:

$$thr2 = 1446.4 - 3.79 \left(\exp \left(\frac{78}{A - 26.1} \right) - 1 \right), \quad (14.10)$$

where $A = 10 \log(P)$ and P is the maximum frequency output in measuring the first signal. In (14.10), the threshold of the second signal depends on the amplitude of the measured first signal.

The next step is to use this approach to measure the false alarm of the second signal. Using only one input signal, we make 1000 runs at a fixed input power level with random frequency and initial phase. The power level changes from -10 to 60 dB above noise. The total outputs crossing the thresholds of the first and second signals are recorded. The number of first signals detected is not shown, but is close to the desired result. The results of the second detection are shown in Figure 14.21. The number of detections is 0, 1, 2, or 3 in 1000 runs, which is close to the design false alarm rate of 10^{-3} (1 in 1000).

14.13 Sensitivity of Second Signal Detection and Instantaneous Dynamic Range

The sensitivity of the first signal is determined in Section 14.10. The sensitivity of the second signal is determined through the following approach. Two signals of the same amplitude with random frequencies and initial phases are used as the input. The two frequencies are separated by a minimum of 30 MHz. Both signals are starting at $S/N = -10$ dB and increase in 1-dB steps until they reach 5 dB. At each S/N level, 1000 runs are made. If the output

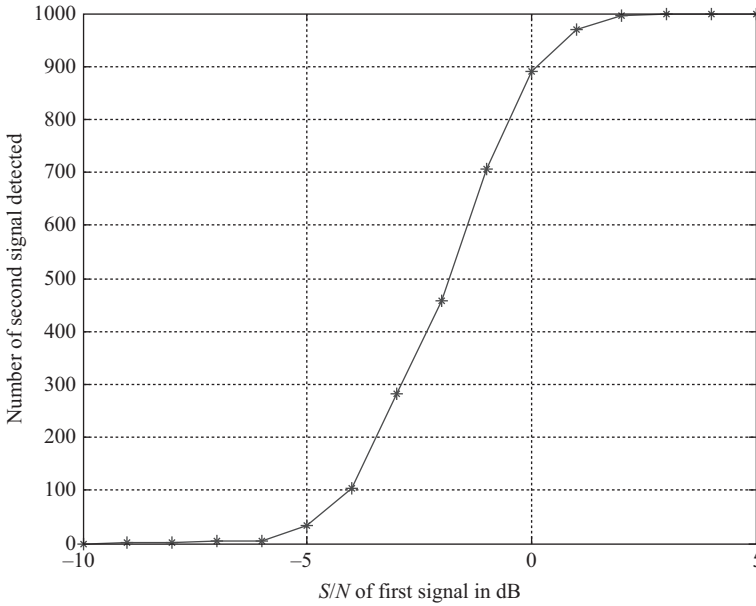


Figure 14.22 Probability of detection on the second signal with two signals of the same amplitude.

amplitude crosses the threshold, which is set based on the first signal amplitude, the second signal is detected. The results are shown in Figure 14.22. If 90% probability of detection is considered as the sensitivity level, the sensitivity is close to 0 dB, which is lower than the sensitivity of the first signal of -1.5 dB. This is close to the expected results, because when the amplitude of the first signal increases, the second threshold also increases.

Finally, the instantaneous dynamic range will be studied. The instantaneous dynamic range can be found through the following approach. The first signal (a strong one) is set at 60 dB above noise. This value is commonly used in the instantaneous dynamic range study. For this study the sampling speed is close to 400 MHz (384.62 MHz) and the equivalent bandwidth is about 200 MHz. The actual input bandwidth is about 2500 MHz and the equivalent noise floor is about -80 dBm, which can be found through

$$Nf = -114 + 10 \log(200) + \log(2500/200) \approx -80 \text{ dBm}. \quad (14.11)$$

Considering that an S/N ratio of 14 dB is required to detect a weak signal, then the weak signal must be above -66 dBm. A strong signal at 60 dB above noise (or -20 dBm) is considered a reasonable selection.

The strong signal stays 60 dB above noise and its frequency and initial phase are randomly chosen. The frequency of the second signal is selected as 20 MHz below the first signal with random initial phase. If the second frequency is below 100 MHz, the lower limit of the receiver's input band, a frequency of 20 MHz will be added to the first signal. Thus the frequency separation of the two signals is kept at 20 MHz. The amplitude of the second signal changes from a S/N ratio of -3 dB to 6 dB in 1-dB steps. At each step, 1000 runs are performed and the number of second signal detections is recorded. The number of second

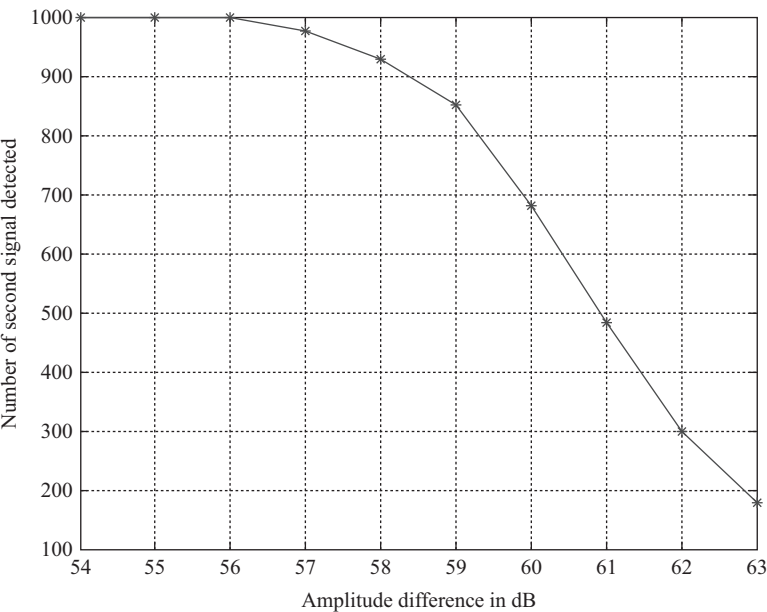


Figure 14.23 Number of second signal detections with a first signal at 60 dB above noise and a frequency separation of 20 MHz.

Table 14.5 Frequency separation versus instantaneous dynamic range.

Frequency separation (MHz)	Instantaneous dynamic range (dB)
15	53.5
20	58.4
40	60
80	60.4

signal detections is plotted against the amplitude difference (in decibels), which is considered the instantaneous dynamic range. The result is shown in Figure 14.23. If 90% of detection is used as the threshold, the instantaneous dynamic range is about 58.4 dB.

The frequency separation is then changed to 15, 20, 40, and 80 MHz. The results are listed in the Table 14.5. As expected, when the frequency separation decreases, the instantaneous dynamic range also decreases. If the frequency separation is beyond a certain value, the instantaneous dynamic range stays at a constant value. The results in Table 14.5 have these properties.

14.14 Conclusion

The preliminary study presented in this chapter indicates that the A-to-I approach using analog biphas coded signal to modulate the input signal can reduce the receiver’s sampling

frequency requirement, and the input bandwidth is determined by the frequency of the analog signal. This approach can increase the input bandwidth, and the frequency measured on one signal can provide the correct frequency over the entire input bandwidth. The sensitivity measured agrees with the expected value. The two-signal instantaneous dynamic range provides a very respectable number.

It appears that the only problem is on two signals with the same amplitude. If only one frame of data is used to find the frequency, the percentage of erroneous frequency reports is very high. If two frames of data are used to find the frequency and the analog codes of the two frames are different, the error can reduce to less than 1 in 10,000 runs, as shown in Section 14.8. This might be acceptable for EW receiver application. Further study to reduce erroneous frequency readings under two-signal conditions should be performed.

Using the same concept, different arrangements can be evaluated. For example, in the specific situation considered in this chapter, the Nyquist input bandwidth is 2500 MHz, the base bandwidth is about 192.3 MHz, and the ratio is 13 times. Different ratios should be studied by taking the desired input bandwidth and the latest ADC technology into account. One big bottleneck in this approach is the algorithm for finding the second signal. In order to find the second signal, the first signal must be removed and the same frequency estimation method is then applied to find the second signal. If a third signal is of interest, the same approach repeats again. The procedure is tedious. If the signal reconstruction can be applied, all the signals can be found through the FFT operation.

Some practical problems need to be further investigated, such as the detection of chirp and BPSK signals and hardware implementation. The specific A-to-I approach considered in this chapter is in its infancy, but appears promising for EW receiver designs. Different concepts of A-to-I approaches should be considered, especially the random sampling approach. Many areas of research are needed to determine the applications of A-to-I in EW receivers.

References

- [1] Liou L. Private communication. Dr. Liou monitored DARPA's A-to-I receiver contracts and was also personally involved in the study.
- [2] Baraniuk RG. 'Compressive sensing' [Lecture Notes]. *IEEE Signal Processing Magazine* 2007;**24**(4):118–121.
- [3] Candes EJ, Wakin MB. 'An introduction to compressive sampling'. *IEEE Signal Processing Magazine* 2008;**25**(2):21–30.
- [4] Mishali M, Eldar YC. 'Blind multiband signal reconstruction: compressed sensing for analog signals'. *IEEE Transactions on Signal Processing* 2009;**57**(3):993–1009.
- [5] Wakin M, Becker S, Nakamura E, Grant M, Sovero E, Ching D, Yoo J, Romberg J, Emami-Neyestanak A, Candes E. 'A nonuniform sampler for wideband spectrally-sparse environments'. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 2012;**2**(3):516–529.
- [6] Becker SR. *Practical Compressed Sensing: Modern Data Acquisition and Signal Processing*. Ph.D. dissertation, California Institute of Technology, 2011.
- [7] Gold R. 'Optimal binary sequences for spread spectrum multiplexing'. *IEEE Transactions on Information Theory* 1967;**13**(4):619–621.

- [8] Yoo J, Turnes C, Nakamura EB, Le CK, Becker S, Sovero EA, Wakin MB, Grant MC, Romberg J, Emami-Neyestanak A, Candes E. 'A compressed sensing parameter extraction platform for radar pulse signal acquisition'. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 2012;**2**(3):626–638.
- [9] Tsui J. *Microwave Receivers with Electronic Warfare Applications*. New York: John Wiley & Sons; 1986:23.
- [10] MathWorks. *Discrete Prolate Spheroidal (Slepian) Sequences: dpss*. Available from <http://www.mathworks.com/help/signal/ref/dpss.html>.

Angle of Arrival Measurements

15.1 Introduction [1–6]

As mentioned in Chapter 2, the angle of arrival (AOA) is the most valuable piece of information that can be obtained from an enemy radar because the radar cannot change its position drastically in a very short time frame (i.e., a few milliseconds). Unfortunately AOA information is also the most difficult information to obtain. It requires several antennas with receivers. The two common approaches to measure AOA are based on amplitude and phase comparisons. Another approach is to use the Doppler frequency shift generated by the aircraft movement. However, this approach is closely related to the phase measurement system. If the requirement is to measure AOA on simultaneous signals, the problem becomes even more complicated because receivers with multiple signal capability are needed.

In an amplitude comparison system, the amplitude of all the receivers must be matched from the antennas through the outputs of the receivers. This approach usually generates an AOA resolution of ± 15 deg, which is less than the desired value of ± 1 deg required in modern electronic warfare (EW) applications. In a phase measurement system, the phase of all the receivers must be matched from the antennas through the outputs of the receivers, which is a very difficult task in receiver design. The phase comparison system usually can generate AOA resolution of ± 1 deg, which satisfies the modern EW requirement. The accuracy obtained from an amplitude comparison system is not theoretically limited. In most of the amplitude comparison systems, the AOA coverage is rather wide, thus the antenna beams are wide, which results in poor AOA accuracy. If antennas have narrow beams (e.g., in monopulse radar), highly accurate AOAs can be obtained.

Theoretically, if the antennas/receivers in a system are not properly balanced, a calibration table can be used to minimize the difference. If a system is far from matched, many calibration points are needed, which is often difficult to implement. In the practical design of an AOA measurement system, the primary effort is usually placed on balancing the antennas/receivers among different channels. A calibration table is used only to remedy the remaining hardware mismatch.

If a digital approach is used to measure the AOA, the data generated will be two-dimensional—in time and in space. Obviously the processing will be more complicated than a one-dimensional case in time. To collect AOA data, multiple analog-to-digital converters (ADCs) must be used. In front of the ADCs, the antennas/receivers must be phase matched, as in a phase measurement system. The digitization of the ADCs should be operated in a

synchronized manner. The number of antennas that can be used in an AOA system is very low in comparison with electronically scanning antennas in a radar system. In an airborne system, the maximum number of antennas might be 10. In a shipborne application, the number might be larger because more room is available.

Amplitude comparison AOA measurement for digital signal processing will not be discussed in this chapter because the approach is not much different from an analog amplitude comparison system. The discussion will concentrate on the phase comparison or related approaches, such as Doppler frequency measurement. Most recent digital AOA studies are concentrated on high-resolution approaches, such as the multiple signal classification (MUSIC), estimation of signal parameters via rotational invariance (ESPRIT), and minimum norm methods. These approaches are quite similar to the frequency measurements discussed in Chapter 11, thus they will not be included here. However, due to the small data size collected in the space domain, these methods should be useful for special applications.

First, a queuing concept will be discussed. This concept is not limited to digital AOA systems, but it may provide some ideas for digital receiver designs. Second, data generated from linear and circular arrays will be presented. A simple approach based on zero crossing, which can process only one signal, will be discussed, as well as a multiple signal approach using fast Fourier transform (FFT). The Chinese remainder theorem will be presented, which can be used to solve antenna location problems. Finally, a simple AOA data collection system that can be used in conjunction with a digital system will be presented.

15.2 Queuing Concept [7]

The queuing concept is probably one of the few AOA measurement approaches that can process simultaneous signals with a reasonable amount of hardware. This concept has been successfully investigated in analog receiver designs. The queuing concept is to measure some quantity in a gross manner over a wide instantaneous bandwidth. The information obtained is used to direct narrowband measurement systems to obtain the fine information on the input signals. There are many different types of queuing arrangements, one of which will be discussed here.

Figure 15.1 shows a frequency queuing system. The input signal is received by an omnidirectional antenna with a wideband receiver of coarse-frequency resolution. The measured frequency of the input signal is used to tune a set of narrowband receivers. One of the narrowband receivers is usually dedicated to measuring the fine-grain information (i.e., frequency, pulse amplitude, pulse width [PW], and time of arrival [TOA]). Other narrowband receivers can be used to measure AOA. The AOA can be measured either through amplitude or phase comparison schemes. Wideband radio frequency (RF) delay lines are used in front of the narrowband receivers to delay the input signal for the frequency tuning to settle.

Although the wideband receiver can process simultaneous signals, this arrangement can provide fine-grain information, including AOA on only one input signal. If more than one signal needs to be processed, additional hardware is required. Each set of narrowband receivers can process only one signal. The number of narrowband systems matches the number of signals to be processed. If the number is high (i.e., four simultaneous signals), the logic circuit to assign the narrowband systems to the input signals can become very complicated.

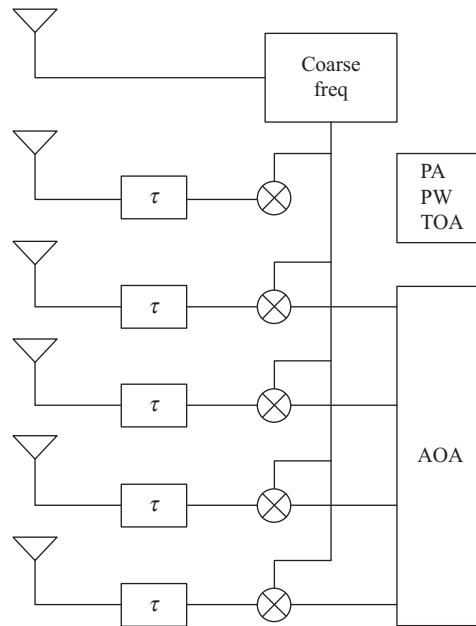


Figure 15.1 A simple queuing AOA system.

The advantage of this arrangement is that these narrowband receivers are required to process only one signal and therefore they are relatively easy to build. In addition, narrowband receivers are easier to match in amplitude and phase among different channels than a wideband system. It is important to note that this approach obtains the AOA and fine-grain information on a pulse-by-pulse basis.

The design may have some possible technical problems in that the wideband delay lines are difficult to build and the insertion loss is usually high. The amplitude and phase of the delay line may be temperature sensitive. A signal passing a long delay line has many phase variations, therefore it is difficult to match the phase among long delay lines.

A disadvantage of this approach is that the sensitivity of the system might be low. The antenna is omnidirectional, and therefore the gain is low. A receiver with coarse channel resolution contains more noise, and therefore the sensitivity is low. The initial signal detection through this low-gain antenna and low-sensitivity receiver combination produces an overall low-sensitivity system. Although high-gain antennas and high-sensitivity narrowband receivers are used to obtain fine-grain information, they are not used for the initial signal detection.

Theoretically, digital technology can be used to build the wideband or the narrowband receiver if digital receivers are used for the narrowband applications where the wide RF delay lines might not be needed to temporarily store information.

15.3 Digital Data from a Linear Antenna Array [7–9]

In this section, the data obtained from a linear antenna array is presented. These data can be used for different AOA measurement schemes. In an EW application, the antenna array

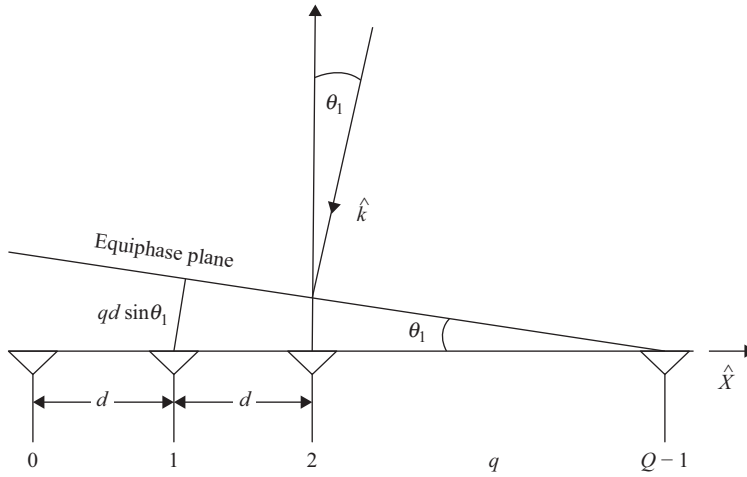


Figure 15.2 A linear array and a plane incident wave.

usually has few elements. A linear antenna array with three elements is not uncommon in practice. Usually a linear array is placed in the horizontal direction to measure the azimuth angle. One linear array can theoretically cover up to π radians or 180 deg of azimuth angle, although it is often limited to 120 deg to avoid operating in the end-fire mode. If the elevation angle is also of interest, a linear array in the vertical direction has to be added. As far as AOA measurement is concerned, the two linear arrays are usually treated separately.

Figure 15.2 shows a linear array of antennas with Q elements from $q = 0, 1, \dots, Q - 1$ along the x direction. The elements in this array are uniformly spaced and the distance is d . Let us assume that there is only one plane wave and the equiphase plane is shown. If the input signal is a sinusoidal wave, the antenna of the q th element will have an output of

$$x(q, t) = A_1 \cos[2\pi f_1(t - \tau_q)], \quad (15.1)$$

where A_1 and f_1 are the amplitude and the frequency of the input signal and τ_q is a phase delay time on the q th element with respect to the first antenna element ($q = 0$). Since the distance between the antenna element is d , this delay time can be written as

$$\tau_q = \frac{-qd \sin \theta_1}{c}, \quad (15.2)$$

where θ_1 is the incident angle of the input signal shown in Figure 15.2 and c is the speed of light. The minus sign in this equation is from the fact that the equiphase plane arrives at the q th element before it reaches the zero antenna element. From this figure, one can see that a linear array cannot distinguish a signal coming from above or below the array; thus, in order to avoid ambiguity, the angle θ_1 is usually limited from $-\pi/2$ to $\pi/2$.

Substituting the result from (15.2) into (15.1), one obtains

$$x(q, t) = A_1 \cos\left[2\pi f_1\left(t + \frac{qd \sin \theta_1}{c}\right)\right]. \quad (15.3)$$

Often this equation is written in a different form. Let $\hat{k}_1$ be a unit vector pointing in the direction of the wave propagation and $\hat{x}$ a unit vector along the direction of array. Do not confuse $\hat{x}$ with the input $x(q, t)$. Equation (15.3) can be written as

$$x(q, t) = A_1 \cos\left(2\pi f_1 t - \frac{2\pi f_1 q d \hat{k}_1 \cdot \hat{x}}{c}\right),$$

where

$$\hat{k}_1 \cdot \hat{x} = \cos\left(\frac{\pi}{2} + \theta_1\right) = -\sin \theta_1. \quad (15.4)$$

If there are two signals, the output from the q th element of the antenna array is the sum of the two signals:

$$\begin{aligned} x(q, t) = & A_1 \cos\left(2\pi f_1 t - \frac{2\pi f_1 q d \hat{k}_1 \cdot \hat{x}}{c}\right) \\ & + A_2 \cos\left(2\pi f_2 t - \frac{2\pi f_2 q d \hat{k}_2 \cdot \hat{x}}{c}\right). \end{aligned} \quad (15.5)$$

If there are M signals, the output from the q th antenna element is

$$x(q, t) = \sum_{m=1}^M A_m \cos\left(2\pi f_m t - \frac{2\pi f_m q d \hat{k}_m \cdot \hat{x}}{c}\right), \quad (15.6)$$

where f_m and $\hat{k}_m$ are the frequency and direction of the m th signal. Often the exponential form is used to express this result as

$$x(q, t) = \sum_{m=1}^M A_m e^{j\left(2\pi f_m t - \frac{2\pi f_m q d \hat{k}_m \cdot \hat{x}}{c}\right)}. \quad (15.7)$$

In order to obtain digitized data, the time t is replaced by integers as $n = 0, 1, \dots, N - 1$.

All the frequency measurement methods discussed in previous chapters can be used to find the AOA. For example, when the FFT is applied to the time domain outputs, the outputs are in the frequency domain. If FFT is applied to data in the space domain that are collected from different antennas at one specific time instance, the output will represent AOA (given knowledge of the frequency f). The major difference is that the time domain data contain many data points and the space domain data points are equal to the number of antennas in the linear array. The Fourier transform of data in the space domain will be further discussed in Section 15.8.

15.4 Outputs from a Circular Antenna Array [10–13]

This section presents the output of a circular array. A circular array is a two-dimensional arrangement, therefore it can be used to measure both the azimuth and elevation angles. These types of arrays occupy more space than the linear array and might be suitable for some special aircraft and shipboard applications. A circular antenna array is shown in Figure 15.3.

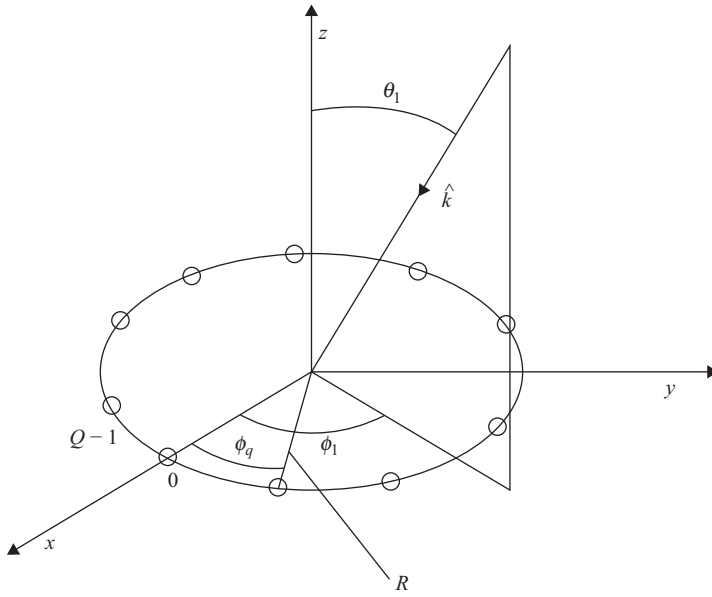


Figure 15.3 Circular antenna array.

In this figure, the radius of the circle is R and there are Q elements from $q = 0$ to $Q - 1$. The array is in the xy plane and the first antenna element is the zero element and is on the x -axis, thus ϕ is zero. The locations of the antennas are at multiples of the angle ϕ_q .

Assume that a plane wave with amplitude A_1 and frequency f_1 is incident on this array with angle ϕ_1 and θ_1 . The output induced at the q th element can be written as

$$e(q, t) = A_1 e^{j2\pi f_1 \left(t - \frac{\tau_{q1}}{c}\right)}, \quad (15.8)$$

where the delay time τ_{q1} referenced to the center of the array is

$$\tau_{q1} = -R \sin \theta_1 \cos \phi_{q1}, \quad (15.9)$$

where R is the radius of the array and θ_1 is the angle between the incident ray and the normal of the array. In this case, because the antenna locations are discrete, the angle ϕ_{q1} can be written as

$$\phi_{q1} = q\phi_q - \phi_1, \quad (15.10)$$

where $q = 0, 1, \dots, Q - 1$, ϕ_1 is the azimuth angle of the incident wave, and ϕ_q is the angle between two adjacent antennas, which is

$$\phi_q = \frac{2\pi}{Q}. \quad (15.11)$$

Substituting (15.9) through (15.11) into (15.8), the result is

$$\begin{aligned} e(q, t) &= A_1 e^{j2\pi f_1 \left(t + \frac{R \sin \theta_1 \cos \left(\frac{2\pi q}{Q} - \phi_1\right)}{c}\right)} \\ &= A_1 e^{j2\pi \left(f_1 t + \frac{R \sin \theta_1 \cos \left(\frac{2\pi q}{Q} - \phi_1\right)}{\lambda_1}\right)} \end{aligned} \quad (15.12)$$

For example, if the signal is incident from the x direction (i.e., $\theta_1 = 90^\circ$), $\phi_1 = 0$. The output from the first antenna element ($q = 0$) is

$$e(0, t) = A_1 e^{j2\pi f_1 \left(t + \frac{R}{c}\right)}. \quad (15.13)$$

This means the output is leading from the center of the circle by R/c , which is the anticipated result.

This result can be written in vector form. The incident direction of the input signal can be represented by a unit vector $\hat{k}_1$ in a Cartesian coordinate as

$$\hat{k}_1 = -(\sin \theta_1 \cos \phi_1 \hat{x} + \sin \theta_1 \sin \phi_1 \hat{y} + \cos \theta_1 \hat{z}), \quad (15.14)$$

where $\hat{x}$, $\hat{y}$, and $\hat{z}$ are unit vectors in the Cartesian system. The minus sign in front means the vector is pointing to the center of the origin. The position of the q th antenna element is in the $\hat{r}_q$ direction as

$$\hat{r}_q = \cos(q\phi_q) \hat{x} + \sin(q\phi_q) \hat{y}. \quad (15.15)$$

The dot product of $\hat{k}_1 \cdot \hat{r}_q$ is

$$\begin{aligned} \hat{k}_1 \cdot \hat{r}_q &= -[\sin \theta_1 \cos \phi_1 \cos(q\phi_q) + \sin \theta_1 \sin \phi_1 \sin(q\phi_q)] \\ &= -\sin \theta_1 \cos(q\phi_q - \phi_1). \end{aligned} \quad (15.16)$$

Thus (15.12) can be written as

$$\begin{aligned} e(q, t) &= A_1 e^{j2\pi f_1 \left(t - \frac{R \hat{k}_1 \cdot \hat{r}_q}{c}\right)} \\ &= A_1 e^{j \left(2\pi f_1 t - \frac{2\pi f_1 R \hat{k}_1 \cdot \hat{r}_q}{c}\right)}. \end{aligned} \quad (15.17)$$

Extending this equation to cover M signals,

$$\begin{aligned} e(q, t) &= \sum_{m=1}^M A_m e^{j2\pi f_m \left(t + \frac{R \sin \theta_m \cos \left(\frac{2\pi q}{Q} - \phi_m\right)}{c}\right)} \\ &= \sum_{m=1}^M A_m e^{j2\pi \left(f_m t + \frac{R \sin \theta_m \cos \left(\frac{2\pi q}{Q} - \phi_m\right)}{\lambda_m}\right)} \\ &= \sum_{m=1}^M A_m e^{j2\pi f_m \left(t - \frac{R \hat{k}_m \cdot \hat{r}_q}{c}\right)} \\ &= \sum_{m=1}^M A_m e^{j2\pi \left(f_m t - \frac{R \hat{k}_m \cdot \hat{r}_q}{\lambda_m}\right)}. \end{aligned} \quad (15.18)$$

This result can be considered as identical to (15.7).

For digitized data, t is replaced by $n = 0, 1, \dots, N - 1$ and the number of antennas is $q = 0, 1, \dots, Q - 1$. Therefore the data are two-dimensional in n and q . If signal processing is applied in the n domain (time), the result will be frequency. If signal processing is applied to q , the result is AOA.

The results from the linear array and the circular array are very much alike. In later discussions, it will be shown that both data can be used and the results will be quite similar. The linear array can only provide azimuth angle information because the corresponding $\hat{k}$ is

two-dimensional. The circular array can provide both azimuth and elevation information because the corresponding $\hat{k}$ is three-dimensional. For simplicity and airborne consideration, the linear array will be used in the following sections.

15.5 Two-Element Phase Array Antenna

The phase measurement discussed in Chapter 8 can be used to measure AOA information. In order to keep this discussion simple, a linear antenna array with only two elements will be presented, as shown in Figure 15.4. In this figure, the θ_1 is in the positive direction. It is also assumed that this is a narrowband system. In other words, there is only one input signal. With two elements, the outputs are

$$\begin{aligned} e(0, t) &= A_1 e^{j2\pi f_1 t} \equiv A_1 e^{j\psi_0(t)} \\ e(1, t) &= A_1 e^{j(2\pi f_1 t + 2\pi f_1 \frac{d \sin \theta_1}{c})} \equiv A_1 e^{j\psi_1(t)}, \end{aligned} \quad (15.19)$$

where A_1 and θ_1 are the amplitude and incident angle of the input signal and d is the distance between the two antenna elements. The phase angles are represented by $\psi_1(t)$ and $\psi_2(t)$. In this equation, continuous time is considered.

In an analog phase measurement system, $\psi_0(t)$ and $\psi_1(t)$ are the measured quantities. The difference between the two phases can be used to measure the AOA information as

$$\psi \equiv \Delta\psi_{10}(t) = \psi_1(t) - \psi_0(t) = \frac{2\pi f_1 d \sin \theta_1}{c} = \frac{2\pi d \sin \theta_1}{\lambda_1}. \quad (15.20)$$

Two interesting points are noted in this equation. First, ψ is independent of time t . Second, there are two unknowns, f_1 and θ_1 .

Since ψ is independent of time t , it needs only one measurement at any time instant to obtain the AOA. This is true only theoretically; however, due to the noise contamination, the result obtained from one or few data samples may be inaccurate. In general, many samples are needed to generate the AOA information.

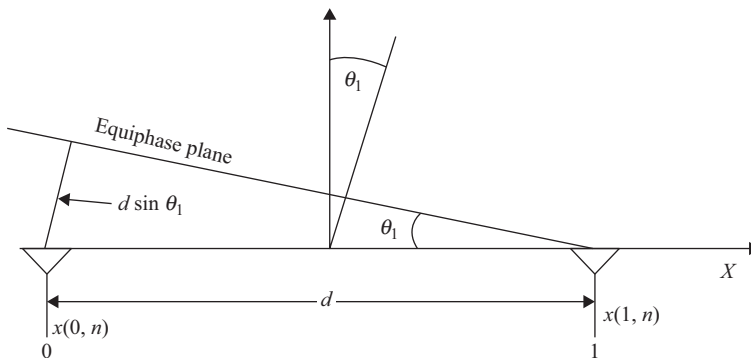


Figure 15.4 A two-antenna element AOA system.

Since there are two unknowns, the frequency f_1 must be found first. This is always true in a phase comparison system where the frequency of the input signal must be measured first. This frequency can be measured by using the data from one antenna element (say, the 0 element). Since this is a narrowband system containing only one input signal, the frequency can be found from the phase comparison system discussed in Chapter 8 as

$$f_1 = \frac{\psi_0(n) - \psi_0(0)}{nt_s} \equiv \frac{\Delta\psi_0(n)}{nt_s}. \quad (15.21)$$

In order to avoid frequency ambiguity, the phase angle $\Delta\psi_0(n)$ must be less than 2π . As discussed before, a small n value that represents a short delay time is used to cover a wide-frequency bandwidth and a large value of n that represents a long delay time is used to generate fine-frequency resolution.

Once the frequency of the input signal is obtained, the AOA information can be obtained from (15.20) as

$$\sin \theta_1 = \frac{c\psi}{2\pi f_1 d} = \frac{\lambda_1 \psi}{2\pi d}. \quad (15.22)$$

As discussed before, the incident angle is limited to $-\pi/2$ to $\pi/2$ and the corresponding difference in $\sin \theta_1$ is $\sin(\pi/2) - \sin(-\pi/2) = 2$. Since the maximum of ψ is 2π , then from the above equation, the maximum separation between the two antenna elements is

$$2 = \frac{\lambda_1 \psi_{\max}}{2\pi d_{\max}} = \frac{2\pi \lambda_1}{2\pi d_{\max}}$$

or

$$d_{\max} = \frac{\lambda_1}{2}. \quad (15.23)$$

This is the well-known relation in linear antenna arrays that the maximum separation between two elements must be less than or equal to $\lambda/2$.

Let us summarize the approach discussed above.

1. In a phase measurement AOA system, the maximum separation between the two antenna elements is $\lambda/2$, where λ is the wavelength of the input signal.
2. From the time domain output of any one of the antenna elements, the frequency can be obtained by measuring the phase difference as a function of time. This is discussed in detail in Chapter 8. The input frequency (or wavelength) is needed to calculate the AOA.
3. From the phase difference between the two antennas, the AOA can be measured through (15.22). Theoretically only one pair of phase measurements can determine the AOA; however, the AOA accuracy is rather poor.

In order to improve the measurement accuracy, the phase measurement can be obtained as

$$\psi = \Delta\psi_{10} = \frac{1}{N} \sum_{n=0}^{N-1} [\psi_1(n) - \psi_0(n)]. \quad (15.24)$$

This is taking the average of the phase difference between the two antennas at many time intervals.

15.6 AOA Measurement Through Zero Crossing

Zero crossing, discussed in Chapter 8, to measure the frequency of an input signal can also be used to measure the AOA information. Similar to the phase measurement system, this approach can process only one input signal. If more than one input signal arrives at the antenna/receivers, the measured AOA could be erroneous. As mentioned in Chapter 8, the zero crossing approach requires only one receiver with one ADC per antenna and an in-phase–quadrature (*I-Q*) channel arrangement is not required. The detailed approach to improve the zero crossing measurement is presented in Chapter 8 and will not be repeated here. Only the concept will be discussed.

A two-element zero crossing AOA system is shown in Figure 15.5. The positions of the two antennas are represented by two circles and the incident angle is θ_1 . The two sinusoidal waves represent the input signal. The first zero crossing occurs at the left antenna at t_0 . At the time instant t_0 , the zero crossing is $d \sin \theta_1$ from the second antenna. Therefore the difference of zero crossing time between the two antennas can be written as

$$\Delta t = \frac{d \sin \theta_1}{c}$$

or

$$\sin \theta_1 = \frac{c \Delta t}{d}. \quad (15.25)$$

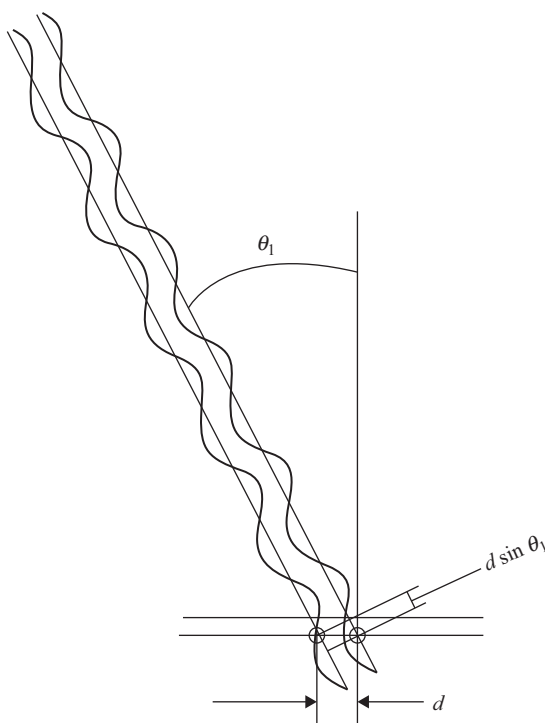


Figure 15.5 Zero crossing of a two-antenna system.

It is interesting to note that in this measurement, the AOA information obtained is independent of the signal frequency because the time difference rather than the phase is measured.

In this arrangement, the two antennas cannot be too far apart, otherwise the system will have ambiguity. The maximum separation can be determined when the incident angle $\theta_1 = 90^\circ$. Since there are two zero crossings per cycle, the maximum distance between two zero crossings is $\lambda/2$. The maximum separation between the two antennas is $\lambda/4$ to avoid comparing the zero crossings between two different zero crossing points.

However, if one can track the sign change before and after the zero crossing, the length between two ambiguous zero crossings is λ . Under this condition, the maximum separation between the two antenna elements is $\lambda/2$ to avoid comparing the zero crossings between two different zero crossing points.

Although this discussion is based on the zero crossing of one data point, in actual measurement many zero crossings should be averaged to reduce the noise effect.

15.7 Phase Measurement in AOA Systems with Multiple Antennas [14, 15]

In the previous section, the accuracy of the AOA measured by a two-antenna system is usually quite limited. In (15.22), a large value of d means a smaller value of θ_1 . Physically this means a longer baseline d can provide more accurate AOA information at the cost of narrower unambiguous incident angle coverage. For a two-antenna linear array, the maximum distance must be less than $\lambda_1/2$ to avoid AOA ambiguity. If d is longer than this value, the measured angle ψ will be greater than 2π and the angle θ_1 will be ambiguous.

In order to avoid ambiguity and provide accurate AOA information, a linear array with multiple antennas can fulfill this need. Figure 15.6 shows a four-element system. In this figure, the shortest distance between the two antennas is d , which is less than $\lambda_1/2$. The other two antenna elements are at $4d$ and $16d$ from the first element. The input angle coverage will be less than π ($-\pi/2 < \theta_1 < \pi/2$).

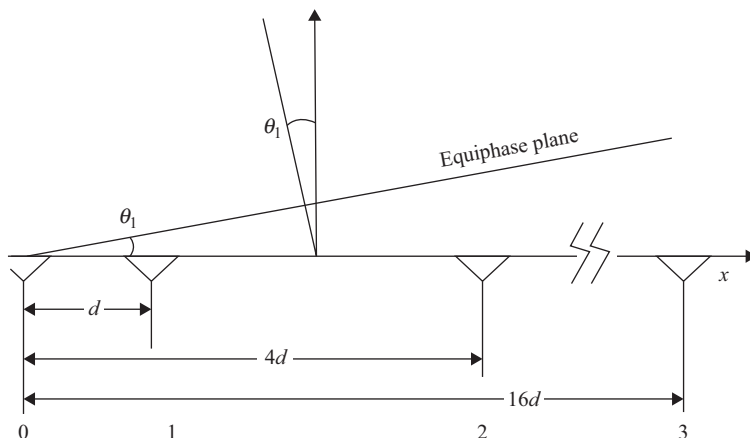


Figure 15.6 A four-element linear antenna array.

For simplicity, let us assume that the angle coverage is equal to π . The antennas with the shortest distance will provide the coarse AOA information and the resolution must be finer than $\pi/4$ (such as $\pi/8$). The second antenna pair has an ambiguous AOA range of $\pi/4$, since the distance between them is $4d$. This pair should provide AOA resolution of at least $\pi/16$ because the longest antenna pair has an ambiguous AOA range of $\pi/16$. The longest antenna pair will provide the final AOA resolution.

This same idea can be applied to zero crossing measurement. Antennas with short separation can be used to resolve ambiguity and antennas with long separation provide better AOA resolution.

15.8 Fourier Transform Over the Space Domain [8, 9, 15]

In this section, the data are obtained from an antenna array at a certain time instant. Fourier transform will be performed on the data to obtain the AOA information. Let us assume that the linear antenna array has uniform spacing and there is only one input signal. Also assume that the data collected are at $t = 0$. This assumption simplifies the calculation but does not lose generality. Under this condition, from (15.7), the input signal is

$$x(q) = A_1 e^{\frac{-j2\pi f_1 q d \hat{k}_1 \cdot \hat{x}}{c}}, \quad (15.26)$$

where A_1 , f_1 , and $\hat{k}_1$ are the amplitude, frequency, and incident direction of the incoming signal, respectively, q is the number of antenna elements, and d is the distance between two adjacent antennas.

If there are Q antenna elements, the Fourier transform in the space domain can be written as

$$\begin{aligned} X(\hat{k}) &= \sum_{q=0}^{Q-1} x(q) e^{\frac{-j2\pi f q d \sin\theta}{c}} = \sum_{q=0}^{Q-1} x(q) e^{\frac{j2\pi f q d \hat{k} \cdot \hat{x}}{c}} \\ &= \sum_{q=0}^{Q-1} A_1 e^{\frac{j2\pi f q d (\hat{k} - \hat{k}_1) \cdot \hat{x}}{c}} = A_1 \frac{1 - e^{\frac{j2\pi f Q d (\hat{k} - \hat{k}_1) \cdot \hat{x}}{c}}}{1 - e^{\frac{j2\pi f d (\hat{k} - \hat{k}_1) \cdot \hat{x}}{c}}}. \end{aligned} \quad (15.27)$$

The amplitude of this function can be written as

$$|X(\hat{k})| = A_1 \frac{\left| 1 - e^{\frac{j2\pi f Q d (\hat{k} - \hat{k}_1) \cdot \hat{x}}{c}} \right|}{\left| 1 - e^{\frac{j2\pi f d (\hat{k} - \hat{k}_1) \cdot \hat{x}}{c}} \right|} = A_1 \frac{\left| \sin\left(\frac{2\pi f Q d (\hat{k} - \hat{k}_1) \cdot \hat{x}}{2c}\right) \right|}{\left| \sin\left(\frac{2\pi f d (\hat{k} - \hat{k}_1) \cdot \hat{x}}{2c}\right) \right|}. \quad (15.28)$$

This is the well-known response of a linear array [1, 2]. The peak of $|X(\hat{k})|$ occurs at

$$\hat{k} = \hat{k}_1$$

or

$$\Delta\theta = 0 \quad (15.29)$$

because $(\hat{k} - \hat{k}_1) \cdot \hat{x} = -\sin(\Delta\theta)$. The peak is in the direction of the incident wave, and this is the expected result.

One can eliminate the frequency dependence in $X(\hat{k})$ by assuming that d and λ are linearly related (i.e., $d = \lambda/2$). Under this specialized condition, the above equation can be simplified as

$$|X(\hat{k})| = A_1 \frac{\left| \sin\left(\frac{\pi Q(\hat{k} - \hat{k}_1) \cdot \hat{x}}{2}\right) \right|}{\left| \sin\left(\frac{\pi(\hat{k} - \hat{k}_1) \cdot \hat{x}}{2}\right) \right|} = A_1 \frac{\left| \sin\left(\frac{\pi Q \sin(\Delta\theta)}{2}\right) \right|}{\left| \sin\left(\frac{\pi \sin(\Delta\theta)}{2}\right) \right|}. \quad (15.30)$$

It is easy to see that $X(\hat{k})$ has a peak at $\hat{k} = \hat{k}_1$, the same result as in (15.29). The first zero occurs at

$$\frac{\pi Q \sin(\Delta\theta)}{2} = \pi$$

or

$$\Delta\theta = \sin^{-1}\left(\frac{2}{Q}\right). \quad (15.31)$$

The beamwidth θ_{bw} can be considered as

$$\theta_{bw} = 2\Delta\theta = 2 \sin^{-1}\left(\frac{2}{Q}\right). \quad (15.32)$$

This equation indicates that a large value of Q , the number of elements in the antenna array, produces a narrow beam and a high AOA resolution. A small number of elements produce a wide beam and poor AOA resolution. Since $d = \lambda/2$, the larger the Q value, the longer the antenna. It is well known that a long linear antenna array will provide fine beamwidth.

Figure 15.7(a) shows the response of a 10-element case, while Figure 15.7(b) shows the response of a 3-element case. The amplitude is plotted in log scale and the angle coverage is π ($-\pi/2 < \Delta\theta < \pi/2$). It is clearly shown that a larger number of antenna elements provide finer AOA resolution. Due to the few terms (also the number of antenna elements) in the spatial Fourier transform, the sidelobes are very high.

If there are simultaneous signals, the high sidelobes make the instantaneous dynamic range low. The high sidelobes also make the parameter encoder designs more complicated. The levels of the sidelobes can be reduced if a proper weighting function is applied to the antenna array. One approach to add a weighting function to an antenna array is to reduce the outputs of some of the antennas. The outputs from the ends of the array have higher attenuation. Figure 15.8 shows a very simple example. The curve at the bottom represents the attenuation added, and they are symmetrical with respect to the center antenna element. The element attenuations should be $A_1 > A_2 > A_3 > A_4 = 0$, where the center element does not have any attenuation. As discussed in Chapter 3, the weighting function will increase the width of the mainlobe. In addition to this problem, if there are only a few antenna elements, the weighting approach may not be very effective anyway.

It appears that if a few antenna elements are used in a system, such as in the airborne case, it is difficult to obtain fine AOA resolution through the Fourier transform in the spatial domain. If there are multiple inputs from different angles, theoretically the Fourier transform in the space domain should generate peaks at all the incident angles. However, if the

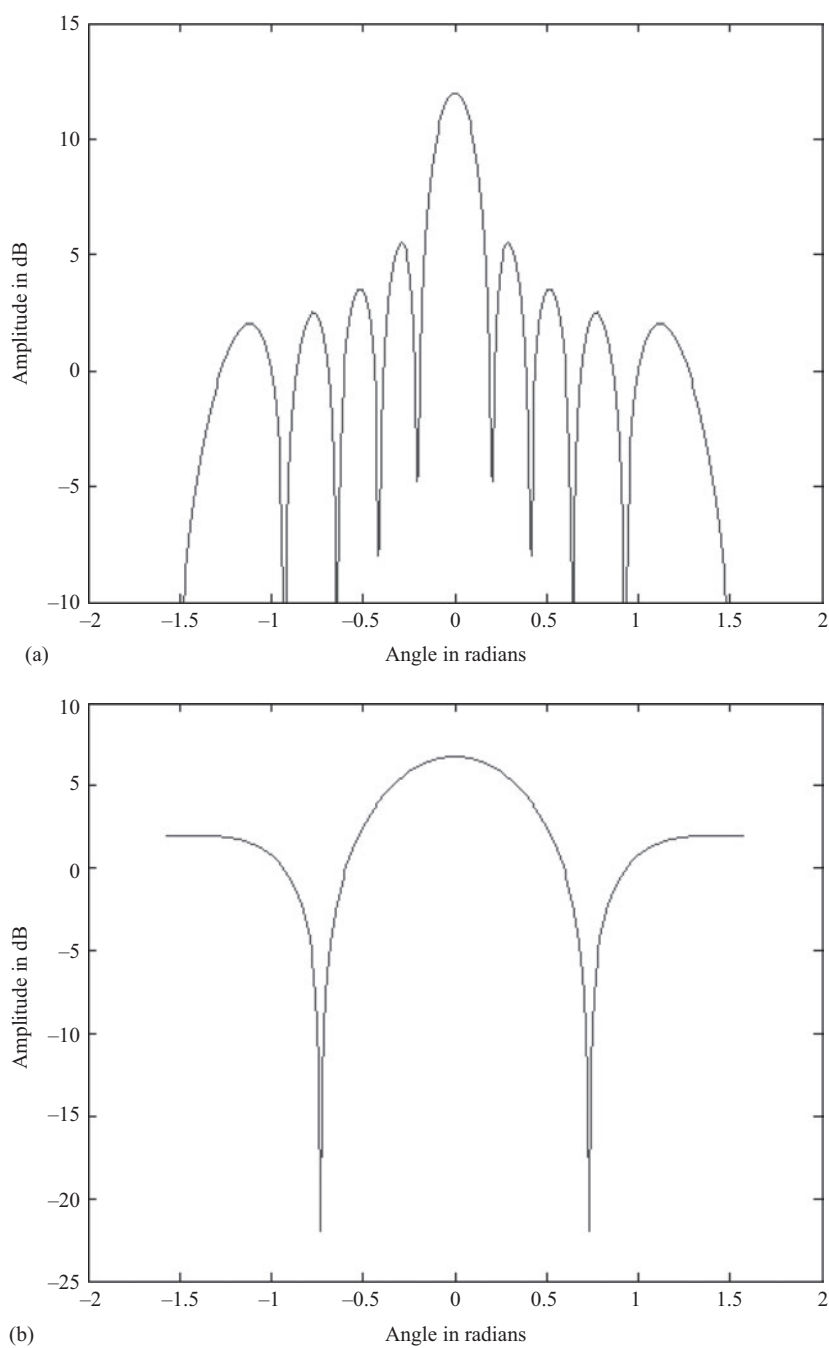


Figure 15.7 Outputs from a spatial Fourier transform: (a) 10 elements and (b) 3 elements.

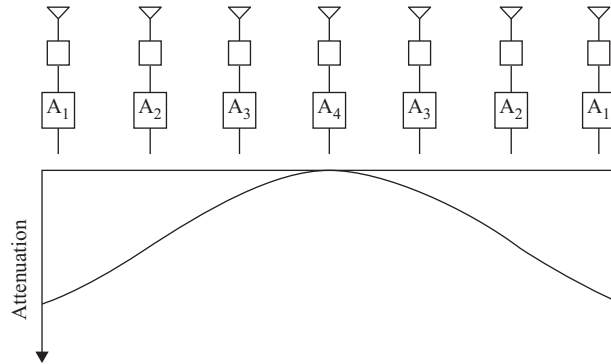


Figure 15.8 Antenna array with a weighting function.

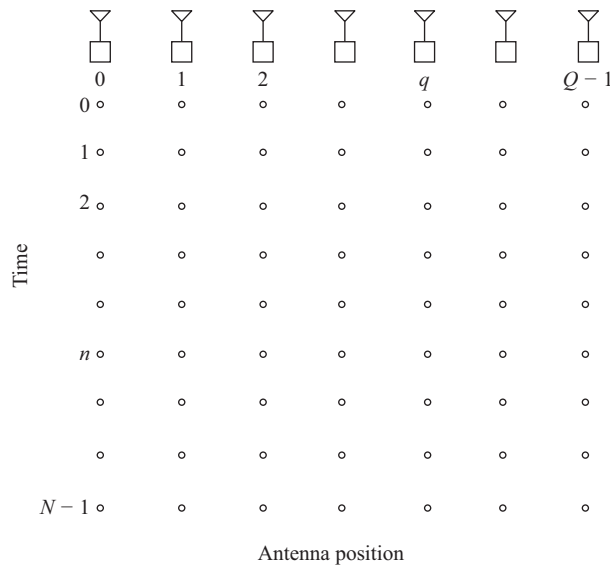


Figure 15.9 Data output from a linear antenna array.

beamwidth is wide due to a limited number of antenna elements and the sidelobes are high, it is difficult to separate simultaneous signals.

15.9 Two-Dimensional Fourier Transform [16, 17]

In this section, the two-dimensional Fourier transform is presented. The input data from an antenna array are two-dimensional: one is in the time domain and the other one is in the space domain, as shown in Figure 15.9. It should be noted that the time intervals are uniform and all the antennas are sampled at the same time. The distances between antennas are also uniform.

The Fourier transform performed on the time domain data will provide frequency information and the Fourier transform performed on the space data will provide AOA information. Like the Fourier transform in the frequency domain, the Fourier transform in the space domain can process simultaneous signals with different input frequencies and incident angles.

A two-dimensional Fourier transform of input data $x(q, t)$ can be written as

$$X(\hat{k}, f) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(q, t) e^{-j2\pi f t} e^{-j\frac{q\hat{k}\cdot\hat{x}}{c}} dq dt. \quad (15.33)$$

This equation is written in continuous form. If there is only one input signal, from (15.7), it can be written as

$$x(q, t) = A_1 e^{j\left(2\pi f_1 t - \frac{2\pi f_1 q\hat{k}_1\cdot\hat{x}}{c}\right)}. \quad (15.34)$$

Substituting this result in (15.33), the result is

$$\begin{aligned} X(\hat{k}, f) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} A_1 e^{-j2\pi(f-f_1)t} e^{j\frac{2\pi f q d(\hat{k}-\hat{k}_1)\cdot\hat{x}}{c}} dt dq \\ &= \int_{-\infty}^{\infty} A_1 e^{-j2\pi(f-f_1)t} dt \int_{-\infty}^{\infty} e^{j\frac{2\pi f q d(\hat{k}-\hat{k}_1)\cdot\hat{x}}{c}} dq. \end{aligned} \quad (15.35)$$

In this equation, the signal is assumed to be continuous in the time domain (i.e., in an analog receiver). However, it is difficult to consider the signal to be continuous in the space domain because the antennas are always in discrete locations. Even in an analog receiver system, the signals received in the space domain are discrete because the antennas are discretely located in space.

Considering that the space domain is discrete, the above equation can be written as

$$X(\hat{k}, f) = \sum_{q=0}^{Q-1} \int_{-\infty}^{\infty} x(q, t) e^{-j2\pi f t} e^{-j\frac{2\pi f q d\hat{k}\cdot\hat{x}}{c}} dt. \quad (15.36)$$

This is a two-dimensional Fourier transform with continuous data in the time domain but discrete data in the space domain.

If the collected data are discrete in both the time and space domains, as shown in Figure 15.9, the two-dimensional Fourier transform can be written as

$$X(\hat{k}, k) = \sum_{q=0}^{Q-1} \sum_{n=0}^{N-1} x(q, n) e^{-j\frac{2\pi n k}{N}} e^{-j\frac{2\pi k q d\hat{k}\cdot\hat{x}}{c}}. \quad (15.37)$$

In this equation, $\hat{k}$ is a unit vector representing the AOA calculated from the spatial Fourier transform and k is the discrete frequency component in the frequency domain calculated from the Fourier transform in the time domain.

The inverse Fourier transform in discrete form is

$$x(q, n) = \frac{1}{QN} \sum_{k=0}^{Q-1} \sum_{\hat{k}=0}^{N-1} X(k, \hat{k}) e^{j\frac{2\pi n k}{N}} e^{j\frac{2\pi k q d\hat{k}\cdot\hat{x}}{c}}. \quad (15.38)$$

The two-dimensional discrete Fourier transform requires $(MN)^2$ complex multiplications. Thus the calculation is rather complicated.

One can consider the two-dimensional Fourier as a combination of two one-dimensional Fourier transforms. For example, one can perform the time domain Fourier transform (one-dimensional) on all the antenna outputs to obtain the frequency domain information. Then, at each frequency component one can use all the outputs from the different antennas to perform the space Fourier transform (one-dimensional) to obtain the AOA information of that frequency component. Therefore the final results will be the frequency of each incoming signal and the corresponding AOA information.

In the two-dimensional Fourier transform, in order to suppress the sidelobes, weighting functions can be added to both the space and time domains. However, since the two-dimensional Fourier transform is complicated and has poor AOA resolution, it appears that this approach might not be very useful in digital receiver applications.

15.10 Frequency Sorting Followed by AOA Measurements

In the previously discussed phase measurement system it is assumed that the receiver processes only one input signal. Usually the phase measurement system can provide fine AOA information. The actual receiver used in a single-signal phase comparison system must have narrow bandwidth to limit the probability of intercepting more than one signal. However, a narrow bandwidth receiver has a low probability of intercept, which is not desirable for EW applications. A wideband phase measurement AOA system may be contaminated by simultaneous signals to produce erroneous information. Although theoretically the spatial Fourier transform can process simultaneous signals with different incident angles, the AOA resolution obtained is very coarse due to the limited antenna elements available in an EW system. It is highly questionable that a wideband AOA system using spatial Fourier transform can be used to separate signals with different incident angles.

In this section, a more promising approach is presented. This is a two-step approach. The first step is to separate the incoming signals by frequency through Fourier transform in the time domain. If the data are collected over a long period of time, the frequency resolution can be reasonably narrow. For example, a 100-ns-long data string can provide approximately 10 MHz of frequency resolution. In EW applications it is often suggested that simultaneous signals are not a severe problem when the frequency channel is 20 MHz wide. Sometimes even a 50- to 100-MHz channel width is considered in order to accommodate short pulses.

The first step is to use the data obtained from one antenna (say $q = 0$) to perform FFT in the time domain. From the power spectrum outputs, the frequencies of the input signals can be obtained. Suppose there are M input signals and their peaks occur at frequencies $k_1, k_2, \dots, k_M$. The outputs can be written as $X(0, k_1), X(0, k_2), \dots, X(0, k_M)$ where the first index is referenced to the antenna. Then the same peaks should be obtained from other antennas as

$$\begin{aligned}
 X(q, k_1) &= \sum_{n=0}^{N-1} x(q, n) e^{\frac{-j2\pi k_1 n}{N}} \\
 X(q, k_2) &= \sum_{n=0}^{N-1} x(q, n) e^{\frac{-j2\pi k_2 n}{N}} \\
 &\vdots \\
 X(q, k_M) &= \sum_{n=0}^{N-1} x(q, n) e^{\frac{-j2\pi k_M n}{N}},
 \end{aligned} \tag{15.39}$$

for $q = 1, 2, \dots, Q - 1$. In these equations, only the frequency components with peaks found from the $q = 0$ antenna are calculated at k_1, k_2 , and so forth. Thus there is less calculation than to find all the Fourier components from all the antenna outputs.

The second step is to find the AOA from $X(q, k_i)$ $i = 0, 1, 2, \dots, M$ obtained from (15.39). Since the frequency information is known, many different methods can be used to find the AOA information at that frequency. A couple of simple approaches are presented here.

The first method is through phase measurement. This can be considered a straightforward approach because the phase information is already available in $X(q, k)$. The FFT output is complex and can be written in terms of amplitude and phase as

$$X(q, k_i) = |X(q, k_i)|e^{j\phi(q, k_i)}, \quad (15.40)$$

where $\phi(q, k_i)$ is the phase of the input signal at frequency k_i at antenna element q . The phase difference of $\phi(q, k_i)$ between two antenna elements is related to the incident angle of the signal by

$$\phi(q_r, k_i) - \phi(q_s, k_i) = \frac{d_{rs} \sin(\theta_i)}{\lambda_i}, \quad (15.41)$$

where q_r and q_s are the two antenna elements and d_{rs} is the distance between them and θ_i and λ_i are the incident angles and the wavelength of the signal at frequency k_i , respectively. As mentioned in the multiple antenna elements case, the antenna pair with the shortest distance between them can be used to resolve AOA ambiguity. The antenna pairs with longer distances between them can be used to provide fine AOA resolution.

A second approach to find the AOA is to use the FFT in the spatial domain. In this approach, the antenna elements should be uniformly located. The inputs to the FFT are the $X(q, k_i)$ values with the same frequency component k_i . This equation can be written as

$$X(\hat{k}, k_i) = \sum_{q=0}^{Q-1} x(q, k_i) e^{-j \frac{2\pi k q d \hat{k}}{c}}. \quad (15.42)$$

The main beamwidth depends on the number of antenna elements and the weighting function used. The possible advantage of this approach over the phase measurement method is the ability to process simultaneous signals. When two incoming signals are of the same frequency but with different incident angles, the phase measurement method can produce erroneous results because the phase measurement cannot process simultaneous signals. While the spatial FFT approach may separate them, the two signals must be widely separated in AOA.

Other possible methods can be used to find the AOA information, such as Prony's method or the MUSIC method. Interestingly, many of the high-frequency resolution methods were originally developed for AOA estimation. These approaches need more calculations, but for small data length they might be applicable for some special cases.

15.11 Minimum Antenna Spacing [18–21]

In all the AOA measurement systems discussed in previous sections, the antennas cannot be spaced too far apart in order to avoid ambiguity in AOA information. The shortest distance between two antennas $d_{\min}$ must be less than half the wavelength of the highest frequency.

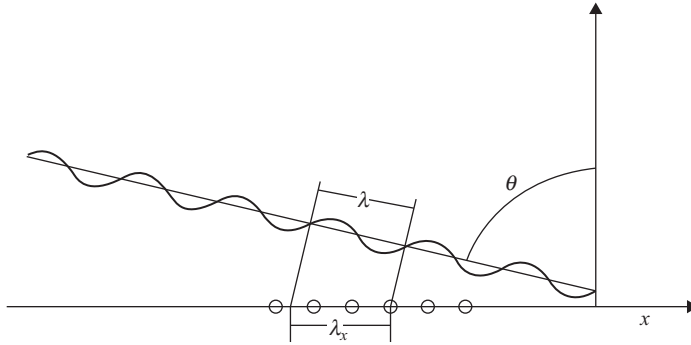


Figure 15.10 Wavelength along the x direction.

It can be written as $d_{\min} < \lambda_{\min}/2$, where $\lambda_{\min}$ is the wavelength corresponding to the highest frequency.

In order to emphasize this requirement, it will be presented again here from the sampling point of view. Figure 15.10 shows a sine wave incident onto an antenna array along the x direction. The wavelength of the input signal is λ and the corresponding wavelength along the x direction is

$$\lambda_x = \frac{\lambda}{\sin \theta}, \quad (15.43)$$

where θ is the incident angle. When $\theta = 90^\circ$ (i.e., the incoming signal is along the x -axis), then $\lambda_x = \lambda_{x\min} = \lambda$. In order to fulfill the Nyquist theorem, one must sample twice per cycle; that is, the shortest distance between two adjacent antennas $d_{\min}$ must be

$$d_{\min} < \frac{\lambda_{\min}}{2}, \quad (15.44)$$

where $\lambda_{\min}$ is the signal with the shortest wavelength (the highest frequency) the antenna array is anticipated to intercept.

Now let us consider some practical problems in antennas. A spiral antenna is commonly used in EW applications, especially in airborne systems. This kind of antenna covers a wide frequency range and a wide angle. It can receive signals with either vertical or horizontal polarization. Only when the input signal is circularly polarized in the opposite direction of the spiral does the antenna have difficulty intercepting it. A spiral antenna can cover 2 to 18 GHz. The diameter of the antenna is approximately equal to $\lambda_{\max}/2$, where $\lambda_{\max}$ is the wavelength corresponding to the lowest frequency. For an antenna with a 2- to 18-GHz frequency range, $\lambda_{\max} = 3 \times 10^{10}/2 \times 10^9 = 15$ cm and $\lambda_{\min} = 3 \times 10^{10}/18 \times 10^9 = 1.67$ cm. Thus the diameter of the antenna is about 7 cm. The closest distance one can put two antennas together is 7 cm from center to center.

In order to meet the relation in (15.44), the minimum distance between two spiral antennas must be less than 0.8 cm (1.67 cm/2). Since the antenna diameter is 7 cm, it is impossible to install the antenna this close. As a result, if the spiral antennas are used in the array to cover a wide frequency, it is inevitable to have ambiguity problems at a high-input frequency. The ambiguity problem might be solved through the Chinese remainder theorem, which is discussed in the next section.

15.12 Chinese Remainder Theorem [22, 23]

The story goes like this. In ancient times in China, there was a general who wanted to know how many bodyguard soldiers he had. Instead of counting the soldiers directly, he arranged the soldiers in rows several times. He did not count the number of rows, but only counted the remainder of each arrangement to find the total number of soldiers. Two examples are used to illustrate this concept.

Example 1

The general arranged his soldiers in rows of 7 and the remainder was 3, and he rearranged his soldiers in rows of 10 and the remainder was 2. The two numbers 7 and 10 must be relative prime of each other. If he had less than 70 (7×10) soldiers, he found the total number of soldiers from the two remainders 3 and 2.

If the unknown number $x < 70$ is divided by 7 with a remainder of 3, the possible values of x are

$$x = 3 + 7n = 3, 10, 17, 24, 31, 38, 45, 52, 59, \text{ and } 66,$$

where n is an integer from 0 to 9 ($10 - 1$). Similarly, the same unknown number divided by 10 with a remainder of 2 can be

$$x = 2 + 10n = 2, 12, 22, 32, 42, 52, \text{ and } 62,$$

where n is from 0 to 6 ($7 - 1$). Comparing these two series, one can find that 52 is the correct answer. The number of arrangements can be extended to more than two. For example, the soldiers can be rearranged in rows of 11. Under this condition, the total number of soldiers can be increased to $x < 770$ ($7 \times 10 \times 11$).

The Chinese remainder theorem can be stated in a slightly formal way as follows. If an unknown x is divided by m_i and the remainder is a_i , the result can be written with the following notation

$$x = a_i \bmod(m_i), \quad \text{where } i = 0, 1, \dots, I - 1. \quad (15.45)$$

In the Chinese remainder theorem, the m_i are relative prime with different i values. The solution of this problem (without proof) can be found as

$$\begin{aligned} M &\equiv \prod_{i=0}^{I-1} m_i & t_i &\equiv \frac{M}{m_i} \\ x &= \left(\sum_{i=0}^{I-1} a_i t_i u_i \right) \bmod(M) \\ t_i u_i &\equiv 1 \bmod(m_i), \end{aligned} \quad (15.46)$$

where I is total number of arrangements and u_i is an integer. Let us use the above example to demonstrate the use of this equation.

Example 2

Given $a_0 = 3, m_0 = 7; a_1 = 2, m_1 = 10; a_2 = 5, m_2 = 11$, where 7, 10, and 11 are relative prime, find x . From the above equation, $M = 7 \times 10 \times 11 = 770$, $t_0 = 110$, $t_1 = 77$, and $t_2 = 70$, thus

$$t_0 u_0 = 110 \times 3 = 1 \bmod(7) \Rightarrow u_0 = 3$$

$$t_1 u_1 = 77 \times 3 = 1 \bmod(10) \Rightarrow u_1 = 3$$

$$t_2 u_2 = 70 \times 3 = 1 \bmod(11) \Rightarrow u_2 = 3.$$

The u_i are obtained through trial and error. Coincidentally, all $u_i = 3$. Then,

$$\begin{aligned} x &= (3 \times 110 \times 3 + 2 \times 77 \times 3 + 5 \times 70 \times 3) \bmod(770) \\ &= (990 + 462 + 1050) \bmod(770) = 2502 \bmod(770) = 192. \end{aligned}$$

Of course, this result can be obtained by using the same approach as in Example 1 by arranging all the possible numbers in three rows. The correct answer will appear in all three rows. With advances in digital signal processing, this approach might be reasonably simple.

15.13 Application of the Chinese Remainder Theorem to AOA Measurements

As mentioned in Section 15.11, one cannot put two antennas close together to fulfill (15.44) because of the large size of the antennas. For spiral antennas (from 2 to 18 GHz) with a diameter of 7 cm, one can put the two antennas $5d_{\min} = 8.35$ cm apart. This arrangement produces an ambiguity problem. In order to resolve this ambiguity, another antenna pair is needed. Figure 15.11 shows such an arrangement. One antenna pair is $5\lambda_{\min}$, the other pair is $8.5\lambda_{\min}$, and the distance between the two end antennas is $13.5\lambda_{\min}$ ($5 + 8.5$). All three numbers—10, 17, and 27—in terms of $\lambda_{\min}/2$ are prime ones. However, these three antennas can be considered as two pairs, and the third pair produces no additional information. In this calculation, only two pairs of antennas will be considered and they are separated by 10 and $17 \lambda_{\min}/2$.

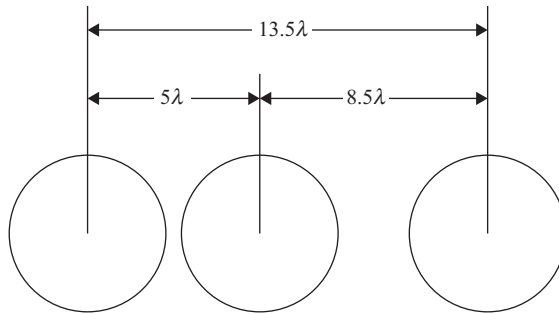


Figure 15.11 Three-element phase interferometric system.

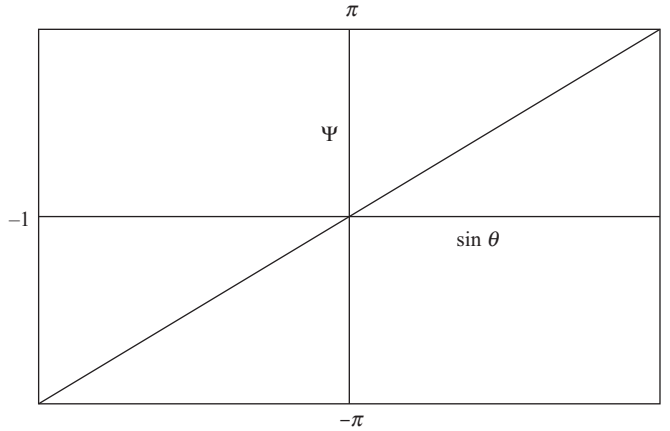


Figure 15.12 ψ versus $\sin \theta$ for $d = \lambda/2$.

Although the remainders are used to find the desired AOA, the calculation is slightly different from the Chinese remainder theorem mentioned in the previous section. First, let us study the case without ambiguity. Assume that the distance between the antenna elements is $\lambda/2$. Under this condition there is no ambiguity and the relation in (15.20) can be written as

$$\psi = \pi \sin \theta. \quad (15.47)$$

In this equation, ψ varies from $-\pi$ to π . As a result, $\sin \theta$ ranges from -1 to 1 . This result is shown in Figure 15.12 and ψ versus $\sin \theta$ is a straight line. In this figure, there is no ambiguity. One value of ψ corresponds to one value of $\sin \theta$.

When $d = 5\lambda$, the result obtained from (15.20) is

$$\psi = 10 \pi \sin \theta. \quad (15.48)$$

In this equation, there should be 10 ambiguities. Since $\sin \theta$ ranges from -1 to 1 , the angle ψ can change from -10π to 10π . However, the value of angle measurement ψ is limited from $-\pi$ to π . Any angle over this range will be folded back into this range. That means any angle value ϕ less than $-\pi$ will be written as $\phi + n\pi$ and any angle value ϕ greater than π will be written as $\phi - n\pi$. The relation of (15.48) is shown in Figure 15.13. In this figure, for every value of ψ there are 10 possible values of $\sin \theta$, and this is the expected result. If a certain phase angle ψ is measured, it is impossible to determine the incident angle of the signal. In comparison with Figure 15.12, the resolution is better because the slope in this figure is steeper.

In order to resolve this ambiguity problem, another pair of 8.5λ antennas is required. The result is shown in Figure 15.14. In this figure at any incident angle θ_1 there are two unique values ψ_{10} and ψ_{17} . Thus if ψ_{10} and ψ_{17} are measured, the incident angle θ_1 can be obtained. In actual measurement, a conversion table can be generated; once ψ_{10} and ψ_{17} are obtained, the incident angle θ_1 can be found from the table.

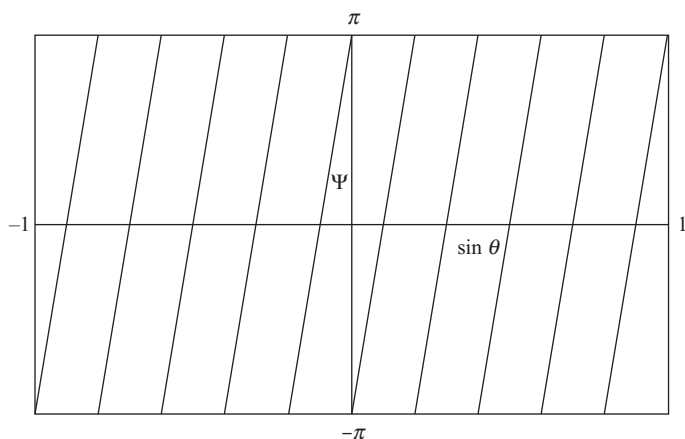


Figure 15.13 ψ versus $\sin \theta$ for $d = 5\lambda$.

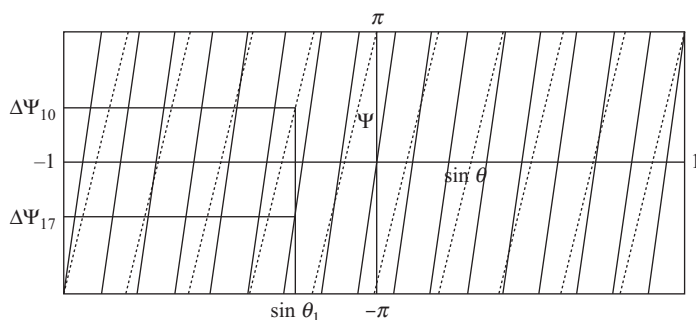


Figure 15.14 ψ versus $\sin \theta$ for $d = 5\lambda$ and $d = 8.5\lambda$.

15.14 Practical Considerations in the Chinese Remainder Theorem [24]

It appears that the Chinese remainder theorem can be used to resolve the antenna array problem in a very neat way. In practice, when the remainder theorem is used to solve any type of problem, there are quite a few problems one must consider very carefully. Some of these problems include the following:

1. In the theoretical discussion in Section 15.12, the remainders are measured very accurately as integers. In practice, there are errors in the measured remainders. In conventional AOA measurement, an error in the final value only affects the accuracy of the measurement. However, if the remainders are used to determine the actual value, a measurement in the final value (the remainder) will cause catastrophic error, which means the error can be very large. This large error can be easily realized from the examples in Section 15.12. If one of the remainders is changed by one numerical number, the result will be totally different.

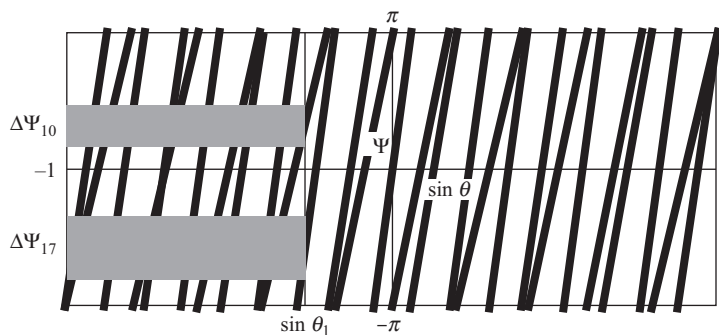


Figure 15.15 ψ versus $\sin \theta$ for $d = 5\lambda$ and $d = 8.5\lambda$ with noise.

2. If noise in the remainder measurement is taken into consideration, the overall ambiguous range must be decreased. For example, if the dividers are 5 and 7, the maximum number should be 34 ($5 \times 7 - 1$). All numbers from 0 to 34 are possible solutions. If the errors are allowed in the remainders, the maximum number should be less than 34. Figure 15.15 provides a visual understanding of this problem.

This figure shows the same result as in Figure 15.14. The only difference is that there is noise in these remainders, and the noise increases the width of the lines. In this figure, at incident angle θ_1 , the corresponding electrical angles measured as ψ_{10} and ψ_{17} cover a range of values rather than a single value as indicated in Figure 15.14. If one decreases the range of θ_1 , this method may still be used to measure the AOA, but the total angle coverage will also be decreased.

3. The relationship between y and $\sin \theta_1$ is linear, as indicated by (15.20) and the straight lines in Figures 15.14 and 15.15. If actual data are collected from antennas, it is possible that these lines are no longer linear, but slightly curved. The deviation from a linear line complicates the measurement procedure. It means it is difficult to generate the conversion table to obtain the incident θ_1 from ψ_{10} and ψ_{17} . The curvature in the curves must be included in the conversion table.
4. Additional hardware can be used to improve the measured results. For example, instead of using two antenna pairs and their remainders to provide the incident angle, one can add another antenna and use three antenna pairs to provide more redundancy in the measurement. This approach not only needs one more antenna element and receiver, but the conversion table will also be more complicated. Increasing the order of redundancy can reduce the probability of erroneous results at the cost of more hardware.

15.15 Hardware Considerations for Digital AOA Measurements [7, 25]

In analog AOA measurement systems, the most difficult problem is to build many parallel antennas/receivers and match the amplitude and phase among them. Some experimental systems have difficulty achieving amplitude and phase matching among different receivers, especially for wideband systems.

In an analog amplitude AOA comparison system, time-sharing one receiver among many antennas is a simpler approach than using many receivers in parallel [7]. However, this approach cannot be extended to an analog phase AOA measurement system because one must compare the phase simultaneously from many different receivers. However, this time-sharing idea can be used in a phase measurement system if it is a digital system.

In digital receiver design, as mentioned previously, the ADC performance is much faster than the processing components following it. Thus one can use one ADC to collect data and several processors to process it. To measure AOA using a digital approach, data as shown in Figure 15.9 should be collected.

Figure 15.16 shows an arrangement that uses one RF receiver and one ADC to collect the necessary data. In this figure there are four antennas; the receiver and ADC are time-shared among the four antennas. There is a four-to-one switch that sequentially rotates through the four antennas. The switching time must match the sampling time of the ADC. If the sampling time is t_s , the switch must operate at the same speed, switching to the next antenna at time t_s .

If it is desirable to have the data collected from all four antennas at the same time, as shown in Figure 15.9, delay lines should be inserted to compensate for the switching delay. The delay time should be in the unit of t_s . In Figure 15.16, antenna 0 has no delay, antenna 1 has t_s delay, antenna 2 has $2t_s$ delay, and antenna 3 has $3t_s$ delay. Since an ADC can operate at 1 GHz, which corresponds to $t_s = 1$ ns, the delay time is not a big problem.

It is anticipated that the delay lines can be omitted in Figure 15.16. Under this condition, the data collected are shown in Figure 15.17. The data collected are not at the same time

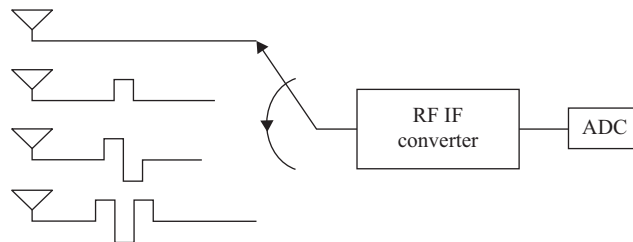


Figure 15.16 One receiver shared among four antennas.

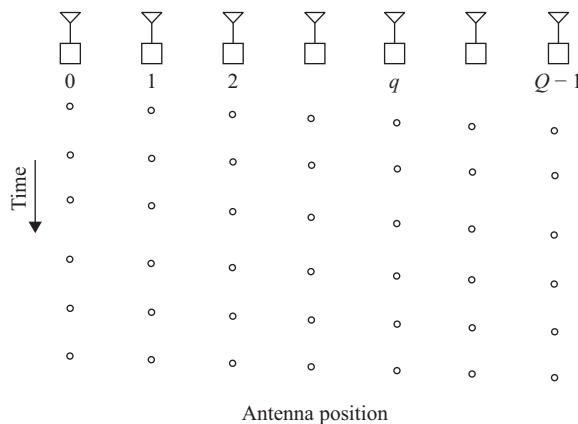


Figure 15.17 Digital data collected from a time-sharing system without delay correction.

instant, but skewed in time. If this type of data is available, digital signal processing might be used to take the time skew into consideration and find the correct AOA.

References

- [1] Capon J. 'High-resolution frequency-wavenumber spectrum analysis.' *Proceedings of the IEEE* 1969;**57**(8):1408–1418.
- [2] Reddi SS. 'Multiple source location-a digital approach'. *IEEE Transactions on Aerospace and Electronic Systems* 1979;**AES-15**(1):95–105.
- [3] Johnson DH. 'The application of spectral estimation methods to bearing estimation problems'. *Proceedings of the IEEE* 1982;**70**(9):1018–1028.
- [4] Kumaresan R, Tufts DW. 'Estimating the angles of arrival of multiple plane waves'. *IEEE Transactions on Aerospace and Electronic Systems* 1983;**AES-19**(1):134–139.
- [5] Shan TJ, Wax M, Kailath T. 'Spatial smoothing approach for location estimation of coherent sources'. *Proceedings of the Seventeenth IEEE Asilomar Conference on Circuits, Systems, and Computers*. New York: IEEE; 1983:367–371.
- [6] Li F, Vaccaro RJ. 'On frequency-wavenumber estimation by state-space realization'. *IEEE Transactions on Circuits and Systems* 1991;**38**(7):800–804.
- [7] Tsui JBY. *Microwave Receivers with Electronic Warfare Applications*. New York: John Wiley & Sons; 1986.
- [8] Jordan EC. *Electromagnetic Wave and Radiating Systems*. Englewood Cliffs, NJ: Prentice Hall; 1950.
- [9] Kraus JD. *Electromagnetics*. New York: McGraw-Hill; 1953.
- [10] Longstaff ID, Chow PEK, Davies DEN. 'Directional properties of circular arrays'. *Proceedings of the Institution of Electrical Engineers* 1967;**114**(6):713–718.
- [11] Sheleg B. 'A matrix-fed circular array for continuous scanning'. *Proceedings of the IEEE* 1968;**56**(11):2016–2027.
- [12] King WP, Harrison CW. *Antennas and Waves: A Modern Approach*. Cambridge, MA: MIT Press; 1969.
- [13] Ma MT. *Theory and Application of Antenna Arrays*. New York: John Wiley & Sons; 1974.
- [14] Shelton J, Kelleher KS. 'Multiple beams from linear arrays'. *IRE Transactions on Antennas and Propagation* 1961;**9**(2):154–161.
- [15] Jacobs E, Ralston EW. 'Ambiguity resolution in interferometry'. *IEEE Transactions on Aerospace and Electronic Systems* 1981;**AES-17**(6):766–780.
- [16] McClellan JH. 'Multidimensional spectral estimation'. *Proceedings of the IEEE* 1982;**70**(9):1029–1039.
- [17] Haykin S (ed.). *Array Signal Processing*. Englewood Cliffs, NJ: Prentice Hall; 1985.
- [18] Kraus JD. *Antennas*. New York: McGraw-Hill; 1950.
- [19] Rumsey V. *Frequency Independent Antennas*. New York: Academic Press; 1966.

- [20] Stutzman WL, Thiele GA. *Antenna Theory and Design*. New York: John Wiley & Sons; 1981.
- [22] Taylor FJ. 'Residue arithmetic a tutorial with examples'. *Computer* 1984;**17**(5):50–62.
- [23] Wolf J.K. 'The Chinese remainder theorem and applications', in Blake JF, Poor HV (eds), *Communications and Networks: A Survey of Recent Advances*. New York: Springer; 1985: chap 16.
- [24] McCormick WS, Tsui JBY, Bakkie VL. 'A noise insensitive solution to an ambiguity problem in spectral estimation'. *IEEE Transactions on Aerospace and Electronic Systems* 1989; **25**(5):729–732.
- [25] Tsui DC. Wright Laboratory/Association for the Advancement of Artificial Intelligence. Private communication.

Time Reversal Study

16.1 Objective

In this chapter we will examine ways to determine the input signal of a receiver from the receiver's output signal using a time reversal filter. The input signal is usually measured by a receiver, which will distort the input signal. Assuming that receiving system is a linear system, then its output information represents the input signal modified by the receiving system's impulse response. To compensate for the impact of the receiving system on the input signal, the impulse response of the receiving system is used to design a time reversal filter. The time reversal filter can compensate for signal distortion caused by the receiving system, thus restoring the input signal. In most applications the input information is of interest, and by knowing the impulse function of the receiving system, the input signal can be obtained. The time reversal filter was first proposed for acoustics [1, 2] and is widely used in communications [3–6]. In this chapter we will demonstrate how the time reversal filter can expedite the calibration of an angle of arrival (AOA) system in which the information of the input signal is used to find the angle information on the signal. The materials presented in this chapter were obtained from a collaborative project of Dr. Lihyeh Liou and David Lin of the Air Force Research Laboratory (AFRL) [7] and Dr. Robert Qiu of Tennessee Technological University. Special thanks go to Dr. Liou for carefully reviewing and helping to revise this chapter.

Since an impulse function is difficult to generate in a laboratory environment, chirp signals can be used as an alternative to determine the impulse response. Based on the impulse response, a time reversal filter can then be determined and used to find the input signal. The impact of the chirp signal on the performance of the resulting time reversal filter is investigated in this chapter. The concept of fractional time delay, which delays the time less than the sampling period, is also discussed. This concept is important for calibrating an AOA measurement system.

16.2 Application to AOA Measurement

Figure 16.1 shows an AOA measurement system with two antennas and receivers. If the two antennas and receivers have identical impulse response, comparing the outputs of the two systems can produce accurate AOA results. If the impulse responses of the two systems are

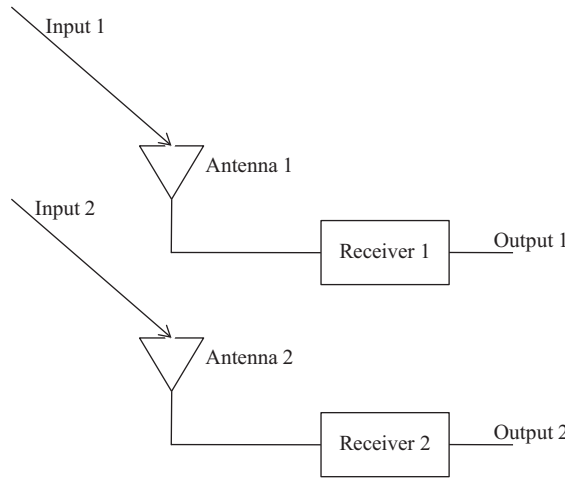


Figure 16.1 A simple AOA measurement system.

not identical the two systems must be calibrated. A conventional way to calculate the two systems uses continuous wave (CW) signal as the input. The amplitude and phase differences between the two systems are measured as a function of input frequency. Good calibration needs many input frequencies. This process is elaborate and a large calibration table will be generated.

In Figure 16.1, if two receivers' input signals can be determined from their output signals, amplitude and phase comparison methods can then be used to measure the AOA since the receivers' impulse functions are not included in the calculations. The receiver's input and output signals are related through the following function:

$$\begin{aligned} y(t) &= x(t) \otimes h(t) \\ Y(f) &= X(f)H(f), \end{aligned} \tag{16.1}$$

where $x(t)$ and $X(f)$ are the inputs in the time and frequency domains, $y(t)$ and $Y(f)$ are the outputs in time and frequency domains, $h(t)$ and $H(f)$ are the impulse responses in the time and frequency domains, and $\otimes$ represents convolution. Once the receiver's output is measured, its input signal can be obtained as

$$\begin{aligned} x(t) &= y(t) \otimes h'(t) \\ h'(t) &= F^{-1} \left\{ \frac{H(f)^*}{|H(f)|^2} \right\}, \end{aligned} \tag{16.2}$$

where the operator $*$ represents the complex conjugate. Since the inputs are not distorted by the impulse response of the receiving system, they can be used to obtain the AOA information. The function $h'(t)$ is defined as the time reversal function of function $h(t)$.

16.3 Measuring the Impulse Response

The definition of the impulse response is the output of a system upon an impulse input. But in a real laboratory test, it is difficult to generate an impulse function. For example, if the sampling frequency is 2.56 GHz, the corresponding sampling time is about 0.39 ns ($1/2.56 \times 10^{-9}$). Therefore one must generate a pulse with a pulse width (PW) less than 0.39 ns. Even if this pulse can be generated, the energy of the pulse will be limited and it is usually difficult to generate a meaningful system output.

In order to overcome this difficulty, a chirp signal is usually used to replace the impulse function. The basic concept is that the bandwidth of an impulse function is infinite and a chirp signal can also cover a broad bandwidth, therefore a chirp signal might be a good substitute for an impulse function. Besides, a chirp signal, $x(t)$, if its bandwidth is large, satisfies the following relation [7]:

$$\delta(t) \cong x(-t) \otimes x(t), \quad (16.3)$$

where $\delta(t)$ is an impulse function and $x(-t)$ is the time reversal version of $x(t)$. When the input signal is a chirp signal, the system output can be written as shown in (16.1). Based on (16.1) and (16.3), the following equation can be derived:

$$x(-t) \otimes y(t) = x(-t) \otimes x(t) \otimes h(t) \cong \delta(t) \otimes h(t) \cong h(t)$$

or

$$x(t) \otimes y(-t) = x(t) \otimes x(-t) \otimes h(-t) \cong \delta(t) \otimes h(-t) \cong h(-t). \quad (16.4)$$

With (16.4), we can determine system's impulse response:

1. Digitize both the input and the output signals to obtain $x(n)$ and $y(n)$. The input is a linear upchirp signal with frequencies covering the entire system bandwidth.
2. Convert both $x(n)$ and $y(n)$ into the frequency domain through the fast Fourier transform (FFT) operation and the outputs are $X(k)$ and $Y(k)$.
3. Multiply $X(k)$ and the complex conjugate of $Y(k)$ or the $Y(k)$ and the complex conjugate of $X(k)$. Notice the complex conjugate of $Y(k)$ is the FFT of $y(-t)$ and the complex conjugate of $X(k)$ is the FFT of $x(-t)$.
4. Take the inverse FFT operation to obtain the desired results.

These operations can be represented in the following equation:

$$H(k) = Y(k)X'(k) \rightarrow H'(k) = Y'(k)X(k)$$

$$h(n) = \sum_{k=0}^{N-1} H(k) e^{\frac{j2\pi kn}{N}}. \quad (16.5)$$

Figure 16.2 shows the results obtained from both the impulse signal and a chirp signal. The system is arbitrarily chosen as a four-tap Butterworth filter with input bandwidth from 141 to 1140 MHz. Since the sampling frequency is 2.56 GHz, the Nyquist input bandwidth is

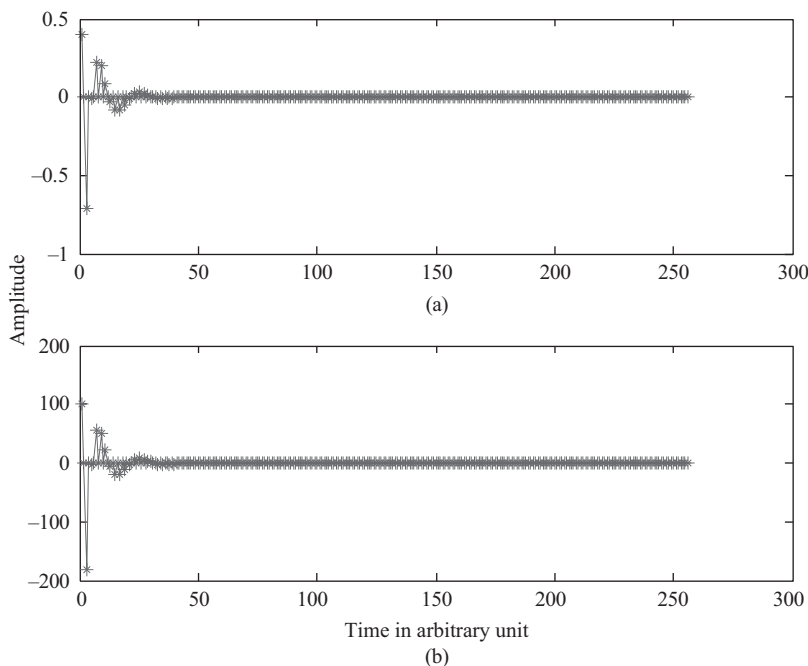


Figure 16.2 Impulse responses of a four-tap Butterworth filter: total length from (a) impulse input and (b) chirp input.

from 0 to 1280 MHz. In this simulation 256 points of input data are used. Figure 16.2 shows whole output and Figure 16.3 shows only the first 50 output samples. It is anticipated that the length of the impulse used in the time reversal filter design is less than 50 taps. The amplitude of the impulse is 1 and the chirp frequency changes from 0 to 1280 MHz, which is half the sampling frequency.

Although these two plots in Figures 16.2 and 16.3 look alike, their amplitudes have different scales. The next section evaluates the use of a chirp signal to simulate an impulse.

16.4 Quality of the Impulse Response Generated by a Chirp Signal

If the impulse response generated by an impulse input is considered as ideal, the impulse response generated from the chirp signal should be compared with it. The difference between them is then used as a parameter to decide how close the impulse response generated by the chirp signal is to the ideal case. Since the two plots in Figure 16.2 have different y -axis scales, one must make them comparable. One approach to achieve this is to find the maximum value in the impulse responses from both outputs of Figures 16.3(a) and 16.3(b) and take the ratio (R) of them. Multiply the outputs from the impulse input (Figure 16.3(a)) and the result is shown in Figure 16.4, which is similar to Figure 16.3(b) generated by the chirp signal.

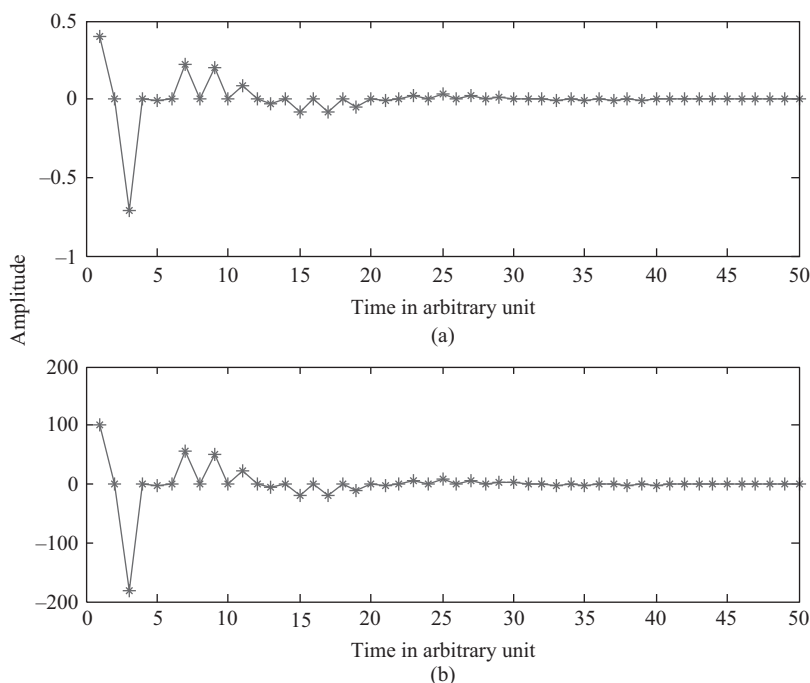


Figure 16.3 Impulse responses of a four-tap Butterworth filter: first 50 points from (a) impulse input and (b) chirp input.

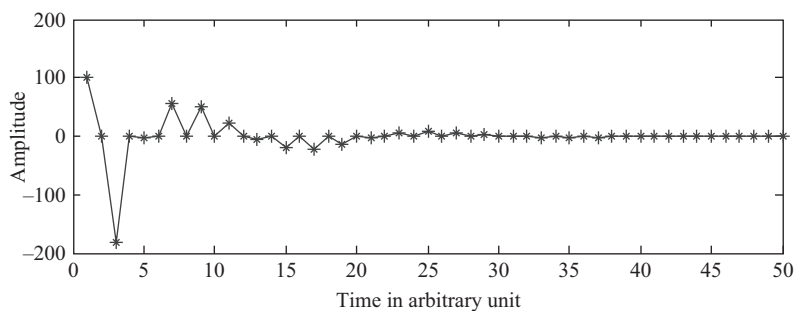


Figure 16.4 Scaled impulse response determined using an impulse signal as input.

The criterion for comparing the similarity between Figures 16.3(b) and 16.4 is to find the difference between each point and reference as Δh . In this study there are 30 points in Δh . This number of 30 is obtained from Figure 16.3. It is desirable to use the minimum number of taps in the time reversal filter design to reduce the calculation load. Sum the squaring of the Δh and refer to it as the error data. Figure 16.4 is used as the reference output because it is generated from the impulse input. Sum the squaring of the data in Figure 16.4 and refer to it as the total value. The ratio of the error data to the total value is referred to as the relative error.

The reason for using the relative error is to show the percentage error rather than the absolute value. These operations can be further explained through following equations:

$$\begin{aligned}
 \Delta h(n) &= h_i(n) - h_c(n) \\
 E &= \sum_{n=1}^{30} \Delta h(n)^2 \\
 T_v &= \sum_{n=1}^{30} h_i(n)^2 \\
 E_r &= \frac{E}{T_v},
 \end{aligned} \tag{16.6}$$

where h_i and h_c are the impulse responses obtained through the impulse input and chirp input, respectively; E is the error data; T_v is the total value; and E_r is the relative error.

16.5 Impact on Input Data Length

In Section 16.3, 256 points are used for the simulation and the results are shown in Figures 16.2 through 16.4. It is interesting to investigate the impact of the data length on relative error. To generate a chirp signal from 0 to 1280 MHz in 100 ns only 256 data points are obtained since the sampling rate is 2.56 GHz. Intuitively there are not enough points since the chirp rate is very fast or the PW is short. When the PW increases and the frequency range stays at 0 to 1280 MHz, the chirp rate decreases. Using the relative error as the measuring parameter, the result is shown in Figure 16.5. In this figure the PW changes from 0.1 to 12.8 μ s. The relative error decreases from about 3.8×10^{-4} to 1.1×10^{-12} , which is an eight order of magnitude difference.

From this simple study it appears that using a slow chirp signal can produce a better impulse response than a fast chirp signal. In order to study other impacts of the chirp signal on the impulse response, a PW of 6.4 μ s (16,384 data points) is used. This value is arbitrarily chosen and under this condition the relative error is 1.3×10^{-11} .

16.6 Impact on Impulse Response from an Input Chirp Signal

In this section the effect of the chirp signal on determining the impulse response is studied; there is no noise in the input signal for this study. The relative error defined in Section 16.3 will be used as a measurement parameter. Two parameters in the chirp will be studied. The first one is the initial phase of the chirp signal and the second is the frequency range.

When the signal frequency range is from 0 to 1280 MHz and the initial phase (ϕ_0 in (13.1)) of the input signal changes from 0 to π , the relative error is shown in Figure 16.6. The relative error changes from about 6.6×10^{-8} to 1.3×10^{-11} . Although the change is three orders of magnitude, the worst relative error is still less than 10^{-7} . Therefore the error contribution from the initial phase change can be neglected. Besides, it might be difficult to control the initial phase of the input signal in data while conducting an actual experiment.

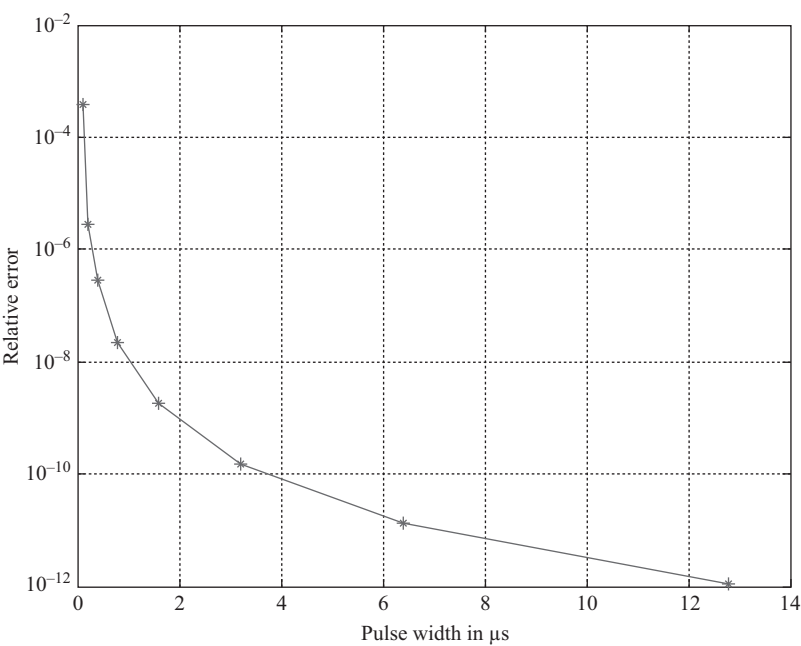


Figure 16.5 Relative error versus PW.

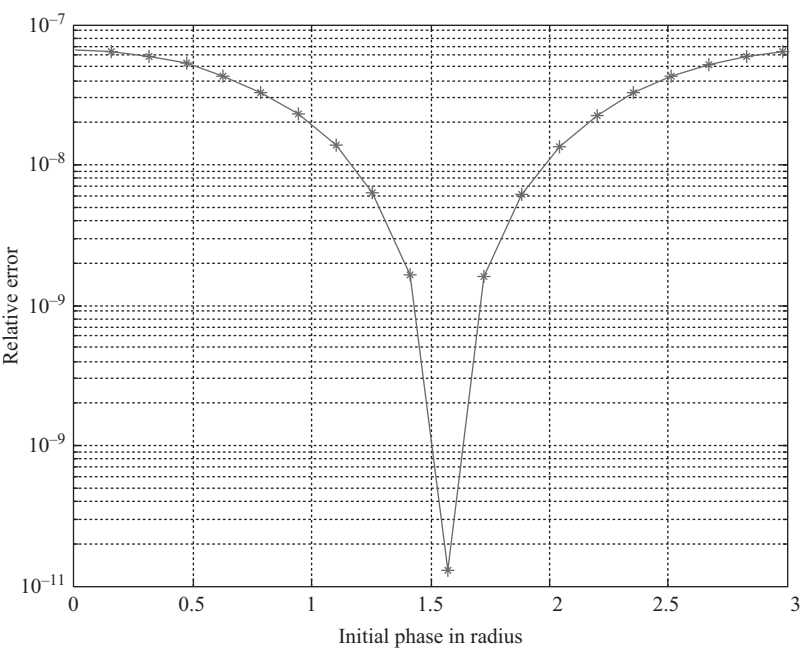


Figure 16.6 Relative error versus initial phase of the chirp signal.

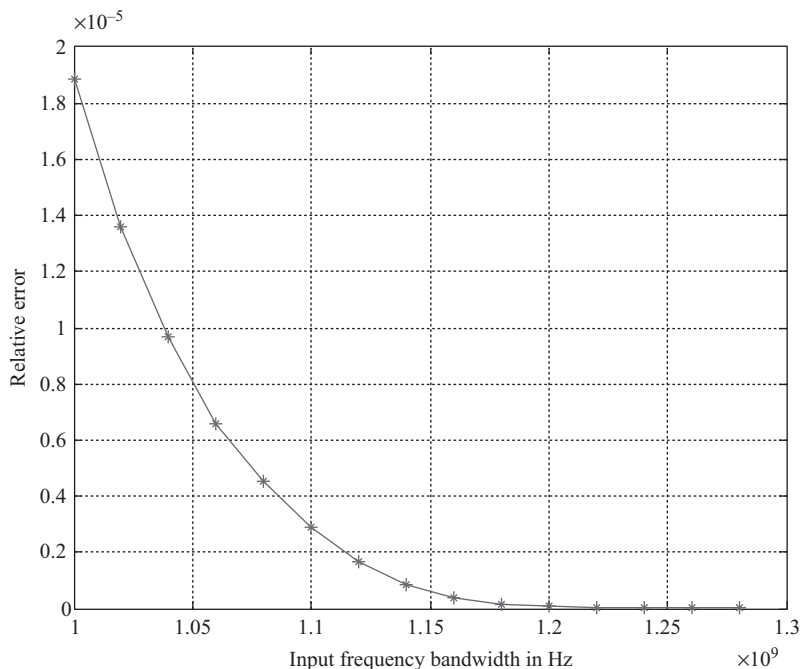


Figure 16.7 Relative error versus chirp signal bandwidth.

It is important to find the impact of the bandwidth of the chirp signal. Theoretically the impulse input in the time domain should cover all the Nyquist bandwidth. In other words, when the input covers all the Nyquist bandwidth, it can loosely represent an impulse function in the time domain. However, in actual measurement the signal generated may not cover the entire Nyquist bandwidth. Thus the bandwidth effect of the input signal will be studied here. It is assumed that the chirp signal should cover at least all the desired frequencies of the 1000-MHz bandwidth. When the initial phase of the input signal is at $\pi/2$ (the best initial phase condition as shown in Figure 16.6), the relation between the chirp signal bandwidth and the relative error is shown in Figure 16.7. When the input bandwidth is in at 1000 MHz (from 141 to 1140 MHz), the relative error is about 1.9×10^{-5} and when the input bandwidth increases to 1280 MHz (from 0 to 1280 MHz), the relative error decreases to about 1.3×10^{-11} . The improvement is close to six orders of magnitude. From this simulation it appears that in order to generate an accurate impulse function, the input signal bandwidth should cover more than the designed frequency range. An impulse function has a flat spectrum with an infinite bandwidth, therefore the response of a system excited by a chirp signal with a wider bandwidth will be closer to the impulse response compared with the one induced by a chirp signal with a narrower bandwidth.

When the initial phase of the input signal is at zero (the worst initial phase condition as shown in Figure 16.6), at a bandwidth of 1000 MHz the relative error is still about 1.9×10^{-5} , but when the bandwidth increases to 1280 MHz the relative error is about 6.6×10^{-8} . This result further illustrates the impact of the initial phase and bandwidth of the input signal.

16.7 Recovering the Input Signal from the Output Signal

The purpose of this study is to find out whether, with the time reversal filter, the input signal can be recovered from the output signal. Once the impulse function h is found, one can determine the input from the output through (16.2). Theoretically the input signal can be obtained from the output signal passing through the time reversal filter. This and the next section study this effect through simulation.

When the input signal passes through a filter and its time reversal filters there is a transient effect on the output signal. The input signal does not contain the transient effect induced by the filter. Even after the leading edges of the input and output signals are neglected, it might still be difficult to compare the input signal against the inverse filter output. Thus we need to use a different approach to mitigate the transient effects of the filter when evaluating time reversal filters. In order to eliminate this difficulty, two filters are used as shown in Figure 16.8. In this figure, the input signal is divided into two parallel paths to two receiving filters. These two filters have different responses, i.e., h_{11} and h_{12} . The two outputs from the filters are different and they are labeled as output 11 and output 12. The simulation is to find the outputs for the inverse filters h_{21} and h_{22} and compare their outputs, i.e., output 21 and output 22. If both inverse filters function as expected, output 21 and output 22 should be approximately identical. Since both filter outputs have transients associated with them, the results can be properly compared.

This approach with two different filters is used to emulate a practical arrangement. Usually to find the AOA information multiple antennas and receivers are used. For simplicity, let us refer to the combination of the antenna and receiver as a receiving unit. All the receiving units are supposed to be identical through the desired operating frequency range. However, in reality these receiving units will be different and the inverse filter can be used to generate better matched inputs. These inputs can be used to find the AOA information.

For the simulation study, the two filters are arbitrarily chosen as a four-tap Butterworth filter and a seven-tap elliptic filter. In order to minimize the edge effect of the filter, the filter passband is selected from 121 to 1160 MHz, which is 40 MHz wider than the desired range of 141 to 1140 MHz. The sampling frequency is kept at 2560 MHz and the impulse responses of the filter are obtained from a chirp signal from 0 to 1280 MHz with the initial phase of $\pi/2$. With these conditions, as shown in Section 16.6, the impulse responses generated are close to the results obtained from an impulse in the time domain.

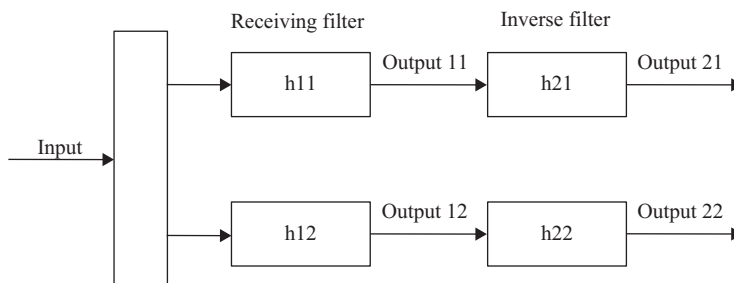


Figure 16.8 Approach to compare input signals.

16.8 Results from Recovered Signals

For the following study, the input signal is a sine wave with a fixed frequency and random initial phase. First, the results of two specific frequencies are shown. Figure 16.9 shows the input frequency at 141 MHz, the lower limit of the input. In this plot, the output data from 76 to 100 points are arbitrarily chosen to avoid the transient effect at the beginning of the filter outputs. Figure 16.9(a) shows the outputs from the receiving filters (h_{11} and h_{12}), which are outputs 11 and 12 in Figure 16.8. These two signals are not well matched. Figure 16.9(b) shows the outputs from the inverse filters, or outputs 21 and 22, and the two signals are better matched. These results show that the inverse filters do correct the different outputs to a better match.

Figure 16.10 shows the input frequency at 640 MHz, the center of the input bandwidth. Figure 16.10(a) shows that the two signals from the receiving filters are well matched. Figure 16.10(b) shows the outputs from the inverse filters. These results are slightly inferior to the results in Figure 16.10(a). Thus, under this special frequency, the inverse filters made the outputs slightly worse.

It is interesting to evaluate the inverse filter effect over the input frequency range. The criterion is similar to the discussions in Section 16.4. In this study, the four-tap Butterworth filter is arbitrarily chosen as the reference filter. The difference between the output responses is obtained and the result is squared and designated as E_f . The response of the Butterworth

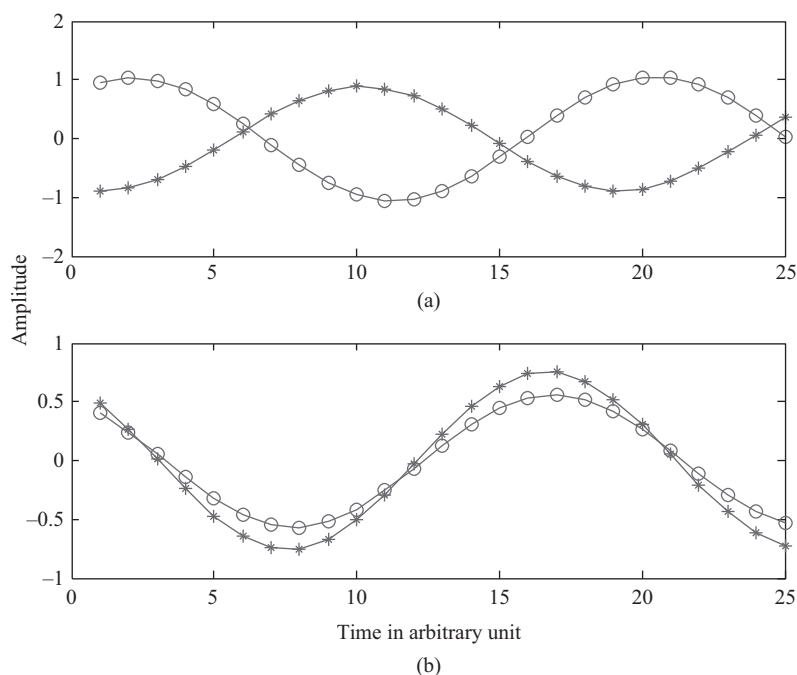


Figure 16.9 Input frequency at 141 MHz: (a) output from input filters and (b) output from inverse filters.

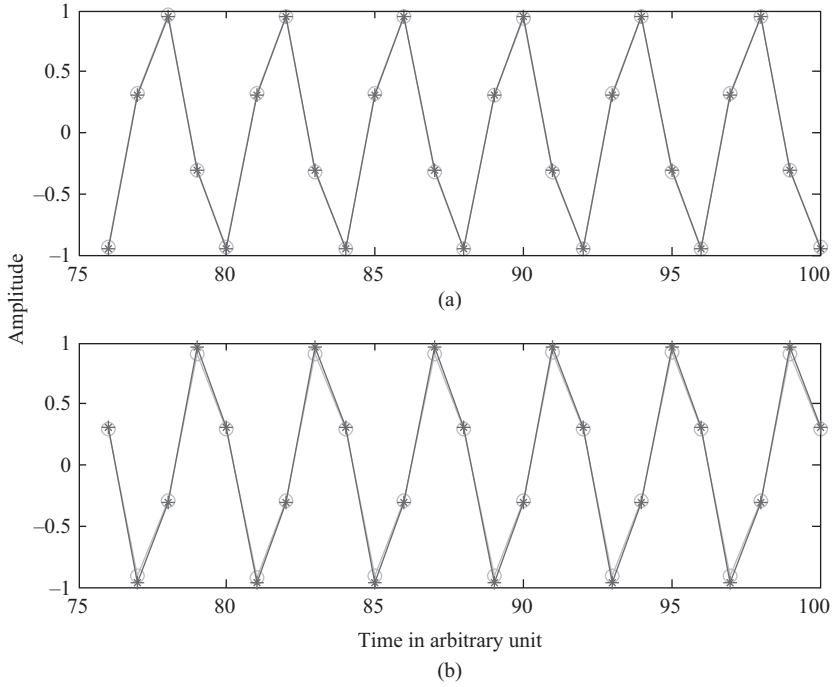


Figure 16.10 Input frequency at 640 MHz: (a) output from input filters and (b) output from inverse filters.

filter is also squared and designated as T_B . The relative error is defined as E_f divided by T_B . These operations are shown in the following equation:

$$\begin{aligned}\Delta x(n) &= x_1(n) - x_2(n) \\ E_f &= \sum_{n=76}^{256} \Delta x(n)^2 \\ T_B &= \sum_{n=76}^{256} x_1(n)^2 \\ E_B &= \frac{E_f}{T_B}.\end{aligned}\tag{16.7}$$

These equations are almost identical to (16.6); the only difference is that the input data are from filter outputs. The data used in this comparison are chosen from 76 to 256 points to avoid the transient effect in the leading edge. In order to reduce confusion with (16.6), new symbols are used in (16.7). The filter outputs $x_1(n)$ can represent either the receiving filter output or the inverse filter outputs of the Butterworth filter path and $x_2(n)$ can represent the outputs from the elliptic filter or the following inverse filters.

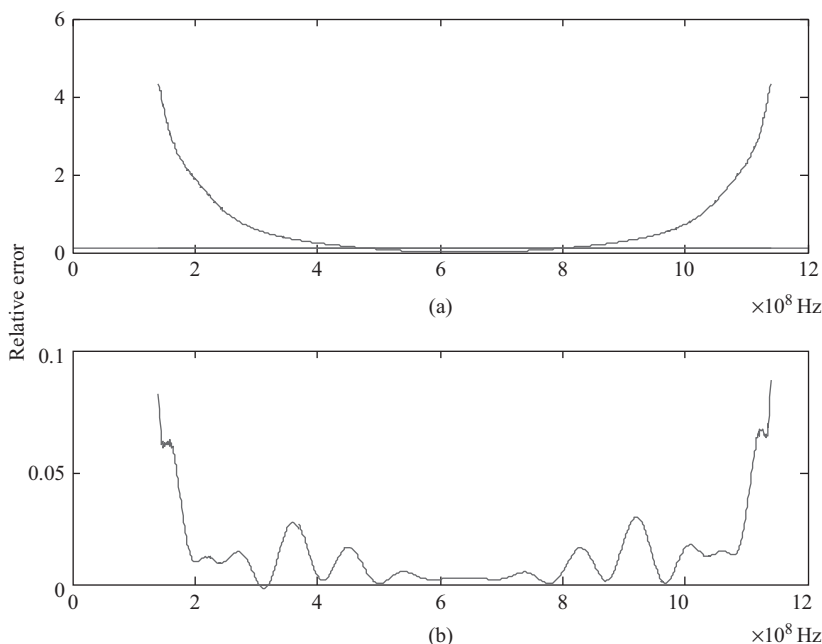


Figure 16.11 Relative errors versus input frequency: (a) input filter output and (b) inverse filter output.

For the simulation study, the input frequency changes from 141 to 1140 MHz with random initial phase. The input frequency changes in 1-MHz steps. At each input frequency, the relative error is calculated. These relative errors are shown in Figure 16.11. Figure 16.11(a) shows the results from the output of the two receiving filters, and the relative errors are quite large. Only the errors in the center frequency range of the filter are small. Figure 16.11(b) shows the results from the inverse filters, and all the relative errors are less than 0.1. A horizontal line is drawn in Figure 16.11(a) with the value of 0.1 to match the maximum scale in Figure 16.11(b). The relative errors in Figure 16.11(a) are less than 0.1 when the input frequency is from about 500 to 800 MHz.

From these simulations, it shows that when a signal passes through different filters, the outputs do not match, but after the inverse filters they are better matched. For AOA measurement, the phases and amplitudes at the antennas should be compared; however, the measured data are at the outputs of the receivers. The mismatch between the two channels will cause AOA errors. After the inverse filters, the inputs are better matched.

16.9 Fractional Time Delay [8, 9]

Theoretically, if a constant delay τ is added to one channel of the AOA measurement process shown in Figure 16.8 to match the phase of the two outputs, the AOA can be obtained from

the delay time. This measurement is frequency independent, thus it is a wideband approach. If an adjustable analog delay time can be introduced, this method seems promising to deliver the right results. But an adjustable analog delay time component is difficult to fabricate. Fortunately a digital approach can be used to obtain similar results.

Since the input signal is digitized at the unit of the sampling time t_s , the delay time in multiples of the sampling time is straightforward. It is more interesting if the delay time τ can be changed in a somewhat continuous manner and less than t_s . The delay time τ is referred to as the fractional delay time if it is less than t_s . The following discussion is based on Laakso *et al.* [8].

The purpose of the following discussion is to find $x(t - \tau)$ from the original signal $x(t)$. The Fourier transform of a signal $x(t)$ delayed by τ can be written as

$$\int_{-\infty}^{\infty} x(t - \tau) e^{-j2\pi ft} dt = e^{-j2\pi f\tau} X(f) \equiv H_d(f) X(f)$$

where

$$H_d(f) = e^{-j2\pi f\tau}. \quad (16.8)$$

The time domain delayed signal $x(t - \tau)$ can be written as the convolution of $h_d(t)$ and $x(t)$ in the time domain:

$$x(t - \tau) = h_d(t) \otimes x(t)$$

where

$$h_d(t) = \int_{-B/2}^{B/2} e^{-j2\pi f\tau} e^{j2\pi ft} df = \frac{e^{j2\pi f(t-\tau)}}{j2\pi(t-\tau)} \Big|_{-B/2}^{B/2} = \frac{2j \sin(\pi B(t-\tau))}{j2\pi(t-\tau)} = B \operatorname{sinc}(2\pi B(t-\tau)), \quad (16.9)$$

where B is the bandwidth of the system and h_d is the transfer function representing the delay time. From this equation one can find that a sinc function convolved with the input signal can produce the desired time domain shift. The sinc function is shown in Figure 16.12. In this figure the circles show the sinc function without delay and the square shows the delayed case. If there is no delay, the sinc function equals one at $x = 0$ and the rest of values are all zero. If there is a delay, the sinc function has many nonzero values.

Consider the two-channel system illustrated in Figure 16.8. If the inverse filters function properly and the incoming signal reaches two receivers at the same time, output 21 and output 22 should be identical. If the signal reaches two receivers at different times due to a nonhorizontal angle of arrival, output 21 and output 22 will be different. If a proper delay time is added in one of the two channels to make output 21 equal output 22, this delay time can then be used to calculate the AOA.

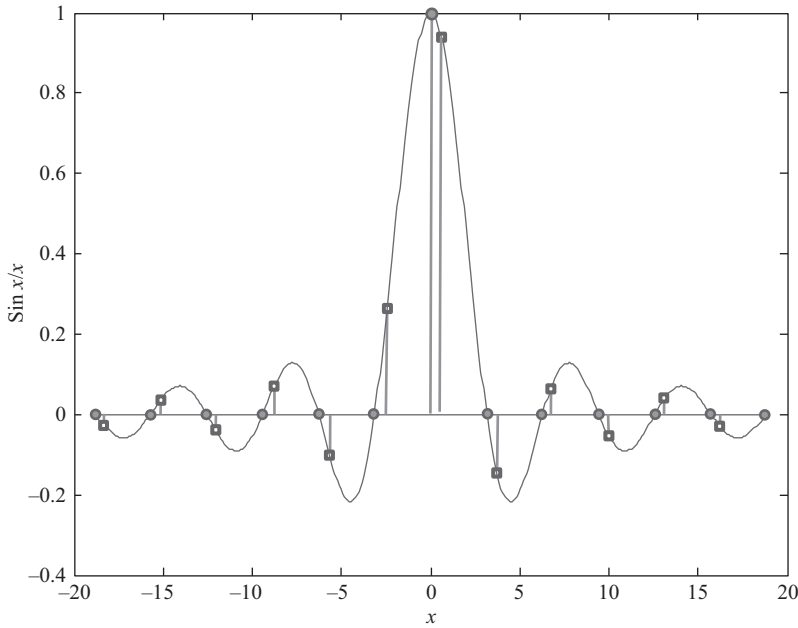


Figure 16.12 Sinc function with and without delays.

16.10 Phase Adjustment

In a receiver system, the input signal is often converted into the desired bandwidth before digitization. For example, if the frequency of interest is from 2 to 18 GHz and the bandwidth of the baseband receiver is from 2 to 3 GHz, the input signal must be converted into the baseband before digitization. If the input signal is obtained through a downconversion processing before the digitization, the local oscillator (LO) must be taken into consideration. There are two terms causing phase shifts: delay time τ and phase shift of $\omega_0\tau$, where ω_0 is the LO angular frequency. The delay time τ is used to determine the AOA. These two values are included in the digitized data. If the input is a sinusoidal function, the operation can be written as $x(t) = \cos(2\pi ft)$ and the delayed value can be obtained from (16.9) through the convolution of $h(t)$.

If the phase angle of $2\pi f_0\tau$ is incorporated in the input signal, the result is

$$\cos(2\pi f(t - \tau) + 2\pi f_0\tau) = \cos(2\pi f(t - \tau))\cos(2\pi f_0\tau) - \sin(2\pi f(t - \tau))\sin(2\pi f_0\tau), \quad (16.10)$$

where f_0 is the mixer frequency. The overall operation can be divided into three steps:

1. Find $\cos(2\pi f(t - \tau))$ by convolving $\cos(2\pi ft)$ with filter $h(t)$ as described in Section 16.9.
2. Introduce a 90-deg phase shift to $\cos(2\pi f(t - \tau))$ and generate $\sin(2\pi f(t - \tau))$ through the Hilbert transform stated in Sections 6.8 and 6.9.
3. Use the operation in (16.10) to obtain the desired results.

These operations are complicated and time consuming. Especially after the convolution, the output must be used to generate a signal with a 90-deg phase shift. A simplified approach is presented in the next section.

16.11 A Different Approach to Introduce the Phase Angle [10]

David Lin of the Air Force Research Laboratory (AFRL) suggested an approach [10] to simplify the operation described in Section 16.10. The goals are

1. Eliminate the generation of the quadrature (90-deg phase shifted) signal (saving the circuitry, extra processing required, and response time);
2. Generate a table containing the information on the impulse filter from a single set of observation data (reducing processing time from that required to gather and average multiple data sets); and
3. Remain independent of input frequency. The table will be a function of AOA and the input frequency band, which is related to the local oscillator frequency f_o .

The size of the table depends on the number of receivers, the bandwidth of the receiver, the total input band, and the AOA resolution. Suppose that the input bandwidth of the receiver system is from 1 to 18 GHz and the instantaneous bandwidth of each receiver is 1000 MHz, then the AOA coverage is from -60 to $+60$ deg with a resolution of 1 deg. Assume also that there are 16 evenly spaced antennas and receivers, thus the total number of filters is 32,912 ($17 \times 121 \times 16$). In order to look into a certain direction, only 17 impulse filters will be used. These 17 filters are selected based on the input band and input AOA.

Figure 16.13 shows a system with only two channels, each of which contains an antenna and a receiver. In order to simplify the discussion, let us refer to them as channel 1 and channel 2. There are two ways to calibrate the system. The first can be considered a straightforward approach. In this approach the input AOA is changed and at each AOA two output data sets are collected. As discussed in Section 16.3, the input is a chirp signal covering the entire frequency range of 1000 MHz. These two inputs are labeled input 1 and input 2 in Figure 16.13. The digitized data are labeled output 1 and output 2. These data are convolved with the locally generated signals, labeled $x_{o1}(n)$ and $x_{o2}(n)$. If the input data are obtained from different AOA information, the two locally generated signals are identical and represented as $x_o(n)$. Although this approach is simple in principle, the data collection is tedious. The experiment is usually carried out in an anechoic chamber and many data sets must be collected. If there are 16 receivers and 121 input AOA data to cover -60 to 60 deg in 1-deg resolution, the total of the data set to be collected will be 1936 (16×121). Lin's approach only measures 16 data sets measured at the boresight. In the scenario, data are usually easier to collect and signals reach all receivers simultaneously. The rest of the data set of the other AOA can be derived analytically. The idea is to artificially introduce the time delay in the local signals $x_{o1}(n)$ and $x_{o2}(n)$. Using this approach, the two input data are identical, represented by $x(n)$ from the boresight, and the two local signals are different.

Before further discussion on this approach, it is important to clarify some concepts. First, let us assume that the inputs $x(n)$ are from boresight and the two impulse responses obtained are $h_1(n)$ and $h_2(n)$. Now let us change the inputs from a certain AOA value and the inputs

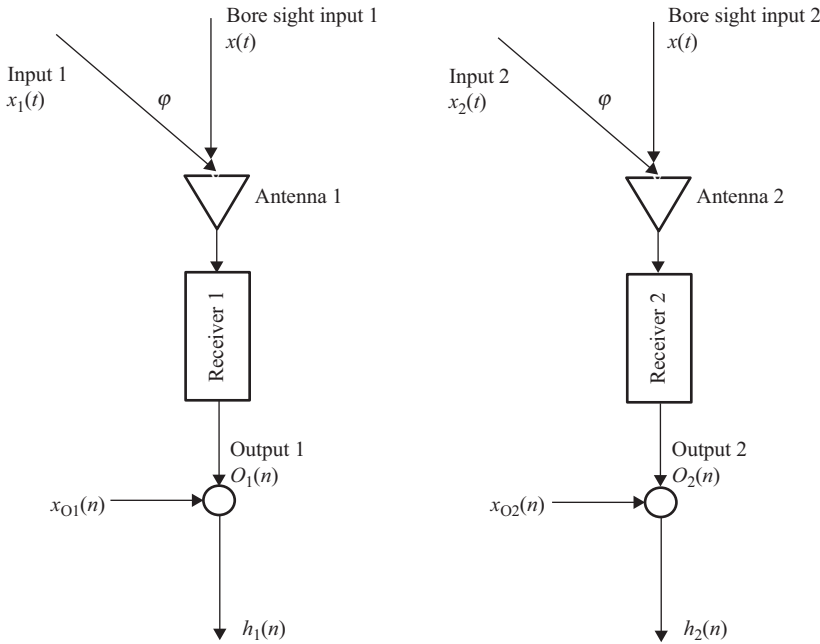


Figure 16.13 A two-channel AOA system.

are labeled $x_1(n)$ and $x_2(n)$. If channel 1 is used as the reference channel, the two impulse functions obtained are $h_{1\varphi}(n)$ and $h_{2\varphi}(n)$, where φ represents the input AOA. Under this condition, $h_1(n) = h_{1\varphi}(n)$. Since channel 1 is used as the reference, there is no time delay compared to itself, and the impulse response does not change. However, $h_2(n) \neq h_{2\varphi}(n)$, because there is a time delay in channel 2 with respect to channel 1.

An important step is to prove that the delay time can be introduced in the local signal rather than obtained from the input signal. The next section proves the results. Since $h_1(n)$ is independent of the input AOA, this proof will be performed only on the second channel.

16.12 Switching the Delay from the Input to the Local Signal

First, let us define the chirp signal in the baseband as

$$\cos(2\pi f(n\Delta t)) = \cos(2\pi f_{start} \cdot n\Delta t + \pi\alpha \cdot (n\Delta t)^2) = \cos(2\pi k_d n + \pi a n^2),$$

where

$$f(n\Delta t) = f_{start} \cdot n\Delta t + \frac{1}{2} \alpha \cdot (n\Delta t)^2, \quad k_d = f_{start} \cdot \Delta t, \quad \text{and} \quad a = \alpha \cdot \Delta t^2. \quad (16.11)$$

The variable f_{start} is the starting frequency of the baseband chirp signal and a is related to the chirp rate of the input signal.

For the input at a certain AOA, the digitized output is $O_{2\varphi}(n)$ and local signal is $x_o(n)$ and they are

$$\begin{aligned} O_{2\varphi}(n) &= \cos\left(2\pi k_d(n - \tau) + \pi a(n - \tau)^2\right) \otimes h_2(n) = \cos\left(\pi a(n - \tau)^2\right) \otimes h_2(n) \\ x_o(n) &= \cos\left(2\pi k_d(N - n) + \pi a(N - n)^2\right) = \cos\left(\pi a(N - n)^2\right) \\ h_{2\varphi}(n) &= O_{2\varphi}(n) \otimes x_o(n), \end{aligned} \quad (16.12)$$

where τ is the delay time between channel 1 and 2, a is related to the chirp rate of the input signal, $h_2(n)$ is the impulse function of channel 2 with boresight input, N is the total data length processed, and k_d is related to the starting frequency of the baseband chirp signal. Since the starting frequency of the baseband chirp signal, f_{start} , is very close to zero, the relation of $k_d = 0$ is used in the above equation.

If the delay time is introduced in the local signal, the results will be

$$\begin{aligned} O_2(n) &= \cos(\pi a n^2) \otimes h_2(n) \\ x_{o2}(n) &= \cos\left(\pi a(N - (n - \tau))^2\right) \\ h_{2\varphi}(n) &= O_2(n) \otimes x_{o2}(n). \end{aligned} \quad (16.13)$$

The approach is to prove that the $h_{2\varphi}(n)$ results obtained from (16.12) and (16.13) are equivalent. Or simply

$$\cos\left(\pi a(n - \tau)^2\right) \otimes \cos\left(\pi a(N - n)^2\right) = \cos(\pi a n^2) \otimes \cos\left(\pi a(N - (n - \tau))^2\right). \quad (16.14)$$

This equation can be written in the frequency domain as

$$\begin{aligned} F_1(k) &= \sum_{n=0}^{N-1} \cos\left(\pi a(n - \tau)^2\right) e^{-\frac{j2\pi nk}{N}} \\ &= \frac{1}{2} \sum_{n=0}^{N-1} \left(e^{j(\pi a(n - \tau)^2)} + e^{-j(\pi a(n - \tau)^2)} \right) e^{-\frac{j2\pi nk}{N}} \\ &= G_1(k) + G_2(k) \\ F_2(k) &= \sum_{n=0}^{N-1} \cos(\pi a n^2) e^{-\frac{j2\pi nk}{N}} \\ &= \frac{1}{2} \sum_{n=0}^{N-1} \left(e^{j(\pi a n^2)} + e^{-j(\pi a n^2)} \right) e^{-\frac{j2\pi nk}{N}} \\ &= K_1(k) + K_2(k) \end{aligned}$$

where

$$\begin{aligned}
 G_1(k) &\equiv \frac{1}{2} \sum_{n=0}^{N-1} e^{j(\pi a(n-\tau)^2)} e^{-\frac{j2\pi nk}{N}} \\
 G_2(k) &\equiv \frac{1}{2} \sum_{n=0}^{N-1} e^{-j(\pi a(n-\tau)^2)} e^{-\frac{j2\pi nk}{N}} \\
 K_1(k) &\equiv \frac{1}{2} \sum_{n=0}^{N-1} e^{j(\pi a n^2)} e^{-\frac{j2\pi nk}{N}} \\
 K_2(k) &\equiv \frac{1}{2} \sum_{n=0}^{N-1} e^{-j(\pi a n^2)} e^{-\frac{j2\pi nk}{N}}
 \end{aligned} \tag{16.15}$$

Based on Fourier Transform's time shifting property,

$$\begin{aligned}
 G_1(k) &= K_1(k) e^{-j2\pi k\tau} \\
 G_2(k) &= K_2(k) e^{-j2\pi k\tau}.
 \end{aligned}$$

With these definitions, the left side of (16.14) can be written as

$$\begin{aligned}
 \cos(\pi a(n-\tau)^2) \otimes \cos(\pi a(N-n)^2) &\Rightarrow (G_1(k) + G_2(k))(K_1(k) + K_2(k))^* \\
 &= G_1(k)K_1^*(k) + G_1(k)K_2^*(k) + G_2(k)K_1^*(k) + G_2(k)K_2^*(k) \\
 &= K_1(k)K_1^*(k)e^{-j2\pi k\tau} + K_1(k)K_2^*(k)e^{-j2\pi k\tau} + K_2(k)K_1^*(k)e^{-j2\pi k\tau} + K_2(k)K_2^*(k)e^{-j2\pi k\tau}.
 \end{aligned} \tag{16.16}$$

The right side of (16.14) can be written as

$$\begin{aligned}
 \cos(\pi a n^2) \otimes \cos(\pi a(N-(n-\tau))^2) &\Rightarrow (K_1(k) + K_2(k))(K_1^*(k)e^{-2\pi k\tau} + K_2^*(k)e^{-2\pi k\tau}) \\
 &= K_1(k)K_1^*(k)e^{-j2\pi k\tau} + K_1(k)K_2^*(k)e^{-j2\pi k\tau} + K_2(k)K_1^*(k)e^{-j2\pi k\tau} + K_2(k)K_2^*(k)e^{-j2\pi k\tau}.
 \end{aligned} \tag{16.17}$$

Since the results of (16.16) and (16.17) are the same, we thus prove (16.14).

16.13 Generate Input Data from One Set of Input Measurements

The impulse response $h_{2\varphi}(n)$ of the second channel can be obtained from (16.13) by inserting different τ values in the local signal. A modified approach can be derived from (16.13), which can be written as

$$h_{2\varphi}(n) = O_2(n) \otimes x_{o2}(n) = \cos(\pi a n^2) \otimes h_2(n) \otimes \cos(\pi a(N-(n-\tau))^2). \tag{16.18}$$

The last factor in (16.18), involving delay time, can be obtained by convolving it with a sinc function as shown in (16.9). The desired impulse function $h_{2\varphi}$ can thus be obtained from h_2 , the impulse response at the boresight angle. This impulse function can be used to convolve with the input signal to determine the AOA information. If a calibration table is desired, this impulse function can be obtained through frequency domain calculation.

16.14 Conclusions

In this chapter, the concept of the time reversal filter is presented. The approach is applicable in a laboratory environment for calibration purposes. Using special generated signals, experimental data can be generated from the digitized output of an antenna receiver system. These data can be used to obtain the taps for the time reversal filter. The input conditions affecting the inverse filter performance are also presented. Finally, it is illustrated that a signal passing through several different signal paths can be recovered to their input conditions. In other words, the system imbalance among different channels can be mitigated. Since the time reversal filter is derived from the system's impulse response, it can cover a wide frequency range.

If there is frequency conversion in the receiver, the problem can be rather complicated. For a wideband system, the delay time causes a phase shift, which is related to the frequency of the oscillator used in the frequency conversion process. This phase shift can be added in the local signal used for convolution with the input signal. Using this approach, only one set of input data is needed to produce all the impulse functions from different input AOAs. Since the approach is wideband, the results are frequency independent. Both the time domain and frequency domain can be used to produce the impulse function.

It is concluded that the time reversal filter concept is applicable to AOA measurement to cancel the difference caused by characteristic mismatch among different channels. The most appealing feature is that a chirp input signal can generate inverse filters covering a wide input bandwidth. Previously the traditional calibration procedure took data at different input frequencies and compensated the difference in the frequency domain. In order to better compensate the difference over the input frequency band, fine frequency resolution is needed in the measurements. This is a time-consuming procedure. It appears that the time inverse filter is a simpler approach performed in the time domain.

References

- [1] Kuperman WA, Hodgkiss WS, Song HC, Akal T, Ferla C, Jackson D. 'Phase conjugation in the ocean: experimental demonstration of an acoustic time-reversal mirror'. *Journal of the Acoustical Society of America* 1998;**103**(1):25–40.
- [2] Fink M. 'Time-reversed acoustics'. *Scientific American* 1999;**281**(5):91–97.
- [3] Edelmann GF, Akal T, Hodgkiss WS, Kim S, Kuperman WA, Song HC. 'An initial demonstration of underwater acoustic communication using time reversal'. *IEEE Journal of Oceanic Engineering* 2002;**27**(3):602–609.
- [4] Yang TC. 'Temporal resolutions of time-reversal and passive-phase conjugation for underwater acoustic communications'. *IEEE Journal of Oceanic Engineering* 2003;**28**(2):229–245.

- [5] Higley WJ, Roux P, Kuperman WA. 'Relationship between time reversal and linear equalization in digital communications'. *Journal of the Acoustical Society of America* 2006;**120**(35):35–37.
- [6] Qiu RC, Zhou C, Guo N, Zhang JQ. 'Time reversal with MISO for ultrawideband communications: experimental results'. *IEEE Antennas and Wireless Propagation Letters* 2006;**5**(1): 269–273.
- [7] Liou LL, Lin DM, Longbrake M, Buxa P, McCann J, Dalrymple T, Tsui JB, Qiu R, Hu Z, Guo N. 'Digital wideband phased array calibration and beamforming using time reversal technique'. *Proceedings of 2010 IEEE International Symposium on Phased Array Systems and Technology (ARRAY)*. New York: IEEE; 2010:261–266.
- [8] Laakso TI, Valimäki V, Karjalainen M, Laine UK. 'Splitting the unit delay [FIR/all pass filters design]'. *IEEE Signal Processing Magazine* 1996;**13**(1):30–60.
- [9] Liou L, Lin D. Air Force Research Laboratory. Private discussions.
- [10] Lin D. Air Force Research Laboratory. Private discussions.

Multiple Fast Fourier Transform Electronic Warfare Receiver

17.1 Introduction

The performance of a fast Fourier transform (FFT)-based electronic warfare (EW) receiver depends on the size of the FFT. An EW receiver using a long FFT size has a higher sensitivity and finer frequency resolution [1]. However, its time domain resolution suffers, thus creating a less accurate estimate on the signal's time domain characteristics, such as time of arrival (TOA) and pulse width (PW). This is especially true for short signals. On the other hand, an EW receiver using a short FFT size can better determine the signal's PW and TOA but might miss long but weak signals and have coarse frequency resolution. One way to solve this issue is to create an EW receiver with multiple FFT sizes and determine signal characteristics based on the outcomes of the FFT with different frame sizes. However, it is not a trivial task to combine signal characteristic estimates generated with different FFT frame sizes and this chapter is dedicated to addressing this task. In this chapter an EW receiver encoder algorithm based on multiple FFT frame sizes is presented. The material presented in this chapter is based on our earlier works [2].

17.2 EW Receiver Diagram

An example will be used to illustrate the design concepts of an EW receiver with multiple FFT sizes. The diagram of the EW receiver based on multiple FFT sizes is illustrated in Figure 17.1. An amplifier is used before digitization so that the range of the signal amplitude will match the full range of the analog-to-digital converter (ADC). The digitized data are multiplied with a 64-, 256-, or 1024-point Blackman window before the corresponding FFT is computed. The results of the three FFTs are passed to three encoders that generate pulse descriptive words (PDWs) consisting of signal carrier frequency, signal power, signal PW, and signal TOA. The angle of arrival (AOA) is not considered in this study. The three PDWs generated by three encoders are then compiled into a single PDW by a PDW combiner. The simulation setting used in this study is listed below:

- ADC sampling frequency: 1.33 GHz
- ADC number of bits: 10

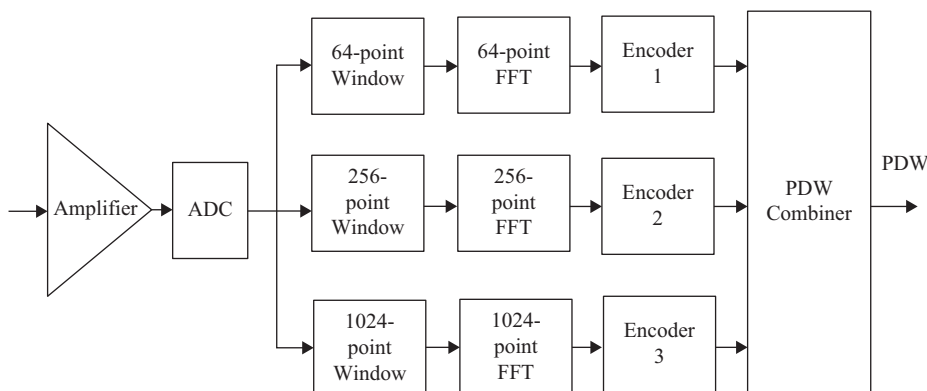


Figure 17.1 Multiple FFT EW receiver diagram.

- Amplifier gain: 12 dB
- Amplifier noise figure: 10 dB
- FFT frame size: 64, 256, or 1024 points

It should be noted that the simulation presented in this chapter focuses on a combination of encoder results and the characteristics of the amplifier and the ADC are not critical.

The minimum FFT frame size (64 points) is determined by the EW receiver's minimum PW requirement of detectable signals. The minimum signal PW requirement of the EW receiver under consideration is 150 ns. To avoid “rabbit-ear effects” [3, 4], the signal needs to be detected in at least two of three consecutive FFT frames. We will explain this detection criterion later. Under a 1.33-GHz sampling rate, the duration of a 64-point frame is 48.12 ns. Therefore the 64-point FFT is chosen. Other FFT frame sizes, 256 points and 1024 points, are chosen arbitrarily. The EW receiver might use more than three FFT frame sizes, but the principle of combining PDWs described in this chapter can still apply.

17.3 Encoder Description

The encoder generates the following signal characteristics: signal carrier frequency, signal power, PW, and TOA. The encoder has a signal characteristics buffer to store present signal characteristics; the PDW generation procedure is summarized below:

1. Get the FFT of the windowed data from the FFT processor. The Blackman window length is either 64, 256, or 1024 points.
2. Determine the peak positions of the FFT results. There might be more than one peak.
3. Estimate peak carrier frequencies and signal power based on the FFT results.
4. If the peak power is above the threshold, the encoder saves the signal characteristics (signal carrier frequency, signal power, and FFT bin number of the detected signal) in the signal characteristics buffer. The FFT bin number can be used as a coarse frequency estimate and the signal carrier frequency estimate is generated using an interpolation method described in Section 13.3. The FFT bin number is used to determine the signal's existence in this receiver design, and its usage will be described in the following paragraphs.

Once the signal characteristics are saved the following procedures are followed.

1. If the signal is detected for the first time, keep its signal characteristics in the signal characteristics buffer.
2. If the signal is detected at the same FFT bin or one of its two adjacent bins in the previous frame or the one before the previous frame, then the encoder assumes that the signal is an existing signal that is still present. Its characteristics in the signal characteristics will be updated.
3. The encoder estimates the signal carrier frequency and signal power by averaging the signal carrier frequency and signal power estimations in all “detected frames.” If there are three or more detected frames, the power estimations of the first and last frames are not used in averaging. When there are only two detected frames, the power estimation of the last frame is used as the signal power estimate because of its easy implementation. It should be noted that the receiver is designed to detect the signal whose PW is at least three times a frame duration. When there are only two detected frames, the signal power estimate might be inaccurate.
4. A signal needs to be detected in at least two frames out of three consecutive frames; otherwise it is considered a false alarm and its characteristics are deleted from the signal characteristics buffer. The maximum allowed number of “undetected frames” between “detected frames” is one. This is why the minimum PW is about 150 ns, or three windows long, to ensure a two-frame detection.
5. For a detected signal, after it is lost in two consecutive frames, the encoder considers the signal disappeared and outputs its PDW, including signal carrier frequency, signal power, TOA, and PW, to a PDW buffer in the main program. The signal characteristics are then deleted from the signal characteristics buffer.

It should be noted that the PDW is only generated for the signal that is detected in more than two frames out of three consecutive frames to avoid a rabbit-ear effect [3]. A rabbit ear usually happens in the leading and falling edge of the signals when signals do not occupy the whole FFT frame and the signal energy thus spreads into multiple frequency bins. When a strong signal appears/disappears in the middle of a weak signal, the weak signal might not be detected in the FFT frame where the strong signal appears/disappears due to the spreading energy from the strong signal, and the spreading energy might create a false alarm. As a result, the receiver might see an increased false alarm rate or miss other signals due to the rabbit-ear effect. The proposed encoding method can effectively mitigate these issues caused by the rabbit-ear effect.

17.4 PDW Combiner Description

The multiple-FFT receiver uses multiple frame sizes to analyze data. The encoder with long frames generates PDWs with better frequency resolution and can detect long weak signals, but its TOA and PW estimations might be inaccurate. The encoder with short frames has better TOA and PW estimations, but it has difficulty in detecting weak signals. By using encoders with long and short frames simultaneously, we hope to combine their advantages. The PDW combiner is used to combine results from

encoders. The PDW combiner has a PDW buffer to temporarily store the PDW for each encoder.

Two important issues need to be solved while combining PDWs: (1) how does the PDW combiner determine if PDWs generated by different encoders belong to the same signal and (2) how does the PDW combiner combine PDWs generated by different encoders? To address these two issues, the PDW combiner needs to perform two tasks: PDW association and PDW combination.

To accomplish these two tasks, several thresholds will be used:

- **Frequency difference threshold.** This threshold is used to determine if signals detected by two encoders with different frame sizes are the same signal. We used the frequency resolution of a short FFT frame as the threshold. Since frequency measurement from the long frames will have finer resolution, it needs only to show that the long-frame PDW frequency falls within the resolution range of the short-frame PDW.
- **PW difference threshold.** This threshold is used to determine if the difference between PW estimations of two encoders is too large. In principle, the short FFT frame generates better PW estimation. However, if the difference between PW estimations is above this threshold, additional investigation is necessary to generate a reliable PW estimation. In our simulation, this threshold is twice the duration of the shorter FFT frame.
- **PW threshold.** This threshold is used to determine if the signal's PW is too short so the signal power estimations of the long frame are unreliable. When the PW is below this threshold, the long FFT frame might generate inaccurate signal power estimation since the signal might not fill up a whole long FFT frame. In our simulation, this threshold is three times the duration of the longer FFT frame.
- **Signal power estimation difference threshold.** This threshold is used to determine if the difference between signal power estimations of two encoders is too large. In principle, the long FFT frame should generate more accurate signal power estimation. However, if the difference between signal power estimations is above this threshold, additional investigation is conducted to determine which signal power estimation is more reliable. A detailed description is provided in the next section. In our simulation, this threshold is 15 dB.
- **TOA estimation difference threshold.** This threshold is used to determine if the difference between the TOA estimations of two encoders is too large. In principle, the short FFT frame generates a better TOA estimation. However, if the difference between TOA estimations is above this threshold, more investigation is necessary to generate reliable TOA estimation. In our simulation, this threshold is twice the duration of the shorter FFT frame.
- **Time of departure (TOD) estimation difference threshold.** The TOD of a signal is defined as the sum of its TOA and PW. This threshold is used to determine if the difference between the TOD estimations of two encoders is too large. In principle, the short FFT frame generates better TOD estimations. However, if the difference between the TOD estimations is above this threshold, more investigation is necessary to generate reliable TOD estimation. In our simulation, this threshold is twice the duration of the shorter FFT frame.

The threshold values used in our receiver program have been determined empirically.

17.4.1 PDW Association

The procedure of the PDW combiner associating PDWs from encoders, i.e., the PDW association procedure is summarized below:

1. After one encoder generates a PDW after the signal disappears, check if other encoders have detected a signal whose carrier frequency difference is less than the frequency difference threshold but the signal is still present. If so, then put the PDW in a PDW buffer. Otherwise, search PDWs in the PDW buffer of other encoders.
2. If the carrier frequency difference between PDWs from different encoders is smaller than a predefined “frequency difference threshold,” then these PDWs are considered to refer to the same signal.
3. If the PDW generated by the long frame is associated with more than one PDW generated by the short frame and the PDWs generated by the short frame do not overlap in time, then PDWs generated by the short frame are considered to associate with the same signal. A new PDW associated with the short frame is created by “connecting” these non-overlapping PDWs and these nonoverlapping PDWs are deleted from the PDW buffer. (Note: In some scenarios, a long signal might be detected as two or more disjointed short signals. For example, when the signal power is near the receiver detection threshold, the signal might be detected in some frames but missed in other frames. The other scenario is that a signal might be missed by the receiver when another signal with higher power arrives or departs in the middle of the signal due to the rabbit-ear effect described earlier.)
4. If two or more PDWs generated by the long frame are associated with a PDW generated by the short frame, the PDW generated by the short frame is considered to associate with multiple signals and is discarded. The encoder with the short frame might not be able to distinguish signals whose frequencies are close, and these signals can be distinguished by the encoder with a long frame.

Once the PDWs from different encoders are determined to belong to the same signal, the next step is to generate an accurate PDW based on PDWs from different encoders.

17.4.2 PDW Combination

When combining PDWs generated by encoders with different frame sizes, we take results from two encoders into account each time. If a signal is detected by 64-, 256-, and 1024-point encoders, then the PDW generated by the 256- and 1024-point encoders are combined first and the resulting PDW is then combined with the PDW generated by the 64-point PDW encoder. Therefore the PDW combination process described here only applies to PDWs from two encoders for each step in the process.

In principle, an encoder with a longer frame should have better signal carrier frequency and power estimations and an encoder with a shorter frame should have better PW and TOA estimations. However, for a long weak signal, the short frame might not detect the whole signal. Moreover, for a strong short signal, it might be detected with the long frame and results in an inaccurate PDW. As a result, special care needs to be taken while the receiver conducts the PDW combination task. The flow charts of combining estimations of signal carrier frequency, TOA, PW, and signal power with different FFT frame sizes are given in Figures 17.2 to 17.5, respectively, and the detailed PDW combination flow chart description is provided below.

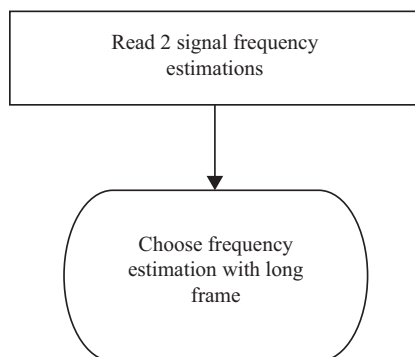


Figure 17.2 Flow chart of combining frequency estimations of two encoders [2].

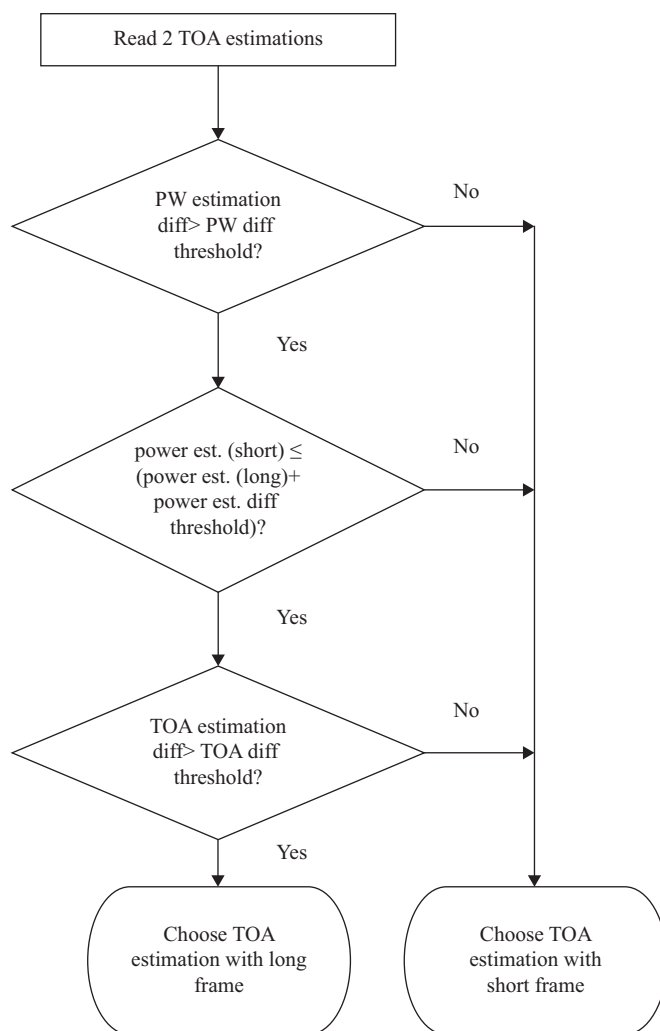


Figure 17.3 Flow chart of combining TOA estimations of two encoders [2].

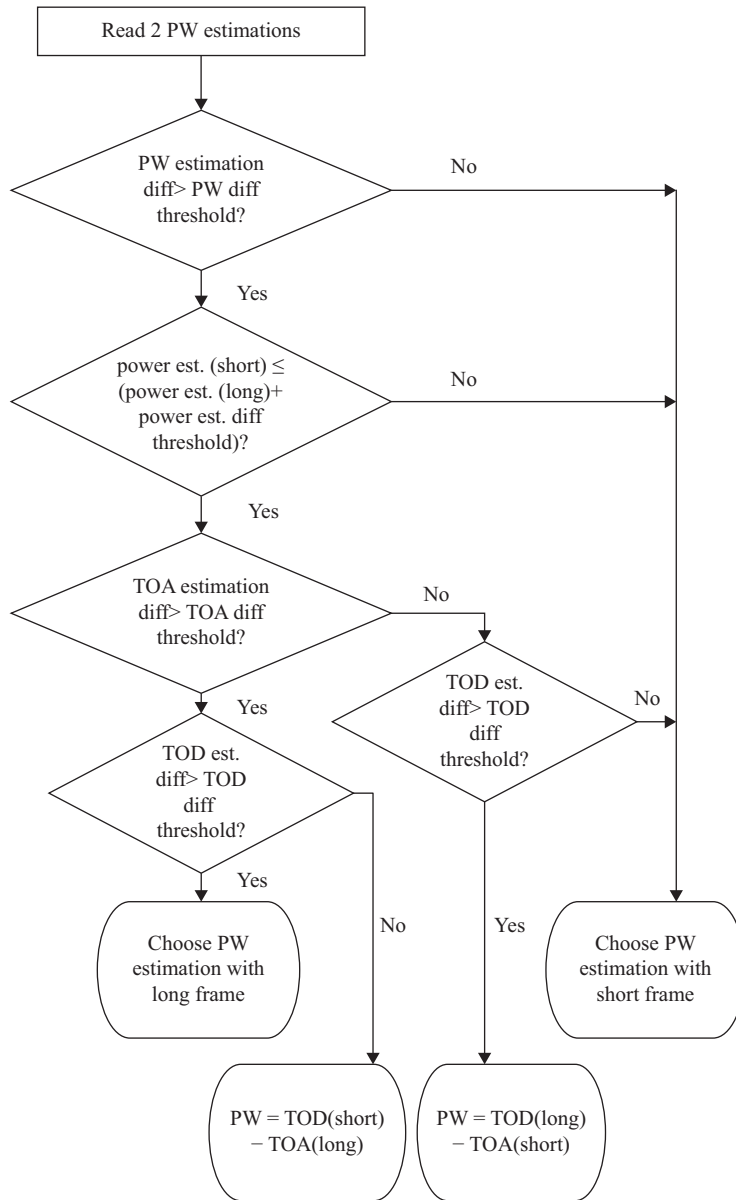


Figure 17.4 Flow chart of combining PW estimations of two encoders [2].

Signal Carrier Frequency Determination

Choose the frequency estimation of the longer frame.

Signal TOA Determination

In principle, choose the TOA estimation of the shorter frame. However, choose the TOA estimation of the longer frame, if *all* of the following conditions are satisfied: (1) the PW

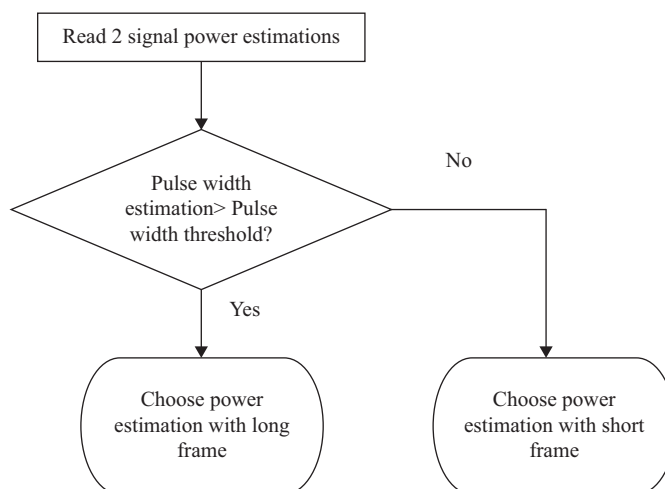


Figure 17.5 Flow chart of combining signal power estimations of two encoders [2].

estimation difference between two PDWs is larger than the PW difference threshold, (2) the signal power estimation of the shorter frame is not larger than the signal power estimation of the longer frame by the signal power estimation difference threshold, and (3) the TOA estimation difference between two PDWs is larger than the TOA estimation difference threshold.

Signal PW Estimation

Among four signal characteristics of PDWs—carrier frequency, power, TOA, and PW—the combination of PW estimates is most complicated. The reason is that PW is defined as the difference between the TOA and TOD ($\text{TOD} = \text{TOA} + \text{PW}$), so the PDW combiner needs to take both TOA and TOD estimates into account. The process of combining PW estimations is described below:

- Choose PW estimation of the longer frame if *all* of the following conditions are satisfied: (1) the PW estimation difference between two encoders is larger than the PW difference threshold, (2) the signal power estimation of the shorter frame is not larger than the signal power estimation of the longer frame by the signal power estimation difference threshold, (3) the TOA estimation difference between two encoders is larger than the TOA estimation difference threshold, and (4) the TOD estimation difference between two encoders is larger than the TOD estimation difference threshold.
- Set the PW estimation as the difference between the TOD estimation of the shorter frame and the TOA estimation of the longer frame if *all* of the following conditions are satisfied: (1) the PW estimation difference between two encoders is larger than the PW difference threshold, (2) the signal power estimation of the shorter frame is not larger than the signal power estimation of the longer frame by the signal power estimation difference threshold, (3) the TOA estimation difference between two encoders is larger than the TOA estimation difference threshold, and (4) the TOD estimation difference between two encoders is less than the TOD estimation difference threshold.

- Set the PW estimation as the difference between the TOD estimation of the longer frame and the TOA estimation of the shorter frame if *all* of the following conditions are satisfied: (1) the PW estimation difference between two encoders is larger than the PW difference threshold, (2) the signal power estimation of the shorter frame is not larger than the signal power estimation of the longer frame by the signal power estimation difference threshold, (3) the TOA estimation difference between two encoders is less than the TOA estimation difference threshold, and (4) the TOD estimation difference between two encoders is larger than the TOD estimation difference threshold.
- Choose the PW estimation of the shorter frame if *any* of the following conditions are satisfied: (1) the PW estimation difference between two encoders is less than the PW difference threshold, (2) the signal power estimation of the shorter frame is larger than the signal power estimation of the longer frame by the signal power estimation difference threshold, or (3) both the TOA and TOD estimation differences between two encoders is less than the TOA estimation difference threshold and TOD estimation difference threshold, respectively.

Signal Power Estimation

If the signal PW estimation is longer than the PW threshold, then use the signal power estimation of the longer frame; otherwise, use the signal power estimation of the shorter frame.

After the PDWs are combined, the PDW combiner outputs the resulting PDW and all of related PDWs are then deleted from the PDW buffers.

17.5 Simulation Results and Discussions

Because the multiple-FFT receiver uses both long and short frames, it can achieve good time and frequency resolution. To demonstrate the advantages of the proposed method, four simulation scenarios are considered: (1) two signals with narrow frequency separation, (2) two signals with short duration, (3) one strong short signal overlaps with a weak long signal, and (4) two signals with a 50-dB power difference. Although the multiple-FFT receiver can detect more than two signals and the detection method is independent of the number of signals, for simplicity only the scenarios with two signals are presented. For the sake of comparison, the PDW generated by the 256-point FFT receiver alone is also included.

Case 1: Two signals with narrow frequency separation

The multiple-FFT receiver can detect two signals with very narrow frequency separation since one of its encoders uses a very long window (1024 points). With the 1.33-GHz sampling rate, the frequency resolution of the 64-, 256-, and 1024-point FFTs are 20.7813 MHz, 5.1953 MHz, and 1.2988 MHz, respectively. With the 1024-point FFT, the multiple-FFT receiver can distinguish two signals whose frequencies are 2.6 MHz (2 FFT bins) apart. We used two signals whose frequency is 3 MHz apart as the test signals. The signal characteristics of two signals and the PDW generated by a multiple-FFT receiver and single-FFT receiver are listed in Table 17.1.

As shown in Table 17.1, the multiple-FFT receiver can separate two test signals and create PDWs with reasonable accuracy. On the other hand, the single-FFT with 256-point frame cannot distinguish these two signals. As a result, the single-FFT only detects one

Table 17.1 Test signal characteristics and estimate of multiple-FFT and single-FFT receivers (case 1).

	Signal 1			Signal 2		
	True value	Multiple-FFT estimate	Single-FFT estimate	True value	Multiple-FFT estimate	Single-FFT estimate
Carrier frequency (MHz)	247	247.00	248.2	250	250.42	NA
PW (ns)	1900	1913.48	2562	1400	1319.29	NA
TOA (ns)	890	874.34	240.03	240	220.56	NA
Power (dBm)	−50	−50.77	−53.13	−60	−60.69	NA

NA: not applicable.

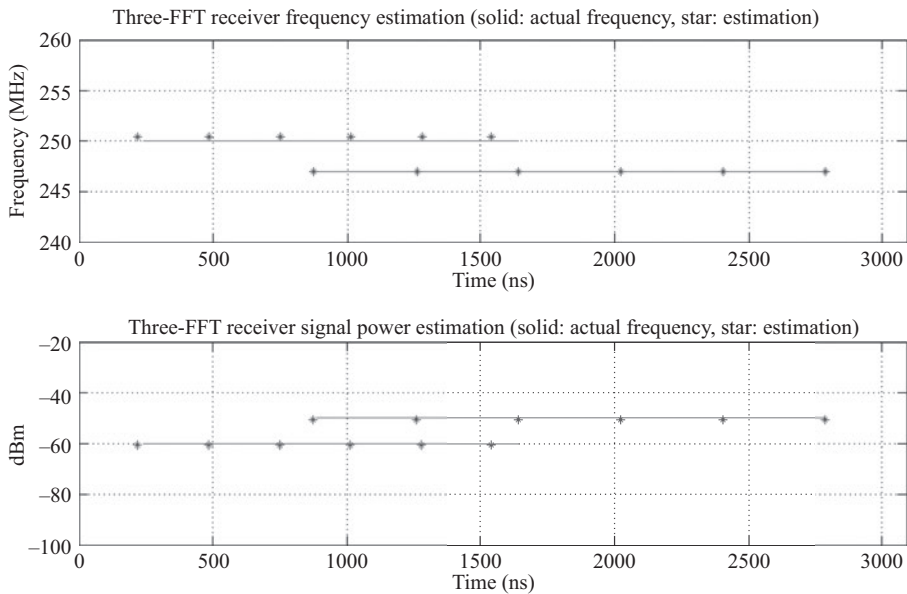


Figure 17.6 Visual display of the multiple-FFT receiver results (case 1).

signal with a very long PW. A visual display of multiple-FFT detection is illustrated in Figure 17.6 (solid line: real signal characteristics; star: EW receiver estimation).

Case 2: Two signals with very short time duration

The multiple-FFT receiver can detect signals with very short duration since one of its encoders uses a very short window (64 points). With the 1.33-GHz sampling rate, the frames of the 64-, 256-, and 1024-point FFTs are 48.1203 ns, 192.4812 ns, and 769.9248 ns, respectively. With the 64-point FFT, the multiple-FFT receiver can detect two signals with a 150-ns PW (3 FFT frames). We used two signals with a PW of 150 ns as the test signals. The signal characteristics of the two signals and the PDW generated by a multiple-FFT receiver

Table 17.2 Test signal characteristics and estimate of multiple-FFT and single-FFT receivers (case 2).

	Signal 1			Signal 2		
	True value	Multiple-FFT estimate	Single-FFT estimate	True value	Multiple-FFT estimate	Single-FFT estimate
Carrier frequency (MHz)	223	222.87	NA	457	456.82	NA
PW (ns)	150	144.36	NA	150	133.81	NA
TOA (ns)	98	96.24	NA	150	149.57	NA
Power (dBm)	-50	-52.67	NA	-40	-41.67	NA

NA: not applicable.

and single-FFT receiver are listed in Table 17.2. It should be noted that since the TOA is estimated based on the ratio between the estimated signal power in the first frame and the average estimated signal power (the first frame is not included), the estimated TOA might be earlier than the actual TOA, as shown in Table 17.2, and it should not be interpreted that the receiver detects the signal before it appears.

As shown in Table 17.2, the multiple-FFT receiver can detect two test signals and create a PDW with reasonable accuracy. On the other hand, the single-FFT with 256-point frame can detect neither signal. A visual display of multiple-FFT detection is illustrated in Figure 17.7 (solid line: real signal characteristics; star: EW receiver estimation).

Case 3: One strong short signal overlaps with a weak long signal

The multiple-FFT receiver can detect one weak long signal and one strong short signal whose carrier frequency difference is small since the receiver uses multiple-frame sizes. The short signal can be detected by the encoder with short frames and the long signal can be detected by the encoder with long frames. We used two signals: one of which is strong but short and another of which is long but weak. The signal characteristics of the two signals and the PDW generated by a multiple-FFT receiver and single-FFT receiver are listed in Table 17.3.

As shown in Table 17.3, a multiple-FFT receiver can detect two test signals and create a PDW with reasonable accuracy. On the other hand, the single-FFT with 256-point frame can detect the long signal but miss the short one. A visual display of multiple-FFT detection is illustrated in Figure 17.8 (solid line: real signal characteristics; star: EW receiver estimation).

Case 4: Two partly overlapping signals with 50-dB power difference

The multiple-FFT receiver can increase instantaneous dynamic range since it has multiple frames. A weak signal can be detected by the FFT with long frame if the duration of the signal is long. In test case 4 we simulate the scenario of two partially overlapping long signals whose power difference is 50 dB. The receiver's minimum instantaneous dynamic range is 45 dB, but a larger instantaneous dynamic range can be achieved provided that the signal carrier frequency difference is larger than the frequency difference between the mainlobe and first sidelobe of the Blackman window. The signal characteristics of

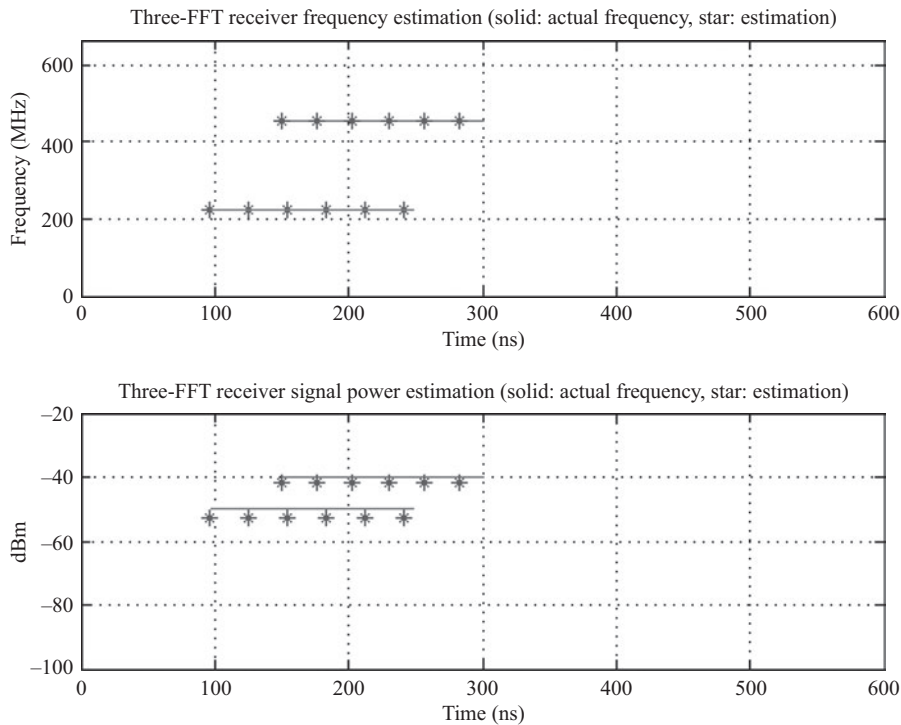


Figure 17.7 Visual display of the multiple-FFT receiver result (case 2).

Table 17.3 Test signal characteristics and estimate of multiple-FFT and single-FFT receivers (case 3).

	Signal 1			Signal 2		
	True value	Multiple-FFT estimate	Single-FFT estimate	True value	Multiple-FFT estimate	Single-FFT estimate
Carrier frequency (MHz)	245	244.91	NA	250	250	249.28
PW (ns)	150	131.99	NA	2450	2118.37	2118.37
TOA (ns)	877	873.74	NA	245	441.47	441.47
Power (dBm)	−30	−29.42	NA	−76	−72.19	−72.19

NA: not applicable.

two signals and the PDW generated by a multiple-FFT receiver and single-FFT receiver are listed in Table 17.4. A visual display of multiple-FFT detection is illustrated in Figure 17.9 (solid line: real signal characteristics; star: EW receiver estimation).

As shown in Table 17.4 and Figure 17.9, both signals can be detected by a multiple-FFT receiver. The PDW of signal 1 is closer to the true signal characteristics since it is detected

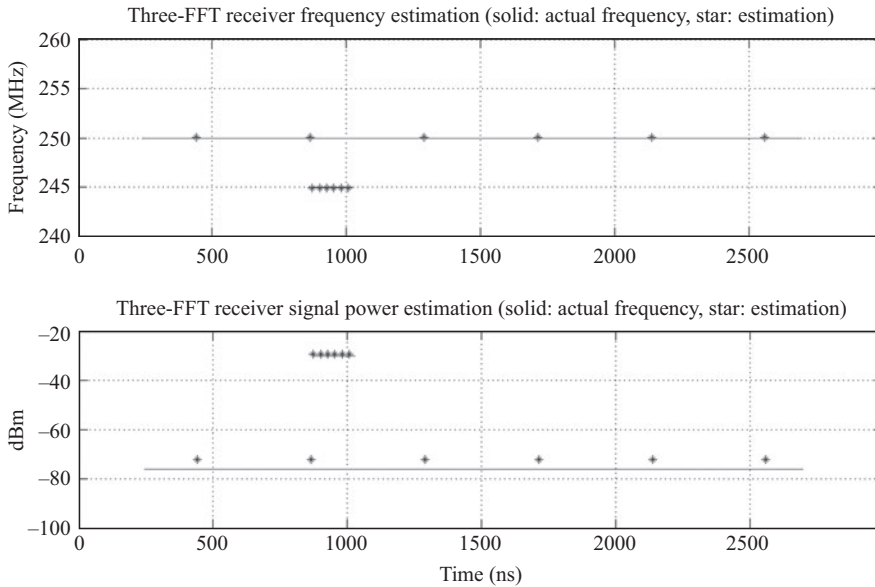


Figure 17.8 Visual display of the multiple-FFT receiver result (case 3).

Table 17.4 Test signal characteristics and estimate of multiple-FFT and single-FFT receivers (case 4).

	Signal 1			Signal 2		
	True value	Multiple-FFT estimate	Single-FFT estimate	True value	Multiple-FFT estimate	Single-FFT estimate
Carrier frequency (MHz)	224	223	223.99	338	338.23	337.51
PW (ns)	1450	1435.84	1452.85	2550	2006.85	1246
TOA (ns)	333	340.79	331.58	613	1020.43	1780.48
Power (dBm)	-27	-28.16	-27.52	-77	-74.78	-74.78

by all three encoders. Signal 2 was detected only by the encoders with 256-point and 1024-point frames. As a result, its TOA and PW estimation are inaccurate. In this specific case, a 256-point FFT receiver can detect both signals. However, it cannot detect the weak signal (signal 2) until the strong signal (signal 1) disappears, thus having smaller instantaneous dynamic range.

The four simulation cases presented in this section demonstrate that the proposed EW receiver design combines the advantage of EW receiver with short FFT frame and an EW receiver with long FFT frame. As a result, the proposed EW receiver can achieve good time and frequency resolution. Our simulation results also show that the multiple-FFT receiver outperforms the 256-point FFT receiver.

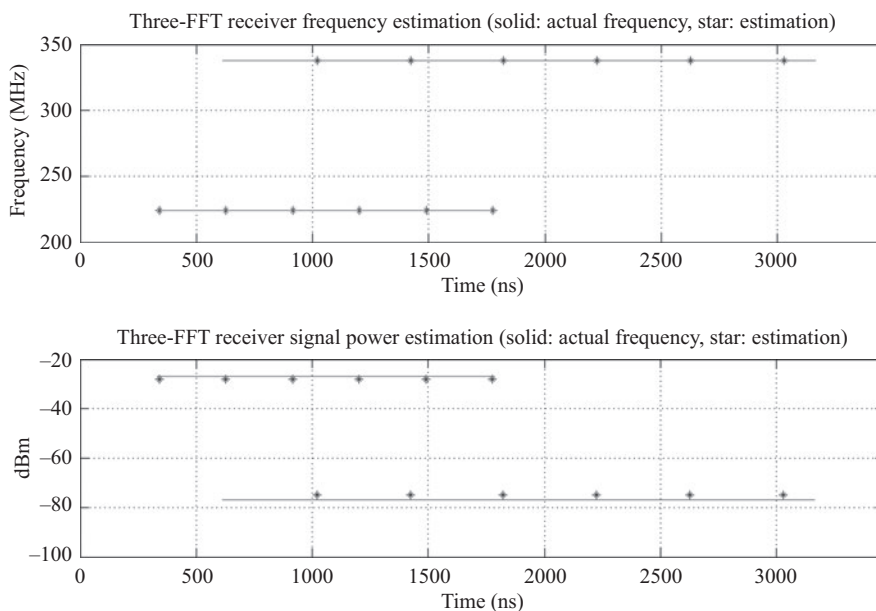


Figure 17.9 Visual display of the multiple-FFT receiver result (case 4).

The accuracy of the TOA and PW estimations require some discussion. For the multiple-FFT receiver, the TOA and PW estimation accuracy is determined by the shortest FFT frame. In this study, maximum TOA and PW estimation errors are 48.12 ns and 96.24 ns, respectively, provided that the signal is correctly detected by the 64-point FFT. If there is more than one signal, these numbers remain valid as long as all signals are correctly detected by the 64-point FFT. However, if two signals cannot be separated by the 64-point FFT (test case 1) or one signal is too weak to be detected by the 64-point FFT (signal 2 in test case 3 and test case 4), then the TOA and PW estimation accuracy deteriorates, as shown in the simulation results. For a weak long signal that can only be detected by a 1024-point FFT, its TOA and PW estimation error can be as large as 769.92 ns and 1539.84 ns, respectively. The TOA and PW estimation accuracy of a multiple-FFT receiver depends on the receiver's instantaneous dynamic range, frequency resolution, and actual signal's PW and demand a thorough systematic study.

17.6 Conclusions

An EW receiver design based on multiple-FFT operations with different frame sizes is presented in this chapter. By combining PDWs generated with long and short FFT frames, we can achieve a better time domain and frequency domain estimation. In this study, only a pulsed continuous wave signal is considered. Continued study is recommended to extend this approach to detect exotic radar signals such as chirp and BPSK signals.

References

- [1] Tsui J. *Special Design Topics in Digital Wideband Receivers*. Boston: Artech House; 2010.
- [2] Cheng C-H, Lin DM, Liou LL, Tsui JB. 'Electronic warfare receiver with multiple FFT frame sizes'. *IEEE Transactions on Aerospace and Electronic Systems* 2012;**48**(4):3318–3330.
- [3] Ji LY, Gao MG. 'Analysis and solution of rabbit ear effect in channelized receiver'. *Proceedings of the 2009 IET International Radar Conference*. New York: IEEE; 2009:1–4.
- [4] Sanders FH. 'The rabbit ears pulse-envelope phenomenon in off-fundamental detection of pulsed signals'. NTIA Technical Report TR-12-487. Washington, DC: U.S. Department of Commerce, July 2012.

Receiver Tests

18.1 Introduction

This chapter discusses the performance of a digital receiver and the procedures for testing it. Receiver performance is one of the most important aspects in receiver research. One of the main impediments in receiver research is that there are no universally acceptable standards in the performance of electronic warfare (EW) receivers. Because of this shortcoming, researchers do not know where to improve EW receiver performance. If one claims to have made some improvements in receiver performance, but cannot report the result quantitatively, it is rather difficult to have these claims accepted. Worse yet, people can make claims about the performances of an EW receiver that cannot even be used in a system. For example, if a receiver misses many signals or produces a large number of spurious responses, in general it cannot be used in any system. Under this condition, no matter how good other performances are, the result should not be reported. There should be some minimum requirements a receiver must be measured against in order to qualify as a functioning receiver.

One interesting experience is in channelized receivers. The main research needed is in the parameter encoder to encode the channel outputs into frequency words. However, since there is no performance standard for the parameter encoder, very few people have worked in this area. Even if one obtains some results, they cannot be published because one may not be able to claim any improvement. On the other hand, the filter bank itself has sufficient performance standards (i.e., insertion loss, bandwidth, frequency and time domain spurious responses, etc.). As a result, a lot of research has been concentrated in this area and many technical papers have been published, but very little effort has been devoted to the major problem of channelized receivers.

In order to stimulate research in EW receivers, not only are performance standards needed, but the standards should also be publicized. In addition, researchers should be encouraged to publish their results so that engineers and scientists working in this area will know the problems and can seek solutions to them. This chapter discusses all the performance issues discussed in Chapter 2.

This chapter discusses different types of receiver tests. It will concentrate on laboratory and anechoic chamber tests rather than field tests since these tests can be performed in a controlled manner and generate results on EW receiver performance.

18.2 Types of Receiver Tests

An EW receiver should be tested under different situations. No matter how thoroughly a receiver is tested, it appears one can never cover all the possible input signal conditions. A receiver should be tested in the following order:

1. Laboratory tests
 - a. Preliminary test
 - b. Regular tests
 - (1) Single-signal test
 - (a) Frequency test
 - (b) Frequency standard deviation test
 - (c) False alarm test
 - (d) Sensitivity test
 - (e) Dynamic range test
 - (f) Pulse amplitude (PA) test
 - (g) Pulse width (PW) test
 - (h) Angle of arrival (AOA) test
 - (i) Time of arrival (TOA) test
 - (j) Shadow time, latency, and throughput rate tests
 - (k) Random frequency/amplitude test
 - (2) Two-signal tests
 - (a) Frequency resolution test
 - (b) Spurious free dynamic range test
 - (c) Instantaneous dynamic range test
 - (d) AOA resolution test
 - (e) Random frequency/amplitude test
2. Anechoic chamber tests
3. Simulator tests
4. Field tests

The laboratory tests are considered the most important tests because they are performed under a controlled environment. Most of the important characteristics of a receiver can be obtained from these tests, therefore these tests results determine the performance of the receiver. The preliminary tests should be carried out first to determine whether the receiver is good enough to undergo regular tests. In this chapter the discussion will concentrate on all the laboratory tests.

Anechoic chamber tests are the same as the laboratory tests. If the radio frequency (RF) input of the receiver can be applied through an RF cable, as shown in Figure 18.1(a), laboratory tests should be used because the input signals can be better controlled. If the input of the receiver is an antenna array, as shown in Figure 18.1(b), the input cannot be applied through an RF cable. In this situation, anechoic chamber tests will be applied. The input signal is applied to the receiver through radiation in an anechoic chamber. The anechoic chamber tests generate all the performance results of a receiver as listed in the laboratory tests and should provide more reliable AOA measurements.

If a receiver has a receiving antenna array and an anechoic chamber is not available, one must simulate a planewave front as the input to the RF connectors. The planewave front is

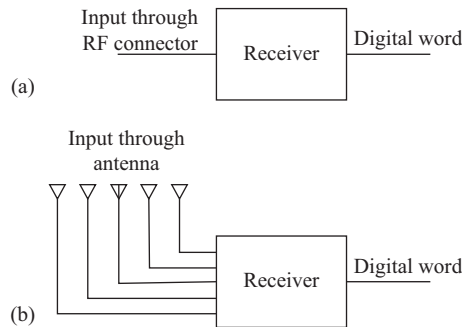


Figure 18.1 A receiver represented by a black box: (a) input through an RF connector and (b) input through an antenna.

difficult to achieve, especially over a wide frequency and incident angle ranges. However, if a receiver has an amplitude AOA measurement system, it can be tested in the laboratory or in the anechoic chamber since it is easy to simulate the signal input for an amplitude comparison AOA measurement system.

Simulator tests usually use special equipment to simulate the signal environment an EW receiver passes through. The environment can be static or dynamic, which changes with time. Under these tests the receiver can be evaluated under very high signal density. Sometimes the outputs of the receiver are processed by a real-time signal processor to determine the performance of the overall system, which includes the signal processor. If a real-time processor is not available, the output data from the receiver under test can be recorded and compared with the input signals.

Theoretically the input signal condition in the field tests is unknown. However, with an unknown input signal condition, the performance of the receiver cannot be determined. In order to remedy this shortcoming, two receivers can be tested at the same time to compare their results. If one can control some radars in the field, certain radars can be turned on to transmit. Under this situation, at least the signals of these radars are known and can be used to check the performance of the receiver.

18.3 Preliminary Considerations in Laboratory Receiver Tests

Before the discussion of laboratory tests of an EW receiver, it is appropriate to consider the philosophy of receiver testing. The tests should be limited to receivers that can process simultaneous signals. Specific test procedures can be designed to evaluate receivers that can process only one signal. For example, the performance of an instantaneous frequency measurement (IFM) receiver and a superheterodyne receiver is often available.

1. The receiver under test is considered as a black box with RF as the input and pulse descriptor words (PDWs) as the output. Therefore these tests can be applied to digital as well as analog receivers.
2. The receiver under test must have PDWs as the output. Each PDW contains all the desired information (i.e., frequency, PA, PW, TOA, and AOA). To emphasize this point,

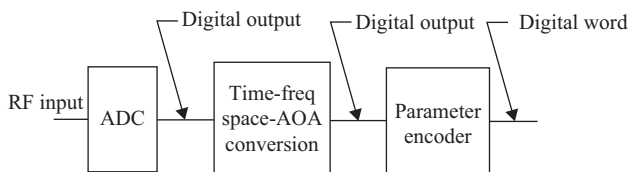


Figure 18.2 Three blocks of a digital receiver.

a digital EW receiver is represented by a block diagram as shown in Figure 18.2. In this figure the time–frequency and space–AOA conversions are through digital signal processing, thus the output is in digital form and represents the information in the frequency and AOA domains. However, these digital data should not be considered as the outputs of the receiver. This information must be converted into a PDW containing the desired information. One should not use other information, even if it is in digital form [i.e., fast Fourier transform (FFT) outputs to evaluate receiver performance] since it is not a trivial task to generate the PDW.

3. *Input conditions.* It is impractical to cover all the input conditions during the test. For example, if a receiver can receive two simultaneous signals and the input bandwidth is 1000 MHz with a single-signal dynamic range of 70 dB (from -60 to 10 dBm) and an instantaneous dynamic range of 40 dB (from -60 to -20 dB), the number of input conditions can be found as follows:
 - a. *Single-signal test.* If the frequency data resolution of the receiver is 1 MHz and the amplitude data resolution is 1 dB, for the single-signal test the required number of inputs is at least 70,000 (1000×70) to test each output condition. However, the frequency and amplitude resolution steps may not be limited to 1-MHz and 1-dB steps. If the signal generator can change the frequency in steps of 1 kHz and the amplitude by 0.1 dB, the possible input conditions can be extended to 700 million ($1,000,000 \times 700$) possibilities, which is impractical to carry out.
 - b. *Two-signal test.* For the two-signal case, the number of input signals is even larger. For each signal there are 40,000 (1000 in frequency $\times$ 40 in amplitude) output cells. The total possible outputs are 799.98 million ($40,000 \times 39,999/2$), which is unreasonably large. This is the minimum number of inputs required to test all the possible outputs. If the signal generators have finer frequency and amplitude steps than the receiver, the number of input conditions can be made unimaginably large.
 If the receiver can process more than two simultaneous signals, the number of input conditions would be even larger. In reality, it is impossible to test a receiver with several thousand input conditions, even if the test setup is automated. It is apparent from this discussion that only a very small percentage of the input conditions can be tested. The limited tests may not reveal all the deficiencies in a receiver since even well-behaved receivers used in a real environment may generate unexpected data.
4. *Type of inputs.* The input signals can be applied to the receiver under test in two possible ways. First, the signal can be applied sequentially (i.e., the frequency is changed from low to high values in uniform steps). Second, the signal can be applied in a random fashion (i.e., random frequency and amplitude).

The advantages of using systematic inputs are that the results can be presented graphically and it is easy to repeat the same test. Sometimes it is desirable to repeat a condition at which the receiver generates erroneous information so the receiver design can be corrected. The major disadvantage of this type of test is that the input may be biased and limited to a very small region.

The advantage of random input signals is that the distribution can be made more uniform over a desired range. A true random input signal is difficult to repeat. A pseudorandom signal generated through a computer can be repeated. It might be difficult to present this type of test result graphically. Probably the simplest way to represent the data is through tabulation.

In a receiver test, both systematic and random input signal conditions should be used.

5. *The test results presentation.* The test results obtained from the receiver must be easily understandable from some simple display (i.e., graphic or tabulation). It is impractical to generate pages of printed data to locate some erroneous information. If the measured data cannot be presented in some simple forms, it does not make much sense to generate them in the first place.
6. *One- and two-signal tests.* Based on the discussion in number 5, it is reasonable to limit receiver tests to one or two signals. Most of the receiver performance definitions are based on one or two signals. If a receiver can process four simultaneous signals, the number of possible input signal conditions is very large. A small portion of the input signal conditions used in a test may not be significant at all. Thus it is reasonable to test only whether the receiver can really process four simultaneous signals.

18.4 Receiver Tests Through Software Simulation

It is highly desirable to evaluate the performance of a receiver before it is fully fabricated. On average, development of a breadboard analog EW receiver takes more than 3 years from design to completion. Unfortunately, sometimes there are obvious design flaws, but they cannot be easily detected. However, once the receiver is built, these flaws are difficult to correct. If all the receiver tests can be performed through software simulation, some of these flaws may be discovered at a much earlier stage and can be easily corrected.

Simulations of analog receiver design have been carried out for some time. The effort can be divided into two parts: (1) the analog portion from RF to video output and (2) the parameter encoder from video to PDW, as shown in Figure 18.3. The work on the first part, from RF to video signal, may be considered very successful. The video output matches the simulated output very closely. Due to the complicated nature of the parameter encoder

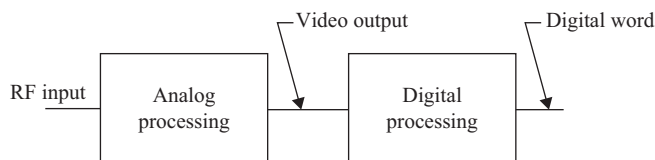


Figure 18.3 Two blocks of an analog receiver.

design, only very limited simulations are achieved, if any at all. With advances in computer technology, there is a chance the parameter encoder can be simulated and evaluated before it is actually built.

For a digital receiver, the time–frequency and space–AOA transforms are all performed in the digital domain. However, as mentioned earlier, these digital outputs can only be considered as intermediate outputs and should not be used as the receiver final outputs. Converting the frequency and AOA information into PDWs is done through digital processing. Therefore all these operations can be simulated by a computer. A receiver should be adequately simulated before actual fabrication. This simulation should be made equivalent to an actual receiver test procedure, which means that with a certain input the receiver should generate a certain output. If the simulation can be processed at very high speed, this simulation can be very useful in digital receiver design.

18.5 Laboratory Test Setup

The laboratory test setup is shown in Figure 18.4. In this figure there are two signal sources. Each signal source contains one signal generator followed by a pin modulator, which is controlled by a pulse generator to produce an RF signal with the desired PW. An attenuator is used to adjust the signal strength. It should be noted that in the arrangement the input to the pin modulator is at a constant power level. Under this condition, when the input power level changes, the shape of the pulse generated does not change. If the attenuator is placed in front of the pin modulator, the input to the pin modulator changes. Under this condition, when the signal power changes, the shape of the signal may change.

The two signal generators are combined through a power combiner as the input to the receiver under test. A power meter, a spectrum analyzer, and an oscilloscope are used to check the output of the signals. The power level at the input of the receiver needs to be calibrated because the power loss from the signal generator to the input of the receiver is frequency dependent. Figure 18.5 shows the uncalibrated and calibrated cases of the power

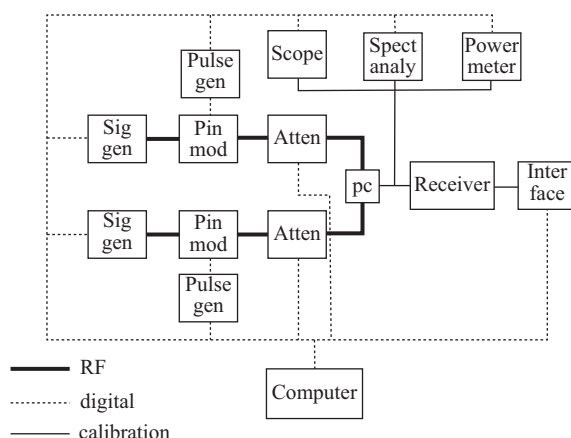


Figure 18.4 Laboratory receiver test setup.

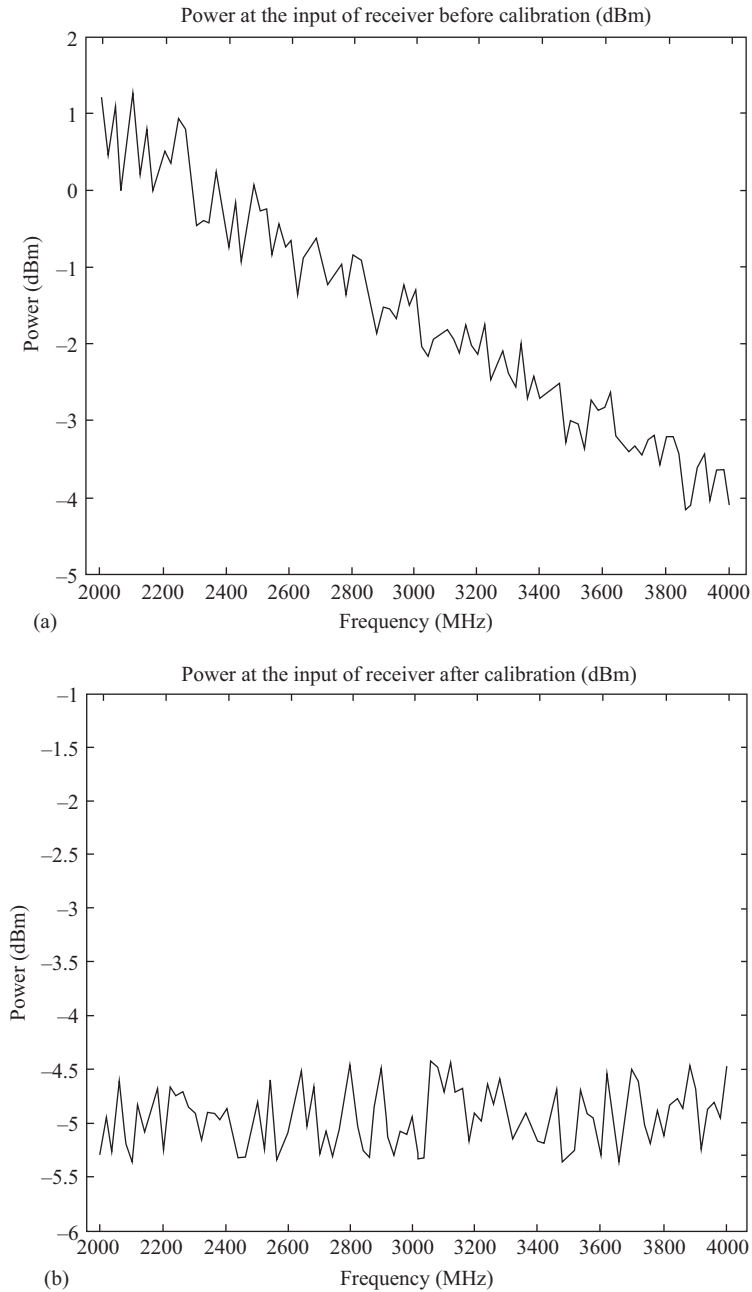


Figure 18.5 Signal power at input of the receiver under test: (a) uncalibrated case and (b) calibrated case.

output from 2 to 4 GHz. As expected, the uncalibrated power output is higher in the low-frequency range than at the high-frequency end, as shown in Figure 18.5(a). The calibrated output is obtained by increasing the attenuation at low frequency to equalize the output power across the entire frequency range, as shown in Figure 18.5(b). The outputs of the receiver are collected through a data collection instrument. The DAS 9200 (Tektronix, Beaverton, OR) can be used to collect data. However, the data-collecting instrument can be a computer used in the direct memory addressing (DMA) mode.

All the equipment except the receiver under test is controlled by a computer. Some receivers are computer-controlled, but under tests the receiver should stay in a certain operation mode. The signal sources are computer-controlled to generate the desired signals. The output data are then compared with the input data. There are two ways to compare these data. One way is to compare each input immediately with the output. The other way is to collect all the data and compare them with all the input data at the end of the test. It appears that the latter approach can save time.

18.6 Anechoic Chamber Test Setup

The anechoic chamber test setup is very much like the laboratory test setup. The only difference is that the signals are radiated to the receiving antenna of the receiver under test. Figure 18.6 shows such an arrangement. Two signal sources used in the test are identical to the ones discussed in the previous section. The only difference is that the two signals are not combined together; instead, each feeds a transmitting antenna. The antennas are directed at the receiver under test. The arrows in this figure indicate the direction of the antenna movement. The transmitting antennas can be manually placed in position. The receiver with the receiving antenna is placed on a turning table. By rotating the receiving antenna, the AOA of the input signal can be changed. To change the incident angle between the two transmitting antennas, the positions of the transmitting antennas must be changed.

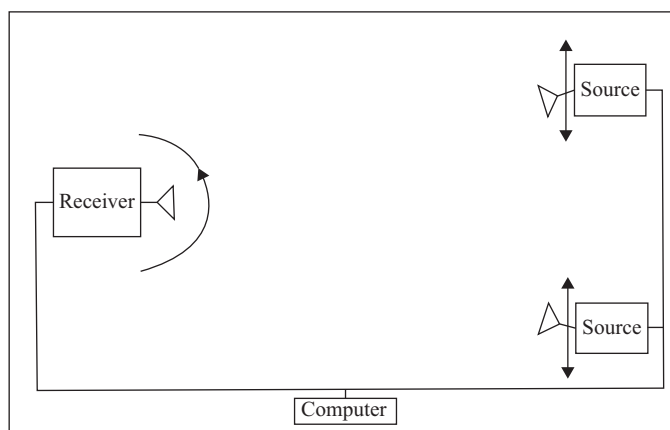


Figure 18.6 Anechoic chamber test setup.

The power to the receiving antenna must be calibrated. It usually requires a standard horn antenna with known gain to measure the signal strength. The power measured is no longer in milliwatts or dBm, but in milliwatts or dBm per unit area. In this calibration procedure, the receiving antenna must line up on the boresight with the transmitting antenna.

18.7 Preliminary Tests

A receiver must pass at least two simple tests to qualify for further tests. If the receiver is designed to cover 1000 MHz (1000 to 2000 MHz) in bandwidth and can process two simultaneous signals in the range of 40 dB (from -60 to -20 dBm), simple one-signal and two-signal tests should be performed. The range of the input signals must be well within the desired performance range. For example, the input frequencies are limited within 800 MHz (from 1100 to 1900 MHz) and the amplitudes are limited to 20 dB (between -50 and -30 dBm). Within these limits, the well-designed receiver should perform satisfactorily.

1. *Single-signal test.* Randomly select one signal within the limited frequency and dynamic range to check for missing signals and spurious signals, which means more than one signal is detected. The random input signal can be selected at 1000 different values to see what percentage of missing and spurious signals is detected. If the percentage of missing and spurious signals is high (i.e., more than 1% missing signals and more than 2% spurious signals), further tests should not be performed.
2. *Two-signal test.* Randomly select two signals within the limited frequency and dynamic range to check for missing and spurious signals. The same criteria as mentioned in the one-signal test can be used to stop further tests.

Past experience indicates that under the single-signal test some poorly designed receivers often generate a high percentage of spurious signals. These receivers should not be tested for the two-signal condition. Even if a receiver passes the single-signal test, it may not pass the two-signal test. A poorly designed receiver under two-signal tests may miss one of the input signals, or worse yet, both signals.

18.8 Single-Signal Frequency Test

Two frequency tests will be discussed in this section: the frequency accuracy test and the standard deviation test on frequency.

18.8.1 Frequency Accuracy Test

In this test, the input frequency is considered as a variable, whereas the PA and PW are constants. During the test, a certain PA and PW is selected. The input frequency is varied from the lower frequency bound of the receiver to the upper frequency bound in predetermined steps. At each frequency interval, only one signal is sent to the receiver. If the input frequency is f_i and the measured frequency is f_m , the error frequency f_e is defined as

$$f_e = f_m - f_i. \quad (18.1)$$

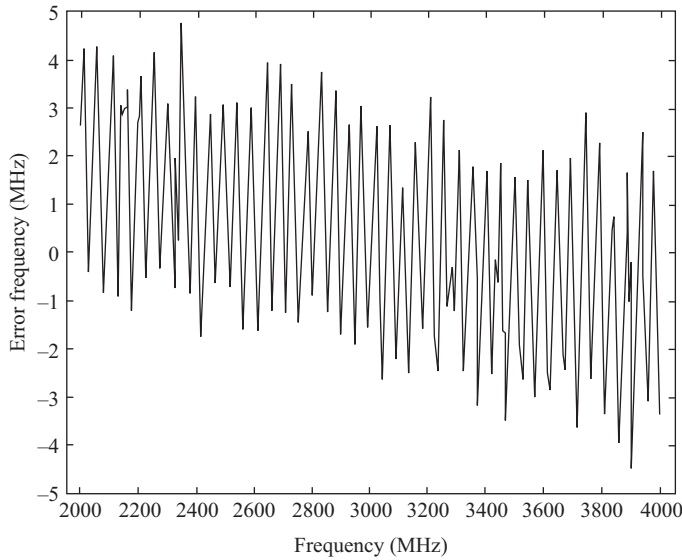


Figure 18.7 Frequency measurement.

The frequency accuracy result is the plot of f_e versus f_i , as shown in Figure 18.7. Using this figure, one can determine the frequency resolution and the bias error. The frequency data resolution can be obtained from the width of the error frequency. From this figure, one can see that the frequency data resolution is about 4 MHz. If the receiver is properly designed, the width of the error frequency is uniform. If N input frequencies are applied to the receiver, the bias frequency f_b is defined as

$$f_b = \sum_{i=0}^{N-1} f_{ei} = \sum_{i=0}^{N-1} (f_{mi} - f_{ii}), \quad (18.2)$$

where the second subscript i represents the i th data point. The first subscript m and i represent the measured and input frequencies, respectively. If there is no bias, $f_b = 0$.

In Figure 18.7, at low frequency the bias is positive and at high frequency the bias is negative. If (18.2) is used to calculate the bias frequency, the result might not be accurate because the bias at low frequency may offset the bias at high frequency. The width of the frequency variation is approximately equal to the frequency resolution cell. In this case, the frequency resolution cell is approximately 4-MHz wide.

18.8.2 Frequency Precision Test

The purpose of this test is to find the consistency of the receiver output data. In some receivers, for the same input condition the receiver may report a different output. In this test,

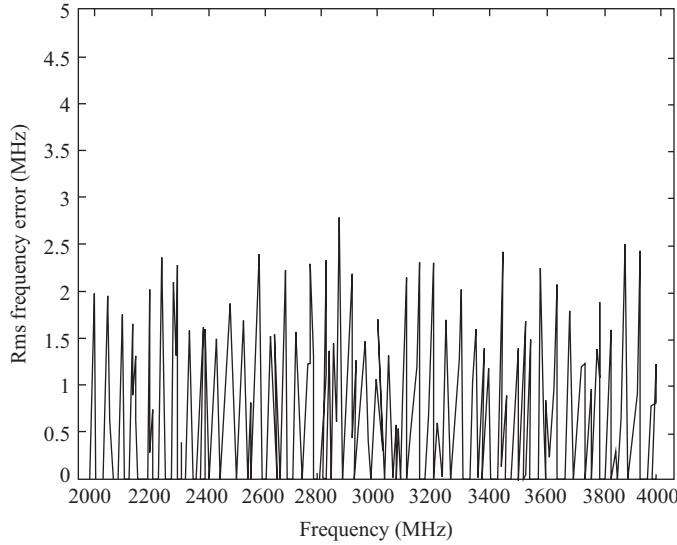


Figure 18.8 The rms values of frequency measurement.

the input data are applied at the same value repetitively N times and the output is measured N times. The root mean square (rms) value is defined as

$$f_s = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (f_{mi} - f_i)^2}, \quad (18.3)$$

where the second subscript i in m_i represents the i th data point and f_i is the input frequency. Figure 18.8 shows the rms values of frequency measurement for a receiver with $N = 100$. In this test it appears that the frequency reading is consistent near the center of the channel where the rms values are equal or close to zero.

18.9 False Alarm Test [1–6]

The false alarm test is the most difficult test, especially if a numerical value is desired. In general, the false alarm rate is very low in an EW receiver and only a limited time can be used to measure it. For example, if a receiver is measured for 1 hour and zero false alarms are recorded, the only claim one can properly make is that the receiver has been tested for 1 hour and zero false alarms were recorded. Actually the receiver false alarm rate can possibly be a few in an hour to one in several days or even months. Thus it is difficult to obtain a numerical value. However, it is desirable to obtain a numerical value in false alarm measurement. If the receiver is tested for a long time, it is difficult to keep the environment (i.e., noise level) unchanged.

There are two approaches to obtain measurable false alarms. One is to decrease the threshold of the receiver. This approach is referred to as importance sampling [1–3].

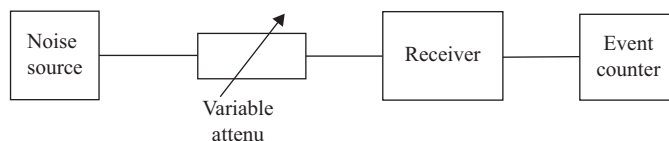


Figure 18.9 False alarm measurement setup.

The difficulty with this method in an EW receiver is that there might be many thresholds. For example, in a channelized receiver, at the output of each channel there is a threshold. To adjust all these thresholds is difficult. The second approach to increase the number of false alarms is to add noise in the front end of the receiver [4]. This method is discussed here.

Noise can be added at the input of the receiver to obtain a measurable false alarm value without changing the threshold. In order to obtain a numerical value, the RF bandwidth B_R to video bandwidth B_V ratio ($\beta = B_R/B_V$) and the noise figure of the receiver must be known. In addition, it must be known whether the receiver is direct current (dc) or alternating current (ac) coupled. The derivation is not discussed here, only the results are presented. The measurement setup is shown in Figure 18.9. In this figure the noise source is known and the attenuator can be adjusted to obtain a measurable false alarm (i.e., several false alarms in 1 second). Under this condition, if one measures over a relatively long time with respect to 1 second (say, e.g., 10 min), a dependable quantity can be obtained. Once this amount of false alarm is obtained, the false alarm can be extrapolated from the measured value.

A computer program presented in Appendix 18.A can be used to find the probability of the false alarm. The inputs to this program are ac or dc coupling, attenuator setting, receiver noise figure, noise source power (in dBm), RF bandwidth, video bandwidth, and measured false alarm time. If this information is furnished, the probability of false alarm of the receiver can be calculated.

18.10 Sensitivity and Single-Signal Dynamic Range

The sensitivity of a receiver might be a function of PW, so it should be tested at a fixed PW. The sensitivity also varies with frequency, so the sensitivity should be measured as a function of frequency.

The procedure to measure the sensitivity at one frequency value is as follows. At a given PW and a fixed frequency, start the PA at a very low value such that the receiver cannot detect the signal. Increase the PA in steps until the receiver can receive the signal (i.e., the signal triggers the data reporting bit). In general, if a receiver is properly designed, at the sensitivity level the frequency reporting should be correct, otherwise the data reporting bit should not be triggered. However, in some receiver designs it is desirable to achieve the maximum sensitivity, even if the frequency reported does not meet the required accuracy.

In order to keep the sensitivity measurement consistent, the frequency accuracy should be evaluated to confirm the sensitivity. It is arbitrarily defined that at the sensitivity level, 90% of the frequency reading should be correct. Thus when the input signal is detected by the receiver, the input signal should be repetitively sent 100 times at this power level. At the same time, the frequency should be recorded. If 90 frequency readings are within a predetermined frequency accuracy, this power is the sensitivity level. If fewer than 90 correct

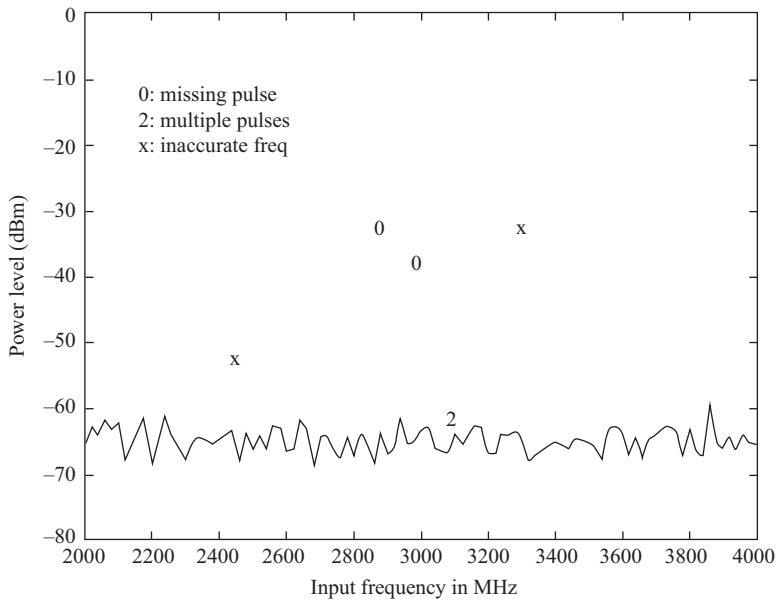


Figure 18.10 Sensitivity as a function of frequency.

frequency readings are obtained, the power level should be increased by one step and the same procedure repeated until more than 90% of the frequency readings are correct.

To find the single-signal dynamic range, keep increasing the input amplitude until additional signals are detected, which are referred to as spurs. In some receivers, when the input is high, the receiver may miss the input signal. Sometimes a receiver can take a high input signal without causing any problem. Under this condition it might be difficult to find the upper limit of the dynamic range because of the limited input power the signal generator can provide. The receiver can be claimed to have a minimum single-signal dynamic range over a certain value. If amplifiers are used to increase the input power level, precautions must be taken to keep the power amplifier from generating undesired spurs. Figure 18.10 shows the sensitivity of a typical receiver.

In this figure, when two signals are recorded the number 2 is printed. If the receiver misses the signal, a 0 is printed. If the frequency reading is outside the desired value, an x is printed. In this figure, the upper limit of the dynamic range is above 0 dBm.

18.11 PA and PW Measurements

These measurements are straightforward. In the PA measurement, the input frequency and the PW are kept constant. Increase the input power in steps and record the PA output at each step. One can either plot the error versus the input amplitude or the output versus the input amplitude. Although the former plot may provide more accurate results, the latter may provide a better pictorial display. Figure 18.11 shows the plot of input amplitude versus output amplitude.

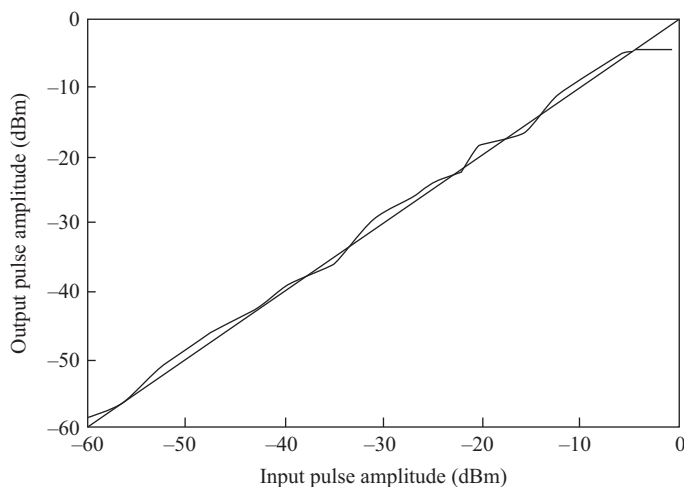


Figure 18.11 Output versus input PA plot.

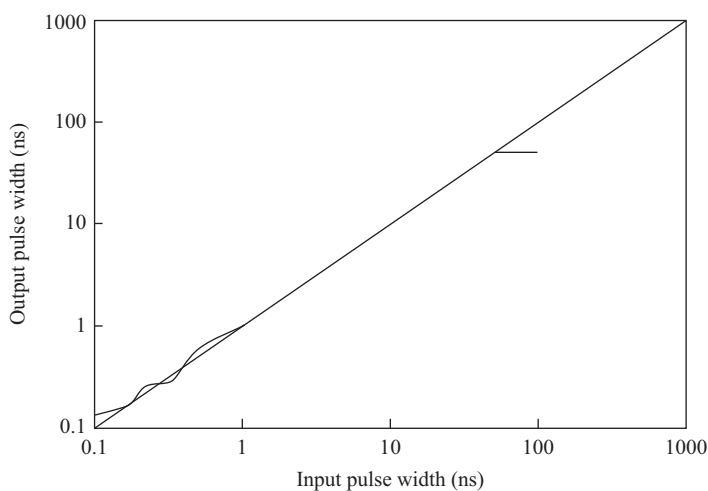


Figure 18.12 Output versus input PW plot.

In the PW measurement, the frequency and PA are kept constant. Increase the PW through either uniform or nonuniform steps from a minimum to a maximum value. The minimum value equals the shortest PW the receiver can process. The maximum value is the PW when the receiver declares a continuous wave signal. The PW output at each step is recorded. A plot of input PW versus output PW is shown in Figure 18.12.

In these tests, if the input signal is applied repetitively at the same value for a number of times, the standard deviation can be obtained.

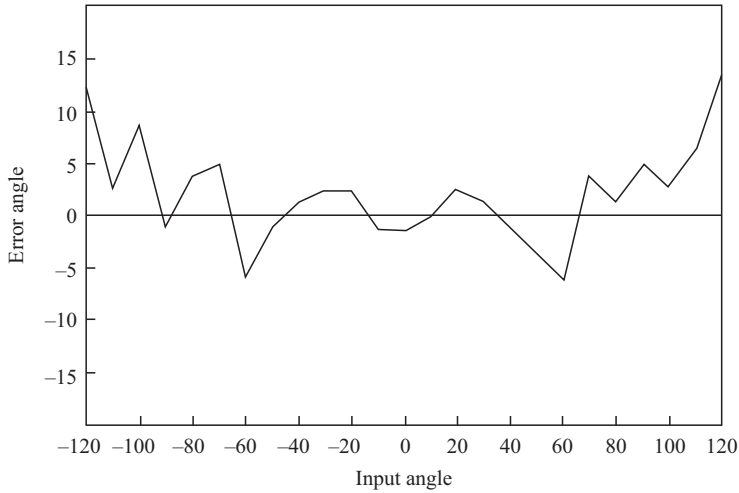


Figure 18.13 Output versus input AOA plot.

18.12 AOA Accuracy Test

In this test, if the receiver uses amplitude comparison to obtain the AOA information, the input can be applied through several RF connectors. The amplitude to each input is controlled separately to simulate the input antenna pattern. If the input antenna pattern cannot be faithfully simulated, the AOA measured can only provide a qualitative result.

If the receiver is a phase interferometric system, the AOA test can be carried out in an anechoic chamber. One input source is required and its position is fixed. The input frequency PA and PW are also fixed. The receiving antenna is placed on a turntable. Rotating the turntable in steps changes the input AOA. Record the output AOA at each step. Figure 18.13 shows the result of an AOA test.

A standard deviation test can be carried out to evaluate the consistency of the receiver performance.

18.13 TOA Test

In this test, the frequency, PA, and PW are fixed. The input signal must be a pulse train with a very stable pulse repetition frequency (PRF), also referred to as pulse repetition interval (PRI). Each pulse received has a TOA associated with it. This TOA is referenced to an internal clock, thus it is difficult to directly compare measured TOAs with input TOAs. Usually the measured difference TOA (or PRI) is compared with the input PRI. The difference TOA is defined as

$$\Delta\text{TOA}_i = \text{TOA}_i - \text{TOA}_{i-1} = \text{PRI}, \quad (18.4)$$

which is obtained from two consecutive TOAs. In this test, the result is presented as the error PRI (ΔTOA_i). The error PRI is defined as

$$\Delta\text{PRI} = \text{PRI}_m - \text{PRI}_i, \quad (18.5)$$

where PRI_m and PRI_i are the measured and input PRI, respectively. Usually the TOA generated by a receiver does not have much of a problem.

18.14 Shadow Time, Throughput Rate, and Latency Tests

The shadow time is the time required for a receiver to accept a second pulse after the first pulse. In this section, the shadow time is defined from the end of the first pulse to the leading edge of the second pulse. Sometimes the shadow time is PW dependent. The explanation for this effect is as follows. The frequency and AOA measurements often start at the leading edge of the pulse. It usually takes more time to measure frequency and AOA than to measure the PA, PW, and TOA. But the PW measurement cannot start until the end of the pulse is reached. If the PW is short, the PW data may be ready before the frequency and AOA measurements, thus it requires relatively more time to accept a second pulse. If the PW is long, the frequency and AOA measurements will be ready before the end of the pulse. Under this condition, as soon as the pulse ends, it takes a relatively short time to accept a second pulse.

The above discussion can be concluded as follows. For a short pulse, the shadow time may be longer than that of a long pulse. Thus the shadow time is PW dependent. However, if the PW is increased beyond a certain value, the shadow time becomes a constant. Therefore the shadow time should be measured at the minimum anticipated PW as well as at the longer PW. Thus the shadow time may have several values.

Another quantity related to the shadow time is the throughput rate. Throughput rate can be defined as the maximum pulse density a receiver can process. The measurement procedure is similar to the shadow time measurement. The PW should be at the minimum value and the input frequency and PA are kept at fixed values. If the receiver is designed to process four simultaneous signals, four synchronized signal generators are required. If the receiver can process two signals, two synchronized generators are needed.

Let us use the two-signal receiver as an example. Starting at a low PRF with two simultaneous signals of minimum PW, the receiver will be able to process all the input signals. Keep increasing the PRF until the receiver can process only half of the input signals, which indicates the receiver processes two simultaneous signals and misses the next two. The maximum number of pulses a receiver can process without missing pulses is the throughput rate. The corresponding shadow time can be found from the trailing edge of the first pulse to the leading edge of the second one as it is defined. For example, if the receiver under consideration can process a minimum PW of 200 ns and the highest PRF without missing pulses is 2 MHz, the throughput rate is 4 MPs/s by taking into consideration that the receiver can process two simultaneous signals.

Latency time is defined as the time delay from the input pulse arriving at the receiver to the time the digital output word is generated. This time can be measured from a scope. It has a time from tens of nanoseconds to a few microseconds.

18.15 Two-Signal Frequency Resolution Test

For the two-signal tests, in order to keep the data under control the two signals are limited to the same PW and the leading edges of the two pulses are time-coincident. If one desires to put a certain delay between the two pulses or use two pulses with different PWs, the test conditions should be carefully specified. The presentation of the output data must be carefully planned.

These tests are to determine the capability of the receiver to separate two signals close in frequency. Set the two signals with the same amplitude and close to the maximum power the receiver can process. The PW should be set long enough so that the spectrum spreading caused by the short PW will not affect the measurement. Keep one input signal near the center of the input bandwidth at a fixed frequency. Set the second signal at a frequency far from the first one and make sure that the receiver can measure both signals correctly. Move the frequency of the second signal toward the first one until the receiver misses one of the input signals or reads one of the input frequencies incorrectly. The minimum frequency separation at which the receiver can process both signals is the frequency resolution.

18.16 Two-Signal Spurious-Free Dynamic Range Test

In this test the frequencies of the two input signals are separated at a constant value. Both frequencies are changed by the same value, therefore the difference frequency between the two signals is kept constant. The minimum frequency separation must be greater than the desired two-signal frequency resolution. Both signals are kept at the same amplitude.

To start the test, both signals are set slightly below the receiver sensitivity level. Increase both signal amplitudes in steps. If both signals are properly received, the power is marked as the lower limit of the dynamic range and the corresponding frequency is the average frequency of the signals. Increase the amplitude of both signals until more than two signals are reported. In order to make sure the extra signal is produced by the third-order intermodulation, one of the extra signals must have a frequency of

$$f = 2f_1 - f_2$$

or

$$f = 2f_2 - f_1, \quad (18.6)$$

where f_1 and f_2 are the two input frequencies. If the extra signal does not match this condition, it might be caused by some other spurious response. The amplitude of the signals keeps increasing until the extra signal fulfills the above condition. This power level is marked as the upper limit of the dynamic range.

During the test, if the receiver reports more or less than two signals (including no output), the number of signals is recorded. If the frequency reading of one or both signals is reported erroneously, an x is recorded. This test can be repeated for different frequency separations. Figure 18.14 shows the test result of an experimental receiver. The frequency scale represents the average frequency of the two input signals.

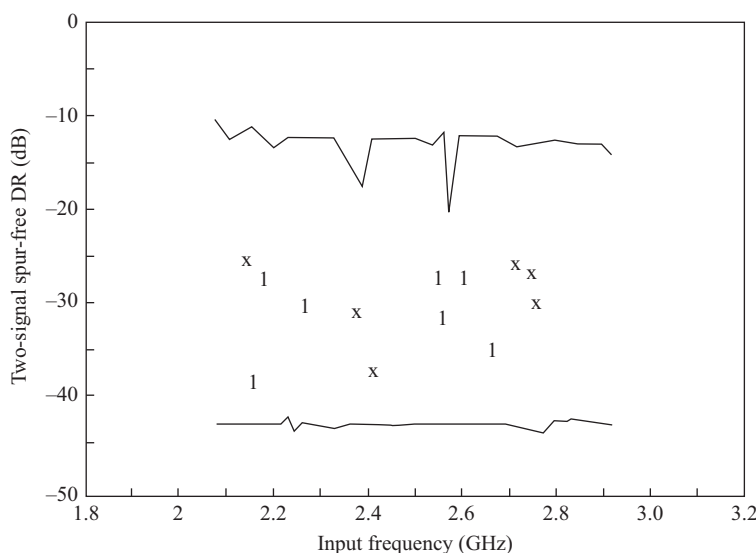


Figure 18.14 Two-signal spur-free dynamic range.

18.17 Instantaneous Dynamic Range Test

This test is to find the capability of the receiver to simultaneously receive a strong and a weak signal. One of the input signals, the first one, is set at a fixed frequency (say, at the center of the input band) and close to the upper limit of the single-signal dynamic range. The second signal starts at the lower frequency limit of the receiver with an amplitude less than the sensitivity level. Increase the amplitude of the second signal until the receiver reports both signals correctly; this is the lower limit of the instantaneous dynamic range at this frequency. Increase the frequency of the second signal and repeat the above procedure to obtain another lower limit. The second signal should cover the entire frequency range of the receiver. The minimum frequency between the two signals should be equal to or slightly greater than the two-signal frequency resolution. Figure 18.15 shows the result of a typical instantaneous dynamic range test.

18.18 Anechoic Chamber Test

If one cannot feed the input of an EW receiver through an RF connector (i.e., the input of the receiver is an antenna array), all the tests mentioned in Sections 18.7 through 18.17 should be performed in an anechoic chamber to determine the performance of the receiver. However, it is suggested that if possible one should avoid testing a receiver in a chamber since it is much more complicated to set up in an anechoic chamber than in the laboratory. In addition, anechoic chambers are not generally available. Even if the input of a receiver has an antenna array, as in a phase interferometric AOA system, one of the input antennas might be disconnected and replaced with an RF connector. If possible, this approach can be used to measure all the parameters except the AOA.

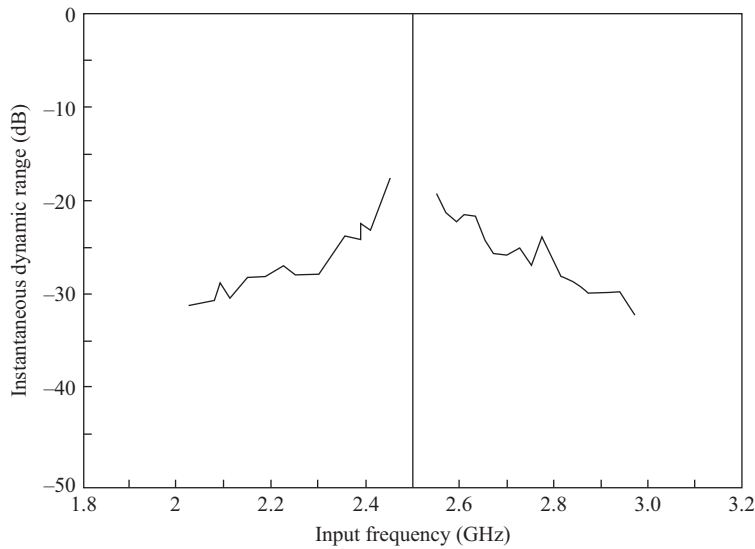


Figure 18.15 Instantaneous dynamic range.

In an anechoic chamber, the distortion of the phase front of the incoming wave should be properly analyzed and measured. The amount of reflection in an anechoic chamber as a function of frequency should also be known. The receiver should be placed at the far field of the transmitting antenna if possible. The wavefront curvature affecting the receiver test should be taken into account. Thus the AOA test carried out in an anechoic chamber should produce more reliable data than through antenna simulation.

18.19 AOA Resolution Test

The purpose of this test is to determine the capability of the receiver to separate two signals with different AOAs. This test should be carried out in an anechoic chamber. If two signals arrive at the receiver with the same frequency but different AOAs, some receivers can differentiate them (i.e., AOA is obtained from the FFT in the space domain) and some receivers can't (i.e., AOA obtained from a phase interferometric system). Since a small number of antennas can be installed in an airborne system, it is anticipated that most of the AOA measurements in EW receivers will be from a phase interferometric system. As discussed in Chapter 15, in some AOA measurements the signals are separated by their input frequencies first. If two signals have the same frequency, they cannot be separated and the phase of the interferometric cannot be used to measure the AOA.

The capability of separating two signals of the same frequency by AOA is the true AOA resolution. The true AOA resolution should be frequency independent. Since most EW receivers do not have this capability, the AOA resolution considered here is frequency dependent.

In order to simplify the measurement procedure, the two input signals should be separated widely in frequency such that they can be separated by the receiver under test. The two input signals have the same PA and PW. The receiving antenna is kept stationary. One of the

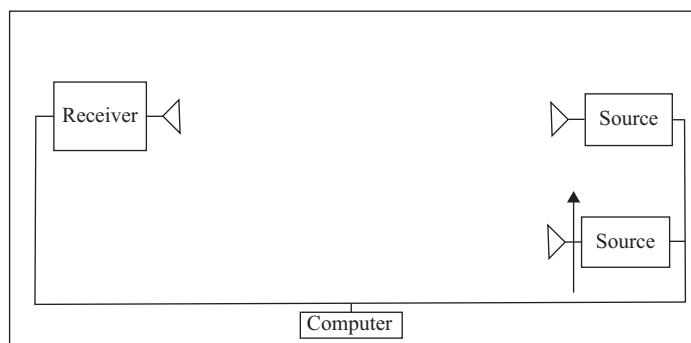


Figure 18.16 Two-signal AOA measurement setup.

input sources is placed at the center of the receiving antenna beam. The second source is placed at one end of the beam and moved toward the other end of the beam, as shown in Figure 18.16. At each step the AOA of both signals is recorded. If the receiver is properly designed, it should correctly encode both AOAs across the entire angle range. If the two signals have the same AOA but different frequencies, the receiver should encode them correctly. The results can be represented by two curves, each representing the input AOA versus the output AOA of each source. The frequency difference between the two input signals should be decreased to repeat the same test to find the limit of the receiver.

18.20 Simulator Test

The purpose of a simulator test is to determine how a receiver performs under a dense signal environment. A simulator simulates an electronic environment that an EW receiver is anticipated to operate in. A simulator, in general, can produce a very dense environment containing many radars. It usually can simulate the antenna pattern of the radar transmitter, such as the scan rate of a radar. A sophisticated simulator has many parallel output ports. These ports can be connected to the parallel input ports of the receiver under test to simulate the input antenna amplitude pattern. Thus if the receiver has an amplitude comparison AOA system, it can be tested through the simulator. It might be difficult to simulate the wavefront of many input signals to test a phase interferometric AOA system with the present technology, but it is possible to simulate the wavefront of one signal.

A simulator should be able to generate hundreds of radar beams, but it is impractical to have hundreds of signal generators to simulate all the beams. Even if one can afford to buy many signal generators, it is difficult to sum them all up into one output without creating significant power loss. Because of this difficulty, a simulator often contains a few signal generators and each one is time-shared to simulate many beams. This kind of design may miss pulses. For example, if a simulator contains four signal generators, it is impossible to generate five time-coincident pulses. The signal generator has a certain settling time before producing the correct frequency, which makes the time-sharing problem more severe.

A simulator can usually generate two kinds of scenarios: static and dynamic. In the static scenario, the EW receiver is at a fixed position on the battlefield. The types of radars stay the

same, but they can be turned on and off and their beams can scan periodically. In a dynamic environment, the EW receiver is moving on the battlefield. Hence the types of radars can appear and disappear along the path of the receiver.

Testing an EW receiver through a simulator faces one major problem. If the receiver misses a pulse or reports an extra pulse, one does not know whether it is a receiver problem or a simulator problem. Two approaches are often used to resolve this. The first one is to dedicate one signal generator to a certain radar, thus this radar is guaranteed to not miss a pulse. The second approach is to use a narrowband superheterodyne receiver to receive a certain signal and compare its result against the output of the EW receiver. The second approach can be used to check both missing pulses and spurious responses generated by the EW receiver.

The simulator test can only produce a qualitative result because it is difficult to report the result in a quantitative manner. However, if the receiver is operated with a real-time signal processor, quantitative results can be obtained from a system operation performance point of view. If the overall system including an EW receiver and an EW processor does not perform satisfactorily, the simulator test can be used to test the EW receiver to determine whether the problem is caused by the receiver or the processor. Under this condition, the output of the EW receiver must be recorded at high speed on a pulse-by-pulse basis.

18.21 Field Test

It may appear that the previous tests can cover all the possible performances of a receiver, but past experience suggests this is not necessarily true. As mentioned before, it is impossible to test an EW receiver under all signal conditions. A field test is used to test the receiver under real radar signals. The receiver must be connected to an antenna or antenna array and placed in the field. The field test can be divided into controlled and uncontrolled tests.

In the controlled field test, one can order certain radars in the field to start and stop transmitting. Thus this test is somewhat similar to an anechoic test. In order to assess the performance of the receiver, a superheterodyne receiver is needed to confirm the received signal. For example, if the receiver reports several signals at certain frequencies with a certain PRF, the PRF can be generated from the TOA measurements. The superheterodyne receiver can be tuned to these frequencies. Each signal should be checked separately to make sure that the frequency and PRF reported by the EW receiver are correct.

In the uncontrolled field test, the EW receiver is taken to an area where anticipated radar signals might be received. One would have no prior knowledge of when and what types of radars will be under operation. It is very difficult to perform this test. If a receiver reports certain data, these are difficult to confirm, especially when the radar pulses are emitted for a short time duration. Under this condition, it is even difficult to use a superheterodyne receiver to check the result because there might not be enough time to tune the superheterodyne receiver to the desired frequency. Sometimes two different EW receivers can be used to check the results against each other. If two different results are reported, however, it is difficult to determine which receiver reported the correct results.

One useful bit of information obtained from past field tests will be briefly mentioned here. A radar beam scanning past an EW receiver always produces the condition that the signal strength is crossing the sensitivity level. If the receiver threshold is not designed with a hysteresis loop, the receiver may report multiple pulses every time the radar signal crosses the threshold.

Let me conclude this book with a funny situation that occurred during a field test. After tuning the receiver through some frequency ranges to search for a signal, finally a signal was received. However, this signal was later confirmed as the frequency generated by the local oscillator of another receiver.

References

- [1] Hahn P, Jeruchim C. ‘Developments in the theory and application of importance sampling’. *IEEE Transactions on Communications* 1987;**35**(7):706–714.
- [2] Jeruchim MC, Hahn P, Smyntek KP, Ray RT. ‘An experimental investigation of conventional and efficient importance sampling’. *IEEE Transactions on Communications* 1989;**37**(6):578–587.
- [3] Jeruchim MC, Balaban P, Shanmugan KS. *Simulation of Communication Systems*. New York: Plenum Publishing; 1992.
- [4] Bahr RK, Bucklew JA. ‘Quick simulation of detector error probabilities in the presence of memory and nonlinearity’. *IEEE Transactions on Communications* 1993;**41**(11):1610–1617.
- [5] Tsui JBY. *Microwave Receivers with Electronic Warfare Applications*. New York: John Wiley & Sons; 1986.
- [6] Xia W. Wright State University. Private communication.

Appendix 18.A

```
% file name: falseala.m
% Find probability of false alarm under normal operating conditions( signal
% is not present).

clear

fprintf(' n The Test of False Alarm Tfa under normal conditions: n');
fprintf(' ----- n');
M=input(' the noise source expressed in power ratio M=? in dB ');
A=input(' attenuator A=? in dB ');
dcac=input(' dc-coupled (1) or ac-coupled (0) ? ');
NF=input(' noise figure of the receiver NF=? in dB ');
Br=input(' RF bandwidth Br in Hz=? ');
Bv=input(' video bandwidth Bv in Hz=? ');
Tfa_i=input(' with noise generator: Tfa=? in sec. ');

r=Br/Bv;
pfa_i=1/(Tfa_i*Br);
F=A+NF;
R=1+10^((M-F)/10);
N=R;
```

```

if dcac==0,
k1=1;
else
k1=N;
end

k2=(N^2)/sqrt(1+(r^2)/2);
k3=4*(N^3)/(2+3*(r^2)/4);
k4=(k1^2)/k2;

vt=10;
vt1=0;
vt2=0;
for ii=1:20,
fprintf(' n%g n',ii);
k5=(vt-k1)/sqrt(k2);
b1=k3*((k5^2)-1)*exp(-(k5^2)/2)/sqrt(2*pi)/6/sqrt(k2^3);
b=b1+quad('xp',k5,(vt*3)/sqrt(k2))/sqrt(2*pi);
if b<=pfa_i,
vt2=vt;
vt=(vt+vt1)/2;
else
vt1=vt;
vt=(vt+vt2)/2;
end
fprintf('vt=%g n',vt);
end

kk1=1;
kk2=1/sqrt(1+(r^2)/2);
kk3=4/(2+3*(r^2)/4);
kk5=(vt-kk1)/sqrt(kk2);

bb1=kk3*((kk5^2)-1)*exp(-(kk5^2)/2)/sqrt(2*pi)/6/sqrt(kk2^3);
pfa=bb1+quad('xp',kk5,(vt*3)/sqrt(kk2))/sqrt(2*pi);
Tfa=1/(pfa*Br);
fprintf(' n Under Normal Conditions in sec : Tfa=%g n n',Tfa);

end

```

Appendix 18.B

```

% filename xp.m
function y=xp(x)
y=exp(-(x.^2)/2);
end

```


Index

- acoustic detection systems 8
- acquisition time 119
- active EW system 7
- adaptive spectrum estimation 357–61
 - noise sensitivity 361
 - see also* high-resolution spectrum estimation
- Air Force Research Laboratory (AFRL) 2, 447
- Akaike information criterion (AIC) 341
- aliasing effect
 - band overlapping caused by 316
 - overlapping 64–5
 - on real-valued data 63, 64–5
- all-pole model 327
- all-zero model 327
- amplifier and ADC combination, noise
 - figure of 148–9
- amplifier and ADC interface 149–50
 - amplitude 149
 - dynamic range 150
 - maximum allowable ADC power 150
 - noise figure 149, 150
 - third-order intercept point 149, 150
 - third-order intermodulation 149
- amplifier chain 143
 - amplifier characteristics 153
 - in analog receiver 144
 - calculated performance 152
 - computer program and results 151
 - defined 143
 - design example 151
 - experimental setup 154–8
 - gain 143, 144
 - noise figure 141, 143, 151
 - performance 143
 - third-order intercept point 143
- amplifiers 523
 - amplifier chain 143–4, 169
 - characteristics 143–7
 - in front of ADC 117, 206, 216, 230
 - gains 206
 - input power 142, 150
 - limiting 272, 277
 - noise figures 144, 148–9
 - output noise 148–51
 - output power 141
 - third-order intercept points 143, 144, 152
 - see also* attenuators
- amplitude-aided phase comparison method 420–3, 432
- amplitude comparison system 413–14, 417, 418, 420, 441, 475, 476
 - accuracy 475
 - defined 475
- amplitudes
 - of delay line 477
 - envelope 247, 249, 250
 - FFT outputs 279
 - in frequency domain 452
 - I* and *Q* channels output 172, 192
 - imbalance 174, 175, 176, 191, 193, 194, 195, 196
 - measurement 455–8
 - of *n*-th harmonic 129
 - pulse 9, 13, 17, 19, 25, 29, 30, 31, 476–7
 - sine wave 122
 - spur 130–4
 - spur, analysis on 126–30
- analog filters
 - followed by monobit receivers 314–17
 - followed by phase comparators 310–11
 - see also* filters

- analog I and Q downconverters 175–8, 195
 - advantage 175
 - with balanced outputs 178
 - drawbacks 175
- analog points 448
- analog receivers 17
 - comparison sensitivity 118–19
 - component selection 117–18
 - input frequency 167
 - RF chain 118
 - two blocks of 543
 - see also* digital receivers
- analog-to-digital converter (ADC) 2, 22, 27, 58, 102, 117, 147–8, 168, 201, 241, 259, 267, 475–6, 499, 523
 - 1-bit 268, 270, 271
 - 2-bit 268, 274
 - 3-bit 120, 128, 129, 132, 133
 - 4-bit 131
 - 6-dB per bit 123
 - 8-bit, FFT output of 133
 - amplifier and ADC interface 149–50
 - amplifiers in front of 117
 - analog and digital receiver sensitivity 118–19
 - apparent maximum/minimum signals to 122–3
 - basic sample and hold circuit 119
 - characteristics of amplifiers in cascade 143–7
 - computer program and results 151–2
 - and demultiplexers 273
 - design example 153–4
 - dynamic range 117, 123, 139
 - dynamic range test 156–8
 - dynamic range vs. bits 134
 - experimental results 154
 - FFT chip 274–5, 279
 - high-speed outputs 2
 - ideal 120
 - ideal, performance 122
 - ideal, quantization noise 123–4
 - impact of 117
 - improvements in 3
 - input bandwidth 120–1
 - input vs. output 120, 122
 - key component selection 117–18
 - maximum and minimum signals to 122–3
 - maximum input power 147, 150
 - maximum sampling frequency 139
 - meaning of M and M' 150–1
 - monobit receiver 271–4
 - noise effects 134–6
 - noise figure and third-order intercept point 141–3
 - noise figure measurement 154–6
 - noise figure of amplifier and ADC
 - combination 148–9
 - noise level 124–5
 - notation 139–41
 - outputs, many levels missing 132
 - outputs, no levels missing 132
 - outputs, time/frequency domains of 130, 131, 132, 133
 - outputs, with input frequency between bins 133
 - output signal 128
 - in parallel 121
 - parameters 117, 147, 151
 - performance 2, 120–2, 155, 162–3, 209
 - quantization error 120, 122, 126
 - quantization noise 123–4
 - requirements on 139
 - sample-and-hold circuit 119–20
 - sampling rate 205
 - sampling window jittering effect 136–8
 - SNR vs. frequency of 138
 - speed 2
 - spur amplitudes 130–4
 - spur amplitudes, analysis on 126–30
 - spurious response 125–6
 - technology 2
 - Tektronix TKAD20C 153, 154
 - transfer characteristic function 128
 - transfer function of 121
- analog-to-information (A-to-I) 4, 447
 - amplitude measurement 455–8
 - data collection 448–9
 - frequency calculation 449–51
 - frequency measurement accuracy 453–5
 - Gold code 448
 - instantaneous dynamic range 470–2
 - probability of detection 451–3
 - probability of false alarm 451–3
 - second threshold obtained from first input signal 469–70
 - sensitivity of second signal detection 470–2
 - two-frame approach 460–8
 - two-signal frequencies, determining 458–60
 - two signals with different amplitudes without noise 463–4

- anechoic chamber tests 540, 556–7
 - defined 540
 - setup 546–7
 - see also* receiver tests
- angle of arrival (AOA) 13, 22, 475, 523
 - accuracy 16, 26
 - accuracy test 553
 - ambiguity 485, 492
 - compressive receiver used for 26
 - data collection 476
 - data resolution 25
 - estimation 492
 - information 15–16
 - narrowband system 16
 - of received pulses 29
 - resolution 25, 475, 486, 487, 491, 492
 - resolution test 557–8
 - simultaneous signal measurement 16
 - time reversal study application to 503–4
 - two-channel 517–18
- angle of arrival (AOA) measurements
 - amplitude comparison 475, 476
 - analog 498–9
 - calibration table 475
 - Chinese remainder theorem application to 494–8
 - circular antenna array, outputs from 479–82
 - digital 475, 498–500
 - Doppler frequency 476
 - Fourier transform over space domain 486–9
 - frequency sorting followed by 491–2
 - hardware considerations 498–500
 - importance of 475
 - linear antenna array, digital data from 477–9
 - methods 16, 479
 - minimum antenna spacing 492–3
 - phase comparison 475, 476
 - phase measurement with multiple antennas 485–6
 - queuing concept 476–7
 - two-dimensional Fourier transform 489–91
 - two-element phase array antenna 482–3
 - two-signal setup 558
 - through zero crossing 484–5
- angle of arrival (AOA) systems
 - phase interferometric 553, 556, 557, 558
 - phase measurement in 483, 485–6
 - queuing 477
 - two-antenna element 482
 - wideband phase measurement 491
- angle quantization cells 243, 244
- angle resolution 243–4
 - number of ADC bits vs. 244
 - quantization levels and 243–4
- antenna arrays
 - circular 479–82
 - linear 477–9
 - phase 482–3
 - problem 497
 - with weight function 489, 492
- antennas
 - diameter 493
 - one receiver shared among 499
 - spacing, minimum 492–3
 - spiral 493, 495
- application specific integrated circuit (ASIC)
 - technology 283
- attenuators
 - amplifier connection results 229
 - cascading amplifiers and 228
 - characteristics 227
 - placement 229
- autocorrelation 95–7, 338, 339, 376
 - 13-bit Barker code 376
 - autocovariance vs. 97
 - biased 98
 - defined 95
 - example 96
 - FFT application to spectrum estimation 99–101
 - input data manipulation method 332–5
 - lag of 96, 332, 338, 339
 - matrix 332, 352, 356
 - method 334–5
 - “sample” 96, 97, 328
 - spectrum estimation 98–9
 - of time series 338, 339
 - unbiased 97
- autocovariance 97
 - autocorrelation vs. 97
 - defined 97
 - “sample” 97
- autoregressive (AR) model 326–7, 328
 - autocorrelation method 334–5
 - backward prediction 335–8

- Burg method 338–41
 - covariance method 334
 - demonstration 331–2
 - equivalent circuit 328
 - input data manipulation 332–5
 - Levinson-Durbin algorithm and 330–2
 - modified covariance method 335–8
 - moving average (ARMA) 326, 327
 - order selection 341
 - plane result 328
 - power spectrum from 331
 - process results 333
 - Yule-Walker equation and 327–30
 - see also* high-resolution spectrum estimation
 - autoregressive moving average (ARMA)
 - model 326
 - averaged index method 415, 436
 - averaged periodogram 84–5
 - average linear prediction error 337, 339
 - backward prediction 335–8
 - coefficient 336, 337
 - defined 337
 - error 337
 - see also* linear prediction
 - band overlapping 315, 316
 - bandwidth
 - 3-dB 308
 - 60-dB 306, 308
 - ADC input 151
 - digital filters 317
 - digital receiver 4
 - equivalent video 208
 - filter 13, 202, 295, 305, 311, 313
 - input 270, 280, 290, 293, 295, 311, 316, 399
 - instantaneous 12, 16, 18, 22, 23, 27, 30, 31, 167
 - maximum unambiguous input 152
 - minimum filter 202
 - monobit receiver 280
 - Nyquist 510
 - optimum 12, 26
 - output 290, 316
 - processing 124–5, 148, 151
 - processing circuit 308
 - resolution 148, 151
 - RF 118, 140, 148, 151
 - sampling 107
 - system noise 13
 - Barker codes 375, 376
 - 11- and 13-bit 378–81
 - 13-bit 376, 377, 378
 - properties 376–7
 - baseband receiver frequency selection 168
 - Bessel function 212, 221
 - biased autocorrelation 98
 - bias frequency 548
 - biphase shift keying (BPSK) signals 375
 - 11- and 13-bit Barker codes, comparison of 378–81
 - basic Barker code properties 376–7
 - carrier frequency of 381
 - differentiating CW signal and 385–91
 - eigenvalue method 396–401
 - FFT method on one frame of data 395–6
 - frequency outputs from 382–3
 - generation 377–8
 - input data with two phase transitions in consecutive frames 405–7
 - long FFT for determination of 382–5
 - minimum-to-maximum ratio 383
 - one-frame eigenvalue and phase comparison methods 405
 - phase comparison method 401–5
 - phase transitions 391–4, 401–5
 - probability of detection, finding 384–5
 - Rayleigh distribution 383
 - threshold and probability of detection 381–2
- “bit reversal” 75
 - Blackman filter 133
 - Blackman-Tukey method 98–9
 - Blackman window 133, 382, 412
 - Boltzmann’s constant 14
 - Bragg cell receivers 20–1
 - disadvantages 19
 - encoder design 21
 - input RF signal 20
 - interferometric approach 21
 - optical portion 21
 - brute force approach 441–2
 - BT* coefficients 265
 - Burg method 338–41
 - defined 339–41

- high spectrum peaks 339
- result 339
- spectrum generated by 338–41
- window function 339
- calibration table 475, 521
- carrier frequency 13
- cascade, characteristics of amplifiers in 143–7
- Cauchy's principal value 180
- causality 184
- changing complex zeros into real zeros 262
- channelization 167
 - approach 302
 - digital 285, 305
 - goal 380
 - with polyphase filter 301–4
- channelized approach 267, 305, 314
- channelized IFM receiver 26–7, 310
- channelized receivers 19–20
 - amplifiers 19
 - amplitude comparison scheme 16, 20
 - center frequency 19
 - design 20
 - IFM 26–7
- characteristic equation 211
- characteristic function 209–10, 211
- Chinese remainder theorem 494–8
 - for antenna array problem 497
 - application to AOA measurement 495–7
 - defined 494
 - examples 494–5
 - practical considerations 497–8
 - problems 497–8
- chip time 375, 377–8
- chirp rate 13, 411, 426–7, 439
 - chirp signal detection vs. 444
 - of measured signal 426–7
 - sensitivity vs. 445
 - vs. S/N at the sensitivity level 444
- chirp signal 10, 411, 412, 505
 - frequency measurement with 420
 - impact on impulse response 508–10
 - impulse response generated by 506–8
 - relative error vs. initial phase of 509
- chirp signal detection 411, 412
 - with amplitude and phase comparison 423–6
- chirp rate vs. 444
- CW signal detection vs. 427, 430
- eigenvalue method 443–4
- FFT-based method 411, 412
- goals 411
- circular antenna array 480
 - azimuth/elevation information 482
 - outputs from 479–82
 - results 481
 - see also* antenna arrays
- circular convolution 66
- coarse frequency 416, 418, 419
- coherent digitizing error 127
- coherent Doppler radar frequency measurement, application to 257–8
 - defined 257
 - methods 257
 - phase error 258
 - pulse train 257, 258
 - requirements 257
- coherent radar 257
- coherent signal 55
- comb function 40–2
 - defined 40
 - Fourier series 35–7
 - Fourier transform 40, 42
- communication intercept receivers 8
- communication receivers
 - EW receivers vs. 1, 8–10
 - intercept receivers vs. 8–10
- complementary metal oxide semiconductor (CMOS) logic 274
- compressive (microscan) receivers 21–2
 - AOA measurement, used for 26
 - defined 22
 - front end 21
 - parametric encoder 22
- conjugation and conjugation symmetry 43–4
- continuous Fourier transform of a function 36
- continuous-time Fourier transform 49
- continuous wave (CW) signal 10, 257, 375, 411, 412, 447, 504
 - BPSK vs., 385–91
 - detection 427
 - frequency estimation of 417–18
 - frequency outputs from 382–3
- controlled field test 559
- convolution
 - duality of 47, 49, 52, 54
 - examples 49–57

- FFT and 286
 - linear 66, 67, 68, 69, 326, 327, 344, 353
 - mathematical representation of 97
 - theorem 46–7
 - of two continuous signals 66
- Cooley–Tukey approach 76
- correction method 193
- cosine function 71–2
- cosine wave 128, 129, 130
- cosine window 52, 70, 71–2
- covariance method 334
 - modified 335–8
 - R matrix 334
 - T matrix 334
 - see also* input data manipulation
- Cramer–Rao bound 14, 15, 26
 - defined 14
 - EW applications and 14
- criterion autoregression transfer (CAT) 341
- critical sampling rate 319
- data collection 448–9, 499
- data points grouping 203
- data resolution
 - AOA 25
 - frequency 27, 244, 245, 246, 542, 548
 - pulse amplitude 25
 - pulse width 25
- TOA 25, 287, 288
- data sample
 - probability of false alarm 204–5
 - threshold setting for 205–6
- decibel value 457
- decimated FFT 298
- decimation 110
 - applications to EW receivers 111–13
 - defined 290
 - DFT through 110–11
 - effects on DFT 293–4
 - in EW receivers 111–13
 - example of 293
 - in frequency domain 296–8
 - and interpolation 290–3
 - by M 290, 292, 294, 298
 - process 293
 - simplified method 113–15
- Defense Advanced Research Projects Agency (DARPA) program 2, 447
- deinterleaving 27–9
- delay time 515, 516
- delta functions 105
- demultiplexers 271–4
- derivative 45
 - step function 53
- detection
 - of BPSK signals 375–408
 - circuit 284
 - EW receiver approach 201–2
 - of FM signals 411–46
 - frequency domain 222–4, 226–7, 229–30, 231
 - L -out-of- N detection 216–18
 - multiple samples 208, 208–9, 212–13, 216–18
 - N sample 221–2
 - probability of 207–8, 209, 212–28, 230
 - single-sample 207–8, 213–16, 218
 - time domain 204, 222, 229, 230
- difference frequency 432
- digital approach 1–3
 - to generate I and Q channels 178
- digital correction of I and Q channel imbalance 191–3
- digital data collection: *see* data collection
- digital EW processors 8
- digital EW receiver 22–3, 27, 202
 - frequency domain detection 203–4
 - time domain detection 204
- digital filters
 - bandwidth 317
 - followed by FFT 322–3
 - followed by monobit receivers 317–18, 321–2
 - followed by monobit receivers and phase comparators 322
 - followed by phase comparators 313–14
- digital mixing/filtering 189
 - ambiguity problem 243
 - I and Q channels 242
 - with modified filters 190
 - phase discontinuity 243
- digital phase measurements 242–3
 - IFM receiver approach vs. 241
- digital receivers
 - dynamic range 259
 - EW 22–3, 27, 31
 - narrowband 23
 - three blocks of 542
 - time frequency and space-AOA transforms 544
 - see also* analog receivers

- digital signal processing
 - performance of 3
 - as problem solver 1–2
- digitized points 377, 381, 448
- digitized waveform 58
- digitizing speed 94
- Dirac delta function 37
- direct calculation 262
- direct memory addressing (DMA) mode 546
- discontinuity, reducing 69
- discrete Fourier transform (DFT) 57, 58, 62, 83, 259, 268, 287
 - advantages over FFT 78
 - aliasing effect on real-valued data 63, 64–5
 - analytic approach to 61–2
 - circular and linear convolution 65–9
 - through decimation 110–11
 - decimation effect on 293–4
 - graphical description 58–61
 - initial data accumulation 79
 - input signal bandwidth 63
 - interpolation effect on 293–4
 - inverse 62, 195
 - Kernel function of 269
 - limited frequency bandwidth 63
 - minimum norm method with 355–7
 - monobit receiver 268
 - output spectra 64
 - overlapping aliasing effect 64
 - properties 57, 74
 - sliding 79–80
 - unmatched time interval 64
 - see also* Fourier transform
- discrete Hilbert transform 181–4
 - data points 181
 - examples of 184–5
 - see also* Hilbert transform
- dithering effect, noise level determined by 124–5
- Doppler frequency measurement 257, 476
- Doppler pulse train 257
 - frequency resolution 258
- Doppler radar, pulse 57
- double difference 432
- duality 43
 - of convolution 47
 - defined 43
 - examples 49–50
 - see also* Fourier transform
- dynamic range 123, 133
- ADC 117
 - ADC impact on 117
 - defined 24
- digital receivers 259
- instantaneous 25, 125, 126, 136, 270, 272, 280, 281, 283, 470–2
- instantaneous, test 540, 542, 556
- lower limit of 117
- single signal 24, 542, 550–1, 556
- single-signal spur-free 139
- spurious free 25, 540, 555–6
- test 156–8
- third-order intermodulation 141
- two-signal spurious free test 556
- two-tone spur-free 143
- types of 24
- eigendecomposition 346
 - defined 346
 - ESPRIT method 352, 353
 - minimum norm method 355
- eigenvalues 346–8, 429
 - application of 347
 - to detect chirp signal 443–4
 - differentiating BPSK signal from CW signal through 385–91
 - minimum 347, 348
 - MUSIC method 348, 349
 - of R 347
 - sensitivity for 439
 - threshold for 436
- eigenvectors 346–8
 - application of 347
 - MUSIC method 348–9
 - of R 347
- electronic counter-countermeasures (ECCM) 7–8
- electronic countermeasures (ECM) 7
- electronic intelligent (ELINT) receiver 25
- electronic order battle (EOB) 11
- electronic support measures (ESM) 7
- electronic warfare (EW) processor 27, 267
 - deinterleaving 27–9
 - pulse repetition interval (PRI) generation 29
 - radar identification 29
 - revisiting 30
 - tracking 29–30
- electronic warfare (EW) receiver 7–8, 285, 326, 412, 447, 473, 539
 - advantages in digital EW receiver 202–4
 - analog, overview of 17–18

AOA information 15–16
 Bragg cell receivers 20–1
 channelized IFM receiver 26–7
 channelized receivers 19–20
 characteristics and performance of 24
 compressive receivers 21–2, 26
 conventional 11
 decimation method applications to 111–13
 design goals 30
 detection approach 201–2
 detection circuit 202
 development obstacles 3
 development trends 25
 diagram 523–4
 digital 22–3, 27
 dynamic range 12, 24
 false alarm rate 24
 FFT-based 523
 frequency data resolution 24
 frequency domain 203–4
 frequency information 13–15
 frequency measurement accuracy 24
 frequency measurement precision 24
 frequency resolution 25
 incoming information 9
 input RF signal 201
 instantaneous dynamic range 25
 instantaneous frequency measurement (IFM)
 receivers 18–19
 intercept systems 8
 intercept vs. communication receivers 8–9
 intermediate frequency (IF) 201
 latency time 25
 noise level 124–5
 optimum bandwidth 12
 outputs 16–17
 parameters measured by 12–13
 performance goals 31
 performance standards 539
 pulse amplitude data resolution 25
 pulse width data resolution 25
 queuing receiver 26
 requirements of 11–12
 sensitivity 12, 24
 shadow time 25
 signal environment 10–11
 single signal 24–5
 spurious free dynamic range 25
 theoretical problem solutions 26
 throughput rate 25

TOA data resolution 25
 traditional 10
 two simultaneous signals 25
 electronic warfare (EW) systems
 different functions of 9
 ECM 8
 functions 9
 input signal response 11
 intercept 8
 uses 7
 emitter coupled logic (ECL) 274
 encoder 285
 encoder description 524–5
 encoding circuit 285
 envelope
 amplitude 247, 249, 250
 detector 210
 time-dependent 247
 time vs. 247, 248
 erroneous signal 281
 error frequency 423, 461, 547
 calculation of 256, 257
 defined 416, 454
 maximum variation of 257
 variation of 257
 ESPRIT method 350–3
 defined 350
 eigendecomposition 352, 353
 input frequencies 353
 matrices definition 352
 program 502–4
 steps 350–3
 see also high-resolution spectrum estimation
 estimation of signal parameters via rotational
 invariance techniques (ESPRIT): *see*
 ESPRIT method
 even function 42–3
 experimental amplifier chain setup
 dynamic range 156–8
 noise figure measurement 154–6
 see also amplifier chain
 false alarms, in frequency domain 225–6
 based on multiple data points 208
 as binomial distribution problem 209
 measurement setup 550
 number of 216
 probability of 204–5, 208, 213–14, 220,
 225–6
 rate 24

- test 549–50
- threshold 277
- time 204–5
 - see also* probability of false alarm
- fast Fourier transform (FFT) 57, 73, 74, 76, 83, 102, 202, 267, 285, 325, 375, 378, 381, 492, 505, 525
 - 32-point 298, 301, 302, 308, 321
 - 64-point 244
 - 128-point 86, 87
 - 256-point 270, 275, 294, 296, 298
 - 512-point 133
 - actual frequency determination by 94–5
 - application to spectrum estimation 99–101
 - based electronic warfare (EW) receiver 523
 - basic chip design 274–5
 - butterflies 75
 - complexity, reducing 268
 - convolution operations and 286, 287
 - Cooley-Tukey diagram 76
 - CW and BPSK signals differentiation by 388–91
 - decimated 293–4, 298–301
 - derivation 74–6
 - to determine BPSK signal 382–5
 - DFT advantages over 78
 - digital filter bank followed by another FFT 322–3
 - effect with interpolated data 294
 - on filled windows 289
 - filter banks and 286–7
 - frame time 281
 - frequency components, of Hanning window 91
 - frequency components, of rectangular window 90
 - frequency components, of sine wave 96
 - frequency determination by 94–5
 - highest output 226
 - imaginary part of 102, 106
 - of interpolated outputs 294, 295
 - inverse 76, 181
 - length 125, 126, 130, 133
 - multiple-FFT receiver 525
 - N*-point 293–4, 298
 - on one chip 267
 - on one frame of data 395–6
 - operation 270–1
 - output data rate from 288–90
 - overlapping consecutive outputs 203
 - overlapping input data and 287
 - on partially filled window (frequency domain) 289, 290
 - on partially filled window (time domain) 290
 - phase transition detection from 397
 - power spectrum 83
 - processing 245
 - real input computed by a complex 77–8
 - real part of 78
 - of rectangular window 291
 - size 523
 - in time domain 479, 491
- fast Fourier transform (FFT) chips 23
 - basic design 274–5
 - high-speed 27
 - input data to 275
 - input signal processing 276
 - layout 275
 - outputs 275
- fast Fourier transform (FFT) method 447
- fast Fourier transform (FFT) operation 412, 453
 - 128-point 427
 - design 102
 - parallel 110
 - real input computed by 77–8
- fast Fourier transform (FFT) outputs
 - ambiguity 103
 - amplitudes 276, 279
 - complexity 268
 - crossing threshold 276, 279
 - data rate from 288–90
 - of FM signal with 100-MHz delta frequency 413
 - frequency bins 132
 - of input signal 332
 - monobit receivers at 268
 - overlapping consecutive 203
 - peaks 325
 - peaks of correct frequencies 276
 - see also* fast Fourier transform (FFT)
- fast Fourier transform (FFT) results, phase measurement and 244–5
- field-programmable gate array (FPGA) 2, 283
- field test 559–60
 - controlled 559
 - uncontrolled 559
 - see also* receiver tests
- filter banks 286
 - analog, followed by monobit receivers 314–17
 - building 286
 - designed 309

- design methodology 294–5
- digital 313, 318
- and FFT 286–7
- followed by phase comparators 310
- shape 285
- two signal conditions 306
- filters
 - analog 304, 310–11, 314–17
 - bandwidth 20, 202, 295, 305
 - Blackman 133
 - design 23, 189, 271, 286, 303
 - FIR 182, 184, 286, 303
 - four-tap Butterworth 505–6
 - IIR 286, 327
 - impulse function of 299
 - inverse 515, 521
 - with limiting skirt 307
 - lowpass 171, 188, 189
 - with narrower skirt 307
 - output 298–301
 - oversampling 321
 - polyphase 301–4
 - steady state 185
 - time reversal 503, 521
 - transient effect 305
 - in wideband channelized receiver 305
 - with wider skirt 307
- filter shape selection 306–9
- final prediction error (FPE) 341
- finite impulse response (FIR) filter 182, 286, 303, 327
- FM (chirp) pulsed signal 10
- FM signals: *see* frequency modulated (FM) signals
- forward prediction
 - coefficients 337
 - defined 336
 - error 337
 - see also* linear prediction
- Fourier series 37
 - comb function 40
 - and continuous Fourier transform 35–7
 - defined 35
 - exponential 353
 - forms 35
- Fourier transform 35, 47, 172, 173, 211
 - comb function 40–2
 - complex plane 50
 - concept 74
 - conjugation and conjugation symmetry 43–4
 - continuous 35–7
 - convolution 46–7
 - cosine function 71–2
 - cosine wave 51, 55
 - derivative 45
 - discrete: *see* discrete Fourier transform (DFT)
 - duality 43
 - even and odd functions 42–3
 - examples 49–57
 - fast: *see* fast Fourier transform (FFT)
 - frequency domain 38, 39, 40, 50, 52
 - frequency shift 45
 - Gaussian function 71, 72
 - generalized cosine window function 52
 - generalized Hamming window 73
 - impulse function 37–9
 - integral 45–6
 - inverse 39, 43, 44, 47, 53, 59
 - isosceles triangle 59
 - linearity 42
 - over space domain 486–9
 - Parseval's theorem 47–9
 - properties of 42–9
 - quadrature phase shift 77
 - rectangular function 37, 38, 70, 71
 - related operations 83–115
 - RF pulse train 54, 55, 56
 - scaling 44
 - signal digitization 57–8
 - sine function 37, 50, 64, 65, 180
 - time domain 38, 39, 40, 52, 491
 - time shift 44
 - two-dimensional 489–91
 - windowed cosine function 51
 - window functions 69
- Fourier transform-related operations 83
 - actual frequency determination by FFT 94–5
 - autocorrelation 95–7
 - autocorrelation spectrum estimation 98–9
 - decimation method applications in EW receivers 111–13
 - discrete Fourier transform (DFT) 110–11
 - FFT, application of 99–101
 - peak position estimation: *see* peak position estimation
 - periodogram 83–5
 - simplified decimation method 113–15
 - sub-Nyquist sampling scheme 101–4, 107–10
 - sub-Nyquist sampling system 104–7
 - zero padding 86–8

- fractional time delay 514–16
- frequency
 - accuracy 547–8, 550
 - accuracy test 547–8
 - bias 548
 - calculation 449–51
 - changing 412
 - channelization 285–323
 - determination by FFT 94–5
 - difference 432, 433, 434
 - error 257, 416, 418, 419, 423, 424, 458, 460, 547, 548
 - filter cutoff 188
 - information 13–15
 - information, obtaining 204
 - instantaneous 172, 241, 247–8, 249
 - local oscillator 176, 178, 187
 - measurement 416–20, 453–5
 - obtaining through averaged index method 436
 - periodic variation 249
 - phase vs. 104–5, 106
 - precision test 548–9
 - sampling 99, 126, 148, 187, 192
 - sensitivity as function of 550–1
 - shift 45
 - sorting 491–2
 - spurious 169
 - test signal 193
- frequency channelization 285
 - analog filters: *see* analog filters
 - basic considerations 305–6
 - decimated FFT 298–301
 - decimation and interpolation 290–4
 - decimation in frequency domain 296–8
 - digital filters: *see* digital filters
 - FFT operation 287, 288–90
 - filter bank design methodology 294–5
 - filter banks and FFT 286–7
 - filter shape selection 306–9
 - increasing output sampling 319–21
 - monobit receivers followed by phase comparators 311–13
 - through polyphase filter 301–4
 - processing after 304–5
- frequency conversion 168–70
 - achieving 168
 - I and Q channels 171–2
 - purpose 168
 - upconversion 169
- frequency data detection
 - comments on 231
 - examples on 229–30
 - input signal conditions in 226–7
 - see also* detection
- frequency data resolution 304
- frequency determination, zero crossing used for 258–9
- frequency difference threshold 526
- frequency domain
 - ADC outputs 130, 131, 134, 136, 139
 - amplitude in 452
 - cosine window 72
 - decimation in 296–8
 - Gaussian window 71, 72
 - generalized cosine window 52
 - Hamming window 185
 - noise in 225
 - noise variance 225
 - probability density in 225
 - probability of detection in 207, 226, 227–9
 - probability of false alarm in 225–6
 - rectangular function 89
 - rectangular window 70, 71
 - response of Parks-McClellan window 300
 - S/N in 228–9
 - window performance and 70
 - see also* time domain
- frequency domain detection 203–4, 222–4
 - comments on 231
 - defined 203
 - examples of 229–30
 - input signal conditions in 226–7
 - suggested approach to 224
- frequency downconversion 169
- frequency downconverters 167
 - analog I and Q downconverters 175–8
 - baseband receiver frequency selection 168
 - digital approach to generate I and Q channels 178
 - discrete Hilbert transform 181–5
 - Hilbert transform 179–81
 - I and Q channel imbalance 191–7
 - imbalance in I and Q channels 172–5
 - in-phase and quadrature-phase channel conversion 171–2
 - narrowband I and Q channels 185–6
 - wideband I and Q channels 187–9, 189–90
 - see also* frequency conversion

- frequency encoder 267
 - defined 270
 - design 275–6
 - functional block 276
- frequency errors 458, 459–60
 - on two-frames signal detection 462
 - for two signals of the same amplitude 460
- frequency measurement
 - accuracy 453–5
 - coherent Doppler radar 257–8
 - rms values 549
 - on two signals 249–50
- frequency modulated (FM) signals 10, 241, 411
 - amplitude comparison 413–14, 420–3
 - brute force approach 441–2
 - chirp rate of measured signal 426–7
 - chirp signal detection 423–6
 - eigenvalue method 429, 436, 443–4
 - frequency measurement 416–20
 - long weak signal, detecting 427–9
 - L*-out-of-*N* method 429, 431–6, 437
 - phase comparison 414–16, 420–3
 - potential problems with 412–13
 - sensitivity study 436–41
 - summation and maximum summation
 - methods 429–31
 - thresholds 436, 437
- frequency resolution 13, 14
 - cell 294, 548
 - data 334, 548
 - defined 25
 - generated from short delay 241
 - uniform 286
- frequency shift 45
- frequency upconversion 169
- gain
 - amplifier 141
 - amplifier chain 143–4, 147
 - correction coefficient for 192
 - highest 147
 - noise figure vs. 155
 - processing 412
 - required RF chain 271
- Gaussian distribution 207, 386, 430, 431, 436
- Gaussian function 71
- Gaussian window 71
 - defined 71
 - frequency domain 71
 - response of 72
 - sidelobes and 71
 - time domain 71, 72
- generalized cosine window function 52
 - time domain 64
- generalized Hamming window 73
- general pole-zero form 326
- global positioning system (GPS) 68
- Global Positioning System (GPS) receiver 267
- Global Position System (GPS) signal 375
- Gold code 375, 448, 460–3
- half-cycle sine wave 133
- Hamming window 52, 73, 185
 - FFT output of 287, 292
 - FIR filter with 184
 - in frequency domain 185
 - generalized 73
 - response of 73
 - sidelobes 73
 - time domain, 290
- Hanning window 72
 - frequency components from FFT of 90, 91
 - peak position estimation for 88–92
 - in time domain 90
- hard limiter effect 270–1
- high-resolution spectrum estimation 325
 - adaptive 357–61
 - autocorrelation method 334–5
 - autoregressive (AR) method 326–7
 - backward prediction 335–8
 - Burg method 338–41
 - covariance method 334, 335–8
 - eigenvectors and eigenvalues 346–8
 - ESPRIT method 350–3
 - input data 332
 - input data manipulation 332
 - least squares Prony's method 345–6
 - Levinson–Durbin recursive algorithm 330–2
 - methods 335–8
 - minimum norm method 353–5
 - minimum norm method with DFT 355–7
 - MUSIC method 348–50
 - order of operation determination 326
 - order selection 341
 - outputs 326
 - prediction model 327

- Prony's method 342–5
- quality improvement 332
- visual displays 326
- Yule–Walker equation 327–30
- Hilbert transform 179–81
 - defined 179
 - direct form of filter for 184, 185
 - discrete 181–4
 - filter input/output 169
 - imaginary part 183
 - result 180
 - in time domain 179
- histogram
 - of frequency estimation errors 417, 419
 - of frequency output with noise as the input 432
 - obtained from convolution of uniform distribution 434
 - of peak noise outputs 431
- hold mode 119
- I and Q channel imbalance
 - digital correction of 191–3
 - wideband digital corrections of 193–7
- I and Q channels 108, 109, 112
 - amplitude error 191
 - analog downconverters 175–8
 - balanced output 178
 - conversion 171–2
 - digital approach to 178
 - digitizing 242
 - graphical representation of outputs 175
 - imbalance in 167, 172–5
 - for instantaneous frequency determination 172
 - narrowband 185–6
 - outputs 189, 191, 246
 - outputs, complex form 171, 192
 - outputs, from filters 191
 - phase error 191
 - relative phase between 178
 - sub-Nyquist sampling with 109
 - wideband 187–9
 - wideband, filter design hardware considerations for 189–91
- IF receivers 167
- image of the true signal 173
- imbalance in I and Q channels 172–5
- importance sampling 549–50
- impulse function 37–9, 503, 505, 510, 521
 - convolution 46
 - defined 37
 - of filter 287, 299
 - representation 37
 - time-shifted 39
- impulse response 503
 - defined 505
 - effect of chirp signal on determining 508–10
 - measurement 505–6
 - quality of 506–8
- infinite impulse response (IIR) filter 286, 327
- infrared intercept receivers 8
- in-phase channel conversion 171–2
- in-phase downconversion 167
- input data generation from one set of input measurements 520–1
- input data manipulations 332
 - autocorrelation method 334–5
 - covariance method 334
- input data rate 288, 304
- input sampling rate 301, 303
- input signal recovery from output signal 511
- instantaneous dynamic range 280, 281, 471, 533, 557
 - free 31
 - high 19, 316
 - of receiver 125
 - test 556
 - see also* dynamic range
- instantaneous frequency
 - defined 247
 - I - Q channels for determining 172
 - maximum value 248
 - measurement 241
 - minimum value 248
 - negative 248, 249
 - time vs. 249
- instantaneous frequency measurement (IFM)
 - receiver 14, 18–19, 102, 168, 241, 272, 304–5, 310, 541
- channelized 26–7
- defined 18
- digital phase measurement vs. 241
- limiting amplifiers 272
- narrowband 310
- performance 541
- phase relation 104
- phase vs. frequency 105
- simultaneous signal processing and 17, 18
- sub-Nyquist sampling vs. 102

- integral 45–6, 48
- intercept point, third-order 142, 144
- intercept vs. communication receivers 8–9
- interface 139
- interleaving 27, 28
- intermediate frequency (IF) outputs 242
- intermediate frequency (IF) receiver 167
- intermediate output frequencies 169
- intermodulation, third-order 141–3
- interpolation 290–3
 - defined 290
 - effects on DFT 293–4
 - example of 293
 - process 293
 - representation 292
 - zero padding 294
- inverse continuous Fourier transform 36
- inverse DFT (IDFT) 62, 66
- inverse FFT (IFFT) 76
- inverse Fourier transform 39, 43, 44, 47, 211, 490
 - in discrete form 490
 - finding 53
 - for quadrature phase shift 53
- isosceles triangle, Fourier transform 59
- iteration, peak position estimation through 92–4
- jitter effect 136–8
 - distribution 138
 - measurement 136
 - sampling window 136–9
- joint density function 207
- kernel function 269
 - changing 281
 - of DFT 268, 269
 - digitized by 1 bit 268
 - increasing number of bits in 281
 - values, digitizing 283
 - values, moving 282
- Kronecker delta 347
- laboratory tests
 - importance 540
 - preliminary considerations 541–3
 - results presentation 543
 - setup 544–6
 - see also* receiver tests
- laser intercept receivers 8
- latency
 - tests 554
 - time 25, 554
- leakage effect, of sine wave 64
- least mean square (LMS) approach 329
- least significant bit (LSB) 123
- least squares method 345–6
- least squares Prony's method 345–6
 - performance 345
 - results 345
- Levinson–Durbin recursive algorithm 330–2
- L'Hospital's rule 42
- limiting amplifiers 19, 272
- Lin, David 503
- linear antenna array
 - data output from 489
 - digital data from 477–9
 - four-element 485
 - q th element 478, 479, 480, 481
 - results 481
 - with uniform spacing 486
- linear chirp signal 411, 422, 427
- linear convolution 97
 - output 117
 - of sequence 66
 - see also* convolution
- linearity 42
 - defined 42
 - use example 50
- linear prediction 327
 - average error 337, 339
 - backward 335–8
 - Burg method 338–41
 - forward 337, 338, 339
- Liou, Lihyeh 503
- local oscillator frequency 171, 176, 178, 187, 188
- long weak signal, detecting 427–9
- look-through time 11
- L -out-of- N approach 208, 212, 220, 222, 230, 429, 437
 - S/N 221, 230
 - summation method vs. 222
 - threshold for 431–6
- lowpass filters 171, 188
- low probability of intercept (LPI) radars 10
- MATLAB program 299, 309, 339, 458
- maximum entropy method (MEM): *see* Burg method
- maximum frequency bin 416

- maximum power 152
- maximum summation method, threshold setting for 429–31
- microscan receivers 224
- minimum antenna spacing 492–3
- minimum description length (MDL) 341
- minimum norm method 353–5
 - defined 353
 - eigendecomposition 353
 - frequency response 355
 - procedure 353–5
 - result 355
- minimum norm method with DFT 355–7
 - defined 355
 - frequency response 357
 - signal matrix 356
 - signal subspace matrix 356
 - steps 356
- missing bits 134
- missing signals vs. amplitude differences 463–4
- mixer
 - circuit 169
 - spur chart 170
- modified Bessel function 207, 212, 221
- modified covariance method 335–8
 - data use 338
 - defined 336
 - noise sensitivity 338
 - see also* covariance method
- monobit receivers 267, 274, 305
 - ADC 271–4
 - analog filters followed by 314–17
 - bandwidth 280
 - basic FFT chip design 274–5
 - channelized 316, 317
 - chip layout 283–4
 - components 270–1
 - deficiency 270, 281
 - demultiplexer 271–4
 - design criteria 270
 - digital filter bank followed by phase comparators and 322
 - digital filters followed by 317–18, 321–2
 - followed by phase comparators 311–13
 - frequency encoder design 275–6
 - idea 269
 - input frequency range 280
 - instantaneous dynamic range 280
 - original concept of 268
 - outputs 279
 - possible improvements 281–3
 - preliminary performance of 279–81
 - probability of false alarm 277
 - RF chain 271–4
 - testing 279–80
 - threshold selection 277–9
 - timing analysis 283, 284
- moving average (MA) model 327
- multiple fast Fourier transform electronic warfare (FFT EW) receiver 523, 524
 - encoder description 524–5
 - EW receiver diagram 523–4
 - PDW association 527
 - PDW combination 527–31
 - simulation results and discussions 531–6
- multiple-sample collection 208–9
 - based on summation 212–13
 - detection based on 208
 - detection scheme for 208–9
 - example 216–18
 - see also* detection
- multiple signal classification method: *see* MUSIC method
- multirate system 303
- MUSIC method 348–50, 476
 - calculation 350
 - defined 348
 - eigenvalues 348
 - eigenvectors 348
 - idea behind 349
 - program 350
 - steps 349–50
- narrowband I and Q channels 185–6
- narrowband receivers, digital 23, 476–7
- noise
 - in ADC 134–6
 - floor 13
 - in frequency domain 223
 - probability density of 205
 - sensitivity and 151
 - spur reduction and 134, 136
 - total 138
- noise dithering 125
 - defined 125
 - signal detected with 125
- noise effects 134–6
 - EW receiver 134

- noise figure
 - of amplifier and ADC combination 148–9
 - measurement 154–6
 - of receiver 141
 - and third-order intercept point 141–3
- noise level determined by processing bandwidth
 - and dithering effect 124–5
- Northrop Grumman DARPA project 447
- N sample detection 221–2
- Nyquist bandwidth 510
- Nyquist rate 103
- Nyquist sampling
 - criterion 23, 270
 - frequency 270
 - rate 103
 - theorem 63, 187, 493
- odd function 42–3, 48
- one and two-signal tests 543
- one-frame eigenvalue method 405
- one-frame test 394, 395
- optimizing selection of threshold 220–1
- order selection 341
 - empirical approach 341
 - methods 341
 - results 341
- output filter
 - shape from decimated FFT 298–301
 - widening with weighting function 298–301
- output sampling rate 290
 - changing 318, 319–20
 - equals input sampling rate 301
 - see also* sampling rates
- overlapping input data 287–8
- oversampling ratio 319
- Parks–McClellan window 298
 - frequency domain 299
 - response of 298–9
 - time domain 299
- Parseval’s theorem 47–9
- passive EW system 7
- peak 133, 381
- peak position estimation
 - for Hanning windows 90–2
 - through iteration 92–4
 - for rectangular window 88–90
- peak summation method 428, 431
- percentage error 258, 508
- periodic equation 249
- periodic function 37
- periodic real-valued function 35
- periodogram 83, 98
 - averaged 84–5
 - of data contaminated with noise 85
 - of noise-only input 84
 - square roots of 84–5
- phase
 - adjustment 516–17
 - angle 517–18
 - antenna array, two-element 482
 - between I and Q channels 178
 - comparison methods 405, 420–2, 437
 - comparison system 483
 - correction coefficient for 192
 - of delay line 477
 - difference 103, 104, 106, 243, 245
 - difference angles 243
 - discontinuity 243
 - error, in coherent radar measurement 258
 - frequency vs. 104, 105
 - instantaneous 247
 - transitions 375, 376, 391, 394, 398–9, 401, 403–4
 - unwrapping 243
- phase comparators 312
 - analog filters followed by 310–11
 - digital filter bank followed by monobit receivers and 322
 - digital filters followed by 313–14
 - monobit receivers followed by 311–13
- phase comparison 394, 405, 414–16, 418, 475, 476
 - amplitude-aided 420–3
- phase measurement 241, 492
 - angle resolution and quantization levels 243–4
 - application to coherent Doppler radar 257–8
 - digital 242–3
 - and FFT results 244–5
 - from first/last data points 245, 246
 - frequency measurement on two signals 249–50
 - with multiple antennas 485–6
 - scheme application 245–6
 - two simultaneous signals, analysis of 246–8
- wideband AOA system 491

- phase relation
 - IFM receiver 104
 - in sub-Nyquist sampling 104–7
- phase shifter, 90-deg 176
- phase shift keying (PSK) signals 241, 375
- phase shifts 106, 521
 - negative 105
 - positive 105
- photodetector array 20
- polyphase coded signals 10
- polyphase filters 302
 - channelization through 301–4
 - individual 303
 - operation of 301
 - time resolution by 303
- power Bragg cell receiver 20
- power spectrum 83–4, 87, 92, 98, 381
- prediction error power 331
- prediction model 326, 327
- preliminary tests 547
- probability density
 - of envelope 205
 - in frequency domain 225
 - of noise 205
- probability density function 205, 207, 418, 430, 432
 - and characteristic function 209–10
 - combined 210
 - of sum of random variables 209
 - of sum samples with square law detector 210–12
- probability of detection 381, 382, 386, 390, 391, 397–8, 405, 451–3, 471
 - 90% criterion 216, 230
 - in frequency domain 226
 - for multiple samples 208
 - for signal conditions 227
 - for single-sample detection 207–8
 - see also* detection
- probability of false alarm 204–5, 208, 384, 390, 429, 441, 451–3, 469
 - calculation 225, 230
 - determining 204, 205, 208, 212, 216
 - in frequency domain 225–6
 - one data sample 204–5
 - representation 209, 214
 - see also* false alarms, in frequency domain
- processing bandwidth, noise level determined by 124–5
- processing gain 412
- Prony's method 342–5
 - defined 342
 - deriving 342
 - least squares approach 345–6
 - matrix form of equation 344
 - origination 328
 - results 342
 - S/N and 345
 - steps 344–5
 - through z transform 345
- pulse amplitude (PA) 13
 - data resolution 25
 - measurement 551–2
- pulsed biphasic shift keying (BPSK) 10
- pulse descriptive words (PDWs) 8, 9, 118, 202, 523, 525–6, 541
 - association 527
 - combination 527–31
 - passing 16
 - TOA readings in 16
 - typical format 17
- pulse Doppler radar 57
- pulse interleaving 28
- pulse repetition frequency (PRF) 10, 42, 553
- pulse repetition interval (PRI) 10, 29, 257, 553
 - agile capability 10
 - error PRI 554
 - generation 29
 - of pulse train 54
- pulse width (PW) 13, 241, 411, 505, 523–5, 527, 529, 530
 - data resolution 25
 - difference threshold 526
 - estimation 530–1
 - measurement 305, 551–2
 - measurement accuracy 29
 - minimum 13
 - multipath problem and 305
 - threshold 526
- Qiu, Robert 503
- quadrature-phase channel conversion 171–2
- quadrature-phase downconversion 167
- quadrature phase shift 53
 - defined 53
 - Fourier transform 53
- quantization error 120, 122, 126

- quantization levels 120, 243–4
 - angle resolution and 243–4
 - probability of noise crossing 216
 - as threshold 209, 219
- quantization noise, of ideal ADC 123–4
- quantization power 147
- quantization steps 134
- “quasi-transfer” function 195
- queuing concept 476–7
 - in AOA system 485–6
 - defined 485
- queuing receiver 26
- “rabbit-ear effects,” avoiding 524, 525
- radar identification 29
- radar intercept receivers 8
- radar pulse, parameters in 12
- radio frequency (RF) 540
 - amplifiers 3
 - bandwidth 118
 - components 18
 - conversion into video signals 304
 - cycles 377, 378, 381
 - front end 18
 - input signal 2, 13
 - pulse train 49, 54, 55, 56
 - synchronized 377–8
 - see also* radio frequency (RF) chain
- radio frequency (RF) chain 267
 - arrangement 272
 - design in wideband receiver 271–2
 - frequency plan of 273
 - monobit receiver 267–70, 275, 279, 280
 - photograph 273
 - required gain of 274
 - wideband 279
 - see also* radio frequency (RF)
- radix 2 FFT 75
- raised cosine window 73
- random modulation preintegrator (RMPI) 447
- Rayleigh distribution 205, 214, 381, 383, 384, 431
- real-valued function 35
- real zero crossings, generating 260–2
- receiver tests 539
 - anechoic chamber test 546–7, 556–7
 - AOA accuracy test 553
 - AOA resolution test 557–8
 - false alarm test 549–50
 - field test 559–60
 - frequency accuracy test 547–8
 - frequency precision test 548–9
 - instantaneous dynamic range test 556
 - laboratory test setup 544–6
 - latency tests 554
 - one and two-signal 543
 - PA and PW measurements 551–2
 - preliminary considerations 541–3
 - preliminary tests 540, 542–3, 547
 - sensitivity and single-signal dynamic range 550–1
 - shadow time 554
 - simulator test 558–9
 - single-signal frequency test 547–9
 - software simulation 543–4
 - throughput rate 554
 - TOA test 553–4
 - two-signal 540, 547
 - two-signal frequency resolution test 555
 - two-signal spurious-free dynamic range test 555–6
 - types 540–1
- rectangular function 37, 38, 70, 71, 89
- rectangular window
 - FFT of 291
 - frequency components from FFT of 84, 88, 90, 91, 96, 111
 - frequency domain 90
 - frequency response after shift 71
 - frequency response near peak 71
 - peak position estimation for 88–92
 - response of 70, 71
 - time domain 89
 - see also* window functions
- relative error 508
 - defined 513
 - vs. input frequency 514
- “remez” program 299
- resolution bandwidth 151
- revisiting 27, 30
- Rician distribution 207, 214
- root mean square (rms) value 549
- sample-and-hold circuit 119–20
 - acquisition time 119
 - aperture time 119
 - aperture window 119
 - sample mode 119
 - sampling window 134, 136
 - time domain response 120
- “sample” autocovariance 96, 97
- sampling

- bandwidth 107
- frequency 86, 99, 107, 117, 118, 120, 121, 124, 126, 130, 139, 140, 148, 151, 171, 182, 187, 190, 287, 288, 304, 308, 311, 316, 317
- importance 552
- Nyquist 23, 103, 187, 315
- period 182
- periodic property of 182
- speed 471
- sub-Nyquist 101–10, 113
- sampling rates
 - ADC 205
 - input 301, 303, 319, 389
 - Nyquist 103, 271
 - output 290, 301, 303, 313, 317, 318, 319–20, 321
 - sub-Nyquist sampling and 103, 108
- sampling time 120, 448, 499, 515
- sampling window
 - jittering effect 136–8
 - sample-and-hold circuit 136
- scaling 44, 48
- second frequency method 394
- second threshold obtained from first input
 - signal 469–70
- sensitivity
 - comparison at threshold levels 219
 - defined 24
 - detection 394, 412
 - determining 467
 - and false detection 466
 - frequency domain detection and 203–4, 222
 - as function of frequency 550, 551
 - improvement of 204
 - measurement 550–1
 - measurement at one frequency value 550
 - optimum 202, 204
 - receiver 202, 204, 206, 208
 - of second signal detection 470–2
 - single-signal dynamic range and 550–1
 - study, of detection methods 436–41
 - tangential 118
- settling time 119
- shadow time 554
 - defined 25, 554
 - measurement 554
 - PW dependence 554
 - short pulse 554
- short time Fourier transform (STFT) 203, 287
- sidelobes
 - Gaussian window and 71
 - Hamming window and 73
 - spatial Fourier transform 487
 - as spurs 126
 - two-dimensional Fourier transform 491
- signal carrier frequency 523, 524, 525, 527, 529
- signal digitization 57–8
- signal power 524, 525, 526, 527, 530–1
 - estimation 531
 - estimation difference threshold 526
- signal processing, data length 30
- signal-to-noise (S/N) ratio 345
 - 3-bit ADC 128, 129, 132, 133
 - effective number of bits obtained from 117
 - effective number of bits vs. input frequency 117
 - expressed in decibels 141, 149
 - in frequency domain 228, 229
 - for L -out-of- N method 212, 220, 221, 222, 230
 - L value vs. 221
 - probability of detection vs. 215
 - in time domain 230
- sign function 53
- simplified decimal method 113–15
 - advantages 114
 - defined 113
 - disadvantages 114
 - see also* decimation
- simulator tests 541, 558–9
 - defined 558
 - EW receiver 558
 - purpose 558
 - qualitative result 559
 - scenarios 558
 - see also* receiver tests
- simultaneous signals
 - amplitude 246–7
 - analysis 246–8
 - detection scheme 246
 - monobit receiver performance on 281
- sine function 36, 37, 64, 65, 264
- sine wave 123, 124, 180, 185, 188, 190, 395, 512
 - amplitude 122
 - frequency components from FFT of 96
 - half-cycle 133
 - leakage effect of 64
 - one bit quantizer and 124

- single-frequency measurement from zero
 - crossing 250–2
- single-frequency zero crossing methods 256–7
 - experimental results 256–7
 - ill condition 253
 - simplified 253–5
 - see also* zero crossing
- single index method 415, 432
- single-sample detection
 - example of 213–16
 - probability of detection for 207–8
 - see also* detection
- single signal 24–5
 - dynamic range 24, 550–1
 - frequency test 542, 547–9
 - ill condition in zero crossing for 253
 - phase comparison system 491
 - simplified zero crossing calculation for 253–5
 - zero crossing calculation for 253–5
- single-signal test 540, 542, 547
 - frequency accuracy 547–9
 - frequency precision 548–9
 - spur-free dynamic range 139
 - see also* receiver tests
- sliding DFT 79–80, 287
- sliding FFT 203
- software simulation 543–4
- space domain 476, 479, 486, 487, 489, 490
 - discrete data in 490
 - Fourier transform over 486–9
- spatial Fourier transform 487, 488, 490, 491
 - outputs 488
 - peaks 487
 - sidelobes 487
- spectral line splitting 339
- spectral splitting 341
- spectrum display and filter bank 19
- spiral antennas 493, 495
- spread spectrum signal 394
- spur amplitudes 130–4
 - analysis on 126–30
- spur chart 169, 170
- spur-free dynamic range of an ADC 139
- spurious free dynamic range
 - defined 25
 - test 540, 555–6
 - two-signal 555–6
 - see also* dynamic range
- spurious responses 125
 - representation 128
- spurious signals 277
- spur levels 125–6
- spurs 551
 - amplitudes 130–4
 - amplitudes analysis 126–30
 - caused by quantization error 126
 - defined 125
 - FFT length and 125, 126
 - generation 150
 - higher-order 170
 - highest level vs. numbers of bits 122, 130
 - noise and 135
 - sidelobes as 126, 133
- square law detector 210–12
 - mathematical analysis 210
 - probability density function from 211
- step function 53
- sub-Nyquist method 102
- sub-Nyquist sampling 101–4
 - defined 101
 - digitized output processing 102
 - “holes” 108
 - with *I* and *Q* channels 108, 109
 - IFM receiver vs. 102
 - input band vs. output band 104
 - input signal 102
 - phase relation in 101, 104–7
 - phase vs. frequency of 106
 - problems and potential solutions in 107–10
 - with two sampling rates 109
- summation method 427
 - chirp rate for 439
 - defined 222
 - L*-out-of-*N* method comparison 222
 - threshold setting for 429–31
- summing outputs 318
- tangential sensitivity 118
- Tektronix TKAD20C 153
- third-order intercept point 142, 144
 - noise figure and 141–3
- third-order intermodulation 141, 142
 - defined 141
 - dynamic range 141
 - products 142–3, 150, 152, 154, 156, 158
- three-element phase interferometric system 495
- threshold
 - on CW signal 394
 - determination 390, 396, 399, 415–16
 - for differentiating CW against BPSK signals 390

- for eigenvalue method 390, 436
- false alarm 277
- first quantization level as 216, 217, 218
- frequency 454
- frequency difference 526
- generation procedure 383
- level selection 216, 218–20
- for L -out-of- N method 431–6
- from numerical integration equation 221
- obtaining 386, 388, 390
- optimizing selection of 220–1
- preliminary 277
- and probability of detection 381
- PW 526, 531
- PW difference 526
- second quantization level as 218, 219
- selection 208
- selection optimization 220–1
- sensitivity comparison 219
- setting, of FFT operation 389
- signal power estimation difference 526
- summation method 430, 431
- TOA estimation difference 526
- TOD estimation difference 526
- threshold level, selection of 218–20
- throughput rate 554
 - defined 25, 554
 - test 554
- time–bandwidth product 412
- time domain
 - 11-bit Barker code 380
 - 13-bit Barker code 376, 380
 - ADC outputs 131, 136
 - ADC outputs with input frequency between bins 133
 - ADC outputs with no levels missing 132
 - cosine window 72
 - data string in 288
 - delayed signal 515
 - detection 204
 - detection 204
 - FFT in 203, 204, 222
 - Fourier transform in 489–90
 - generalized cosine window 52
 - Hamming window 292
 - Hanning window 90
 - Hilbert transform in 179, 181
 - impulse function in 510
 - noise alone in 222
 - plot of two signals at same amplitude 159
 - rectangular function 89
 - rectangular window 70
 - response of Parks-McClellan window 305
 - S/N 229, 230
 - transient effect and 20
 - see also* frequency domain
 - time of arrival (TOA) 13, 16, 17, 523, 525, 527, 528, 536, 553–4
 - condition reported in reverse order 16–17
 - data resolution 25
 - difference 25
 - estimation difference threshold 526
 - readings in PDWs 16–17
 - test 553–4
 - see also* angle of arrival (AOA)
 - time of departure (TOD) estimation difference threshold 526
 - time reversal filter 503
 - time reversal study 503
 - application to AOA measurement 503–4
 - fractional time delay 514–16
 - impact on impulse response from input chirp signal 508–10
 - impact on input data length 508
 - impulse response measurement 505–6
 - input data generation from one set of input measurements 520–1
 - input signal recovery from output signal 511
 - phase adjustment 516–17
 - phase angle 517–18
 - quality of impulse response 506–8
 - results from recovered signals 512–14
 - switching the delay from input to local signal 518–20
 - time shift 44
 - defined 44
 - see also* Fourier transform
 - Toeplitz matrix equation 330–1
 - tracking 29–30
 - tracking time 119
 - transfer function 326, 327
 - ADC 120, 121
 - Hilbert 182
 - triangular window 99
 - two-dimensional Fourier transform 489–91
 - with continuous/discrete data 490
 - discrete 490
 - of input data 489
 - sidelobes 491
 - see also* Fourier transform

- two-element phase array antenna 482–3
 - zero crossing 484
- two-frame approach
 - for detecting two signals 460–3
 - issues of generating thresholds for 464–6
 - sensitivity and false detection of second detection of 466–8
- two-signal frequencies, determining 458–60
- two-signal spur-free dynamic range 143
- two signals with different amplitudes without noise 463–4
- two-signal test 542, 547
 - frequency resolution 555
 - spurious-free dynamic range 555–6
 - see also* receiver tests
- two simultaneous signals 25
- two-threshold arrangement 413, 414
- two-tone spur-free dynamic range 158
 - defined 158
- unbiased autocorrelation 97
- uncontrolled field test 559
- video signals 304
- weak signal, detecting 415, 427–9
- weapon radars 10
- weighting function 290, 295, 298, 487, 489, 491, 492
- white Gaussian noise 347
- wideband digital corrections of I and Q channel imbalance 193–7
- wideband I and Q channels 187–9
 - hardware considerations for filter design for 189–90
- wideband receiver 375, 476
- wideband RF delay lines 476
- wideband systems 1
- window functions 69, 385, 411
 - 256-point 301
 - Blackman window 133, 375, 382–4, 388, 395, 401, 412, 413, 430, 439, 523, 524, 533
 - Burg method 339
 - cosine window 71–2
 - Gaussian window 71, 72
 - Hamming window 73
 - Hanning window 72, 73, 90, 91
 - Park-McClellan window 298, 300
 - rectangular window 70
 - triangular window 99
- windowing 184
- Wright Laboratory (WL) 2
- Yule–Walker equation 327–30, 336
 - defined 327
 - from LMS approach 329
 - matrix from 329
 - obtaining 329–30
- zero crossing
 - advantages 259
 - AOA measurement through 484–5
 - calculation, for single signal 253–5
 - coefficients calculation 262–4
 - coherent Doppler radar frequency
 - measurement, application to 257–8
 - data point before 253
 - error frequency calculated from 256
 - for general frequency determination 258–9
 - ill condition in 253
 - limitation 251
 - measurement accuracy 261
 - of multiple signals 259, 261
 - n th 256
 - real 260–2
 - single-frequency 256–7
 - single-frequency measurement from 250–2
 - for single signal and remedy 253
 - spectrum analysis 259–60, 262–4, 265
 - spectrum analyzer 264–5
 - straight-line approximation, worst error 255
 - three points selected to calculate 252
 - time 253, 254, 255, 259
 - time calculation 253
 - total 261
 - true time 252, 254
- zero overlapping approach 290
- zero padding 86–8, 288, 294
 - defined 86
 - effect of 87
 - interpolation 294
 - peak frequency component selection and 88
 - scheme 71