

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

Software Receiver Design

Build Your Own Digital Communications System in Five Easy Steps

Have you ever wanted to know how modern digital communications systems work? Find out with this step-by-step guide to building a complete digital radio that includes every element of a typical, real-world communication system. Chapter by chapter, you will create a MATLAB realization of the various pieces of the system, exploring the key ideas along the way, as well as analyzing and assessing the performance of each component. Then, in the final chapters, you will discover how all the parts fit together and interact as you build the complete receiver. Can you decode the messages hidden within the received signals?

In addition to coverage of crucial issues, such as timing, carrier recovery, and equalization, the text contains over 400 practical exercises, providing invaluable preparation for industry, where wireless communications and software radio are becoming increasingly important. Various extra resources are also provided online, including lecture slides and a solutions manual for instructors.

C. Richard Johnson, Jr. is the Geoffrey S. M. Hedrick Senior Professor of Engineering at Cornell University, where he has been on the faculty since 1981. He is a Fellow of the IEEE and co-author of *Telecommunication Breakdown* (2004, with William A. Sethares) and *Theory and Design of Adaptive Filters* (2001).

William A. Sethares is a Professor in the Department of Electrical and Computer Engineering at the University of Wisconsin in Madison. He is the author of *Rhythm and Transforms* (2007) and *Tuning, Timbre, Spectrum, Scale* (2005).

Andrew G. Klein is an Assistant Professor at Worcester Polytechnic Institute. In addition to working in academia, he has also held industry positions at several wireless start-up companies.

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in
Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

Software Receiver Design

Build Your Own Digital Communications
System in Five Easy Steps

C. RICHARD JOHNSON, JR.

Cornell University

WILLIAM A. SETHARES

University of Wisconsin in Madison

ANDREW G. KLEIN

Worcester Polytechnic Institute



CAMBRIDGE
UNIVERSITY PRESS

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning and research at the highest international levels of excellence.

www.cambridge.org

Information on this title: www.cambridge.org/9780521189446

© Cambridge University Press 2011

This publication is in copyright. Subject to statutory exception and to the provisions of relevant collective licensing agreements, no reproduction of any part may take place without the written permission of Cambridge University Press.

First published 2011

A catalogue record for this publication is available from the British Library

ISBN 978-1-107-00752-9 Hardback

ISBN 978-0-521-18944-6 Paperback

Additional resources for this publication at www.cambridge.org/9780521189446

Cambridge University Press has no responsibility for the persistence or accuracy of URLs for external or third-party internet websites referred to in this publication, and does not guarantee that any content on such websites is, or will remain, accurate or appropriate.

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

To the Instructor . . .

. . . though it's OK for the student to listen in.

Software Receiver Design helps the reader build a complete digital radio that includes each part of a typical digital communication system. Chapter by chapter, the reader creates a MATLAB[®] realization of the various pieces of the system, exploring the key ideas along the way. In the final chapters, the reader “puts it all together” to build fully functional receivers, though as MATLAB code they are not intended to operate in real time. **Software Receiver Design** explores telecommunication systems from a very particular point of view: the construction of a workable receiver. This viewpoint provides a sense of continuity to the study of communication systems.

The three basic tasks in the creation of a working digital radio are

1. building the pieces,
2. assessing the performance of the pieces,
3. integrating the pieces.

In order to accomplish this in a single semester, we have had to strip away some topics that are commonly covered in an introductory course and emphasize some topics that are often covered only superficially. We have chosen not to present an encyclopedic catalog of every method that can be used to implement each function of the receiver. For example, we focus on frequency division multiplexing rather than time or code division methods, and we concentrate only on pulse amplitude modulation and quadrature amplitude modulation. On the other hand, some topics (such as synchronization) loom large in digital receivers, and we have devoted a correspondingly greater amount of space to these. Our belief is that it is better to learn one complete system from start to finish than to half-learn the properties of many.

Whole Lotta Radio

Our approach to building the components of the digital radio is consistent throughout **Software Receiver Design**. For many of the tasks, we define a “performance” function and an algorithm that optimizes this function. This

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

approach provides a unified framework for deriving the AGC, clock recovery, carrier recovery, and equalization algorithms. Fortunately, this can be accomplished using only the mathematical tools that an electrical engineer (at the level of a college junior) is likely to have, and **Software Receiver Design** requires no more than knowledge of calculus, matrix algebra, and Fourier transforms. Any of the fine texts cited for further reading in Section 3.8 would be fine.

Software Receiver Design emphasizes two ways to assess the behavior of the components of a communication system: by studying the performance functions and by conducting experiments. The algorithms embodied in the various components can be derived without making assumptions about details of the constituent signals (such as Gaussian noise). The use of probability is limited to naive ideas such as the notion of an average of a collection of numbers, rather than requiring the machinery of stochastic processes. The absence of an advanced probability prerequisite for **Software Receiver Design** makes it possible to place it earlier in the curriculum.

The integration phase of the receiver design is accomplished in Chapters 9 and 15. Since any real digital radio operates in a highly complex environment, analytical models cannot hope to approach the “real” situation. Common practice is to build a simulation and to run a series of experiments. **Software Receiver Design** provides a set of guidelines (in Chapter 15) for a series of tests to verify the operation of the receiver. The final project challenges the digital radio that the student has built by adding many different kinds of imperfections, including additive noise, multipath disturbances, phase jitter, frequency inaccuracies, and clock errors. A successful design can operate even in the presence of such distortions.

It should be clear that these choices distinguish **Software Receiver Design** from other, more encyclopedic texts. We believe that this “hands-on” method makes **Software Receiver Design** ideal for use as a learning tool, though it is less comprehensive than a reference book. In addition, the instructor may find that the order of presentation of topics in the five easy steps is different from that used by other books. Section 1.3 provides an overview of the flow of topics, and our reasons for structuring the course as we have.

Finally, we believe that **Software Receiver Design** may be of use to non-traditional students. Besides the many standard kinds of exercises, there are many problems in the text that are “self-checking” in the sense that the reader will know when/whether they have found the correct answer. These may also be useful to the self-motivated design engineer who is using **Software Receiver Design** to learn about digital radio.

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

How We've Used Software Receiver Design

The authors have taught from (various versions of) this text for a number of years, exploring different ways to fit coverage of digital radio into a “standard” electrical engineering elective sequence.

Perhaps the simplest way is via a “stand-alone” course, one semester long, in which the student works through the chapters and ends with the final project as outlined in Chapter 15. Students who have graduated tell us that when they get to the workplace, where software-defined digital radio is increasingly important, the preparation of this course has been invaluable. After having completed this course plus a rigorous course in probability, other students have reported that they are well prepared for the typical introductory graduate-level class in communications offered at research universities.

At Cornell University, the University of Wisconsin, and Worcester Polytechnic Institute (the home institutions of the authors), there is a two-semester sequence in communications available for advanced undergraduates. We have integrated the text into this curriculum in three ways.

1. Teach from a traditional text for the first semester and use **Software Receiver Design** in the second.
2. Teach from **Software Receiver Design** in the first semester and use a traditional text in the second.
3. Teach from **Software Receiver Design** in the first semester and teach a project-oriented extension in the second.

All three work well. When following the first approach, students often comment that by reading **Software Receiver Design** they “finally understand what they had been doing the previous semester.” Because there is no probability prerequisite for **Software Receiver Design**, the second approach can be moved earlier in the curriculum. Of course, we encourage students to take probability at the same time. In the third approach, the students were asked to extend the basic pulse amplitude modulation (PAM) and quadrature amplitude modulation (QAM) digital radios to incorporate code division multiplexing, to use more advanced equalization techniques, etc.

We believe that the increasing market penetration of broadband communications is the driving force behind the continuing (re)design of “radios” (wireless communications devices). Digital devices continue to penetrate the market formerly occupied by analog (for instance, digital television has now supplanted analog television in the USA) and the area of digital and software-defined radio is regularly reported in the mass media. Accordingly, it is easy for the instructor to emphasize the social and economic aspects of the “wireless revolution.” The impact of digital radio is vast, and it is an exciting time to get involved.

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

Some Extras

The course website contains extra material of interest, especially to the instructor. First, we have assembled a complete collection of slides (in .pdf format) that may help in lesson planning. The final project is available in two complete forms, one that exploits the block coding of Chapter 14 and one that does not. In addition, there are several “received signals” on the website, which can be used for assignments and for the project. Finally, all the MATLAB code that is presented in the text is available on the website. Once these are added to the MATLAB path, they can be used for assignments and for further exploration.¹

Mathematical Prerequisites

- G. B. Thomas and R. L. Finney, *Calculus and Analytic Geometry*, 8th edition, Addison-Wesley, 1992.
- B. Kolman and D. R. Hill, *Elementary Linear Algebra*, 8th edition, Prentice-Hall, 2003.
- J. H. McClellan, R. W. Schafer, and M. A. Yoder, *Signal Processing First*, Prentice-Hall, 2003.

¹ The .m scripts will run with either MATLAB or GNU Octave, which is freely available at <http://www.gnu.org/software/octave>. When using the scripts with MATLAB, the Signal Processing Toolbox is required; all scripts have been tested with MATLAB v7.10/R2010a, but are expected to work with older versions of MATLAB. For Octave, the scripts were tested with Octave v3.2.3 and the required Octave-Forge toolboxes `signal` v1.0.11, `specfun` v1.0.9, `optim` v1.0.12, `miscellaneous` v1.0.9, and `audio` v1.1.4. When using Octave, the script `firpm_octave.m` can be renamed `firpm.m` so that identical code will run in the two platforms.

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

Contents

	<i>To the Instructor . . .</i>	<i>page v</i>
	Step 1: The Big Picture	1
1	A Digital Radio	2
	1.1 What Is a Digital Radio?	2
	1.2 An Illustrative Design	3
	1.3 Walk This Way	12
	Step 2: The Basic Components	15
2	A Telecommunication System	16
	2.1 Electromagnetic Transmission of Analog Waveforms	16
	2.2 Bandwidth	18
	2.3 Upconversion at the Transmitter	20
	2.4 Frequency Division Multiplexing	22
	2.5 Filters that Remove Frequencies	23
	2.6 Analog Downconversion	24
	2.7 Analog Core of a Digital Communication System	26
	2.8 Sampling at the Receiver	28
	2.9 Digital Communications Around an Analog Core	29
	2.10 Pulse Shaping	30
	2.11 Synchronization: Good Times Bad Times	33
	2.12 Equalization	34
	2.13 Decisions and Error Measures	35
	2.14 Coding and Decoding	37
	2.15 A Telecommunication System	38
	2.16 Stairway to Radio	38
3	The Six Elements	40
	3.1 Finding the Spectrum of a Signal	41
	3.2 The First Element: Oscillators	44
	3.3 The Second Element: Linear Filters	46

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

x	Contents	
	3.4 The Third Element: Samplers	49
	3.5 The Fourth Element: Static Nonlinearities	52
	3.6 The Fifth Element: Mixers	53
	3.7 The Sixth Element: Adaptation	55
	3.8 Summary	56
	Step 3: The Idealized System	58
4	Modeling Corruption	59
	4.1 When Bad Things Happen to Good Signals	59
	4.2 Linear Systems: Linear Filters	65
	4.3 The Delta “Function”	65
	4.4 Convolution in Time: It’s What Linear Systems Do	70
	4.5 Convolution \Leftrightarrow Multiplication	72
	4.6 Improving SNR	76
5	Analog (De)modulation	80
	5.1 Amplitude Modulation with Large Carrier	81
	5.2 Amplitude Modulation with Suppressed Carrier	84
	5.3 Quadrature Modulation	90
	5.4 Injection to Intermediate Frequency	93
6	Sampling with Automatic Gain Control	98
	6.1 Sampling and Aliasing	99
	6.2 Downconversion via Sampling	103
	6.3 Exploring Sampling in MATLAB	108
	6.4 Interpolation and Reconstruction	110
	6.5 Iteration and Optimization	114
	6.6 An Example of Optimization: Polynomial Minimization	115
	6.7 Automatic Gain Control	120
	6.8 Using an AGC to Combat Fading	127
	6.9 Summary	129
7	Digital Filtering and the DFT	130
	7.1 Discrete Time and Discrete Frequency	130
	7.2 Practical Filtering	141
8	Bits to Symbols to Signals	152
	8.1 Bits to Symbols	152
	8.2 Symbols to Signals	155
	8.3 Correlation	157
	8.4 Receive Filtering: From Signals to Symbols	160
	8.5 Frame Synchronization: From Symbols to Bits	161

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

9	Stuff Happens	165
	9.1 An Ideal Digital Communication System	166
	9.2 Simulating the Ideal System	167
	9.3 Flat Fading: A Simple Impairment and a Simple Fix	175
	9.4 Other Impairments: More “What Ifs”	178
	9.5 A <i>B³IG</i> Deal	187
	Step 4: The Adaptive Components	191
10	Carrier Recovery	192
	10.1 Phase and Frequency Estimation via an FFT	194
	10.2 Squared Difference Loop	197
	10.3 The Phase-Locked Loop	202
	10.4 The Costas Loop	206
	10.5 Decision-Directed Phase Tracking	210
	10.6 Frequency Tracking	216
11	Pulse Shaping and Receive Filtering	226
	11.1 Spectrum of the Pulse: Spectrum of the Signal	227
	11.2 Intersymbol Interference	229
	11.3 Eye Diagrams	231
	11.4 Nyquist Pulses	237
	11.5 Matched Filtering	242
	11.6 Matched Transmit and Receive Filters	247
12	Timing Recovery	250
	12.1 The Problem of Timing Recovery	251
	12.2 An Example	252
	12.3 Decision-Directed Timing Recovery	256
	12.4 Timing Recovery via Output Power Maximization	261
	12.5 Two Examples	266
13	Linear Equalization	270
	13.1 Multipath Interference	272
	13.2 Trained Least-Squares Linear Equalization	273
	13.3 An Adaptive Approach to Trained Equalization	284
	13.4 Decision-Directed Linear Equalization	288
	13.5 Dispersion-Minimizing Linear Equalization	290
	13.6 Examples and Observations	294
14	Coding	303
	14.1 What Is Information?	304
	14.2 Redundancy	308

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

xii

Contents

14.3	Entropy	315
14.4	Channel Capacity	318
14.5	Source Coding	323
14.6	Channel Coding	328
14.7	Encoding a Compact Disc	339
Step 5: Putting It All Together		341
15	Make It So	342
15.1	How the Received Signal Is Constructed	343
15.2	A Design Methodology for the \mathcal{M}^6 Receiver	345
15.3	No Soap Radio: The \mathcal{M}^6 Receiver Design Challenge	354
16	A Digital Quadrature Amplitude Modulation Radio	357
16.1	The Song Remains the Same	357
16.2	Quadrature Amplitude Modulation (QAM)	358
16.3	Demodulating QAM	363
16.4	Carrier Recovery for QAM	367
16.5	Designing QAM Constellations	378
16.6	Timing Recovery for QAM	380
16.7	Baseband Derotation	384
16.8	Equalization for QAM	387
16.9	Alternative Receiver Architectures for QAM	391
16.10	The Q^3AM Prototype Receiver	397
16.11	Q^3AM Prototype Receiver User's Manual	398
Appendices		404
A	Transforms, Identities, and Formulas	404
A.1	Trigonometric Identities	404
A.2	Fourier Transforms and Properties	405
A.3	Energy and Power	409
A.4	\mathcal{Z} -Transforms and Properties	409
A.5	Integral and Derivative Formulas	410
A.6	Matrix Algebra	411
B	Simulating Noise	412
C	Envelope of a Bandpass Signal	416
D	Relating the Fourier Transform to the DFT	421
D.1	The Fourier Transform and Its Inverse	421
D.2	The DFT and the Fourier Transform	422

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)

E	Power Spectral Density	425
F	The Z-Transform	428
	F.1 Z -Transforms	428
	F.2 Sketching the Frequency Response from the Z -Transform	432
	F.3 Measuring Intersymbol Interference	435
	F.4 Analysis of Loop Structures	438
G	Averages and Averaging	442
	G.1 Averages and Filters	442
	G.2 Derivatives and Filters	443
	G.3 Differentiation Is a Technique, Approximation Is an Art	446
H	The B^3IG Transmitter	451
	H.1 Constructing the Received Signal	453
	H.2 Matlab Code for the Notorious B^3IG	455
	H.3 Notes on Debugging and Signal Measurement	459
	<i>Index</i>	460

Cambridge University Press

978-0-521-18944-6 - Software Receiver Design: Build Your Own Digital Communications System in Five Easy Steps

C. Richard Johnson, William A. Sethares and Andrew G. Klein

Frontmatter

[More information](#)**With thanks to ...**

Tom Robbins for his foresight, encouragement, and perseverance.

The professors who have taught from earlier versions of this book: Mark Andersland, Marc Buchner, Brian Evans, Edward Doering, Tom Fuja, Brian Sadler, Michael Thompson, and Mark Yoder.

The classes of students who have learned from earlier versions of this book: ECE436 and ECE437 at the University of Wisconsin, EE467 and EE468 at Cornell University, and ECE 3311 at Worcester Polytechnic Institute.

Teaching assistants and graduate students who have helped refine this material: Jai Balkrishnan, Rick Brown, Wonzoo Chung, Qingxiong Deng, Matt Gaubatz, Jarvis Haupt, Sean Leventhal, Yizheng Liao, Katie Orlicki, Jason Pagnotta, Adam Pierce, Johnson Smith, Charles Tibenderana, and Nick Xenias.

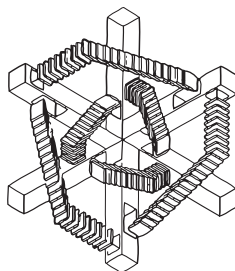
The companies that have provided data or access to engineers, and joined us in the field: Applied Signal Technology, Aware, Fox Digital, Iberium, Lucent Technologies, and NxtWave Communications (now AMD).

The many talented academics and engineers who have helped hone this material: Raul Casas, Tom Endres, John Gubner, Rod Kennedy, Mike Larimore, Rick Martin, Phil Schniter, John Treichler, John Walsh, Evans Wetmore, and Robert Williamson.

The staff of talented editors at Cambridge University Press: Phil Meyler, Sarah Finlay, Sarah Matthews, Steven Holt, and Joanna Endell-Cooper. Where would we have been without \LaTeX guru Sehar Tahir?

Those with a well-developed sense of humor: Ann Bell and Betty Johnson.

Dread Zeppelin, whose 1999 performance in Palo Alto inspired the title *Telecommunication Breakdown*. In this new, improved, expanded (and more revealingly titled) version, the legacy continues: how many malapropic Led Zeppelin song titles are there?

**Dedicated to ...**

Samantha and Franklin