

Contents

Preface

xv

1 Introduction

1.1	Digital signal processing and its benefits	1
1.2	Application areas	3
1.3	Key DSP operations	5
1.3.1	Convolution	5
1.3.2	Correlation	7
1.3.3	Digital filtering	9
1.3.4	Discrete transformation	11
1.3.5	Modulation	11
1.4	Digital signal processors	13
1.5	Overview of real-world applications of DSP	13
1.6	Audio applications of DSP	15
1.6.1	Digital audio mixing	15
1.6.2	Speech synthesis and recognition	16
1.6.3	The compact disc digital audio system	19
1.7	Telecommunication applications of DSP	23
1.7.1	Digital cellular mobile telephony	23
1.7.2	Set-top box for digital television reception	27
1.7.3	Adaptive telephone echo cancellation	28
1.8	Biomedical applications of DSP	29
1.8.1	Fetal ECG monitoring	30
1.8.2	DSP-based closed loop controlled anaesthesia	33
1.9	Summary	35

Problems	35
References	35
Bibliography	36
2 Analog I/O interface for real-time DSP systems	37
2.1 Typical real-time DSP systems	38
2.2 Analog-to-digital conversion process	39
2.3 Sampling – lowpass and bandpass signals	40
2.3.1 Sampling lowpass signals	40
2.3.2 Sampling bandpass signals	56
2.4 Uniform and non-uniform quantization and encoding	65
2.4.1 Uniform quantization and encoding (linear pulse code modulation (PCM))	66
2.4.2 Non-uniform quantization and encoding (nonlinear PCM)	68
2.5 Oversampling in A/D conversion	71
2.5.1 Introduction	71
2.5.2 Oversampling and anti-aliasing filtering	71
2.5.3 Oversampling and ADC resolution	74
2.5.4 An application of oversampling – single-bit (oversampling) ADC	78
2.6 Digital-to-analog conversion process: signal recovery	84
2.7 The DAC	84
2.8 Anti-imaging filtering	86
2.9 Oversampling in D/A conversion	86
2.9.1 Oversampling D/A conversion in the CD player	87
2.10 Constraints of real-time signal processing with analog input/output signals	90
2.11 Application examples	91
2.12 Summary	92
Problems	92
References	102
Bibliography	102
3 Discrete transforms	104
3.1 Introduction	104
3.1.1 Fourier series	106
3.1.2 The Fourier transform	109
3.2 DFT and its inverse	111
3.3 Properties of the DFT	118

3.4	Computational complexity of the DFT	120
3.5	The decimation-in-time fast Fourier transform algorithm	121
3.5.1	The butterfly	127
3.5.2	Algorithmic development	128
3.5.3	Computational advantages of the FFT	132
3.6	Inverse fast Fourier transform	132
3.7	Implementation of the FFT	133
3.7.1	The decimation-in-frequency FFT	134
3.7.2	Comparison of DIT and DIF algorithms	134
3.7.3	Modifications for increased speed	134
3.8	Other discrete transforms	135
3.8.1	Discrete cosine transform	135
3.8.2	Walsh transform	136
3.8.3	Hadamard transform	139
3.8.4	Wavelet transform	141
3.8.5	Multiresolution analysis by the wavelet method	144
3.8.6	Signal representation by singularities: the wavelet transform method	147
3.9	An application of the DCT: image compression	151
3.9.1	The Discrete Cosine transform	152
3.9.2	2D DCT coefficient quantization	153
3.9.3	Coding	153
3.10	Worked examples	154
	Problems	158
	References	160
	Appendices	161
	3A C language program for direct DFT computation	161
	3B C program for radix-2 decimation-in-time FFT	167
	3C DFT and FFT with MATLAB	170
	References for Appendices	171
4	The z-transform and its applications in signal processing	172
4.1	Discrete-time signals and systems	173
4.2	The z-transform	174
4.3	The inverse z-transform	179
4.3.1	Power series method	179
4.3.2	Partial fraction expansion method	182
4.3.3	Residue method	188
4.3.4	Comparison of the inverse z-transform methods	194
4.4	Properties of the z-transform	194
4.5	Some applications of the z-transform in signal processing	197

4.5.1	Pole-zero description of discrete-time systems	197
4.5.2	Frequency response estimation	200
4.5.3	Geometric evaluation of frequency response	201
4.5.4	Direct computer evaluation of frequency response	204
4.5.5	Frequency response estimation via FFT	205
4.5.6	Frequency units used in discrete-time systems	205
4.5.7	Stability considerations	208
4.5.8	Difference equations	209
4.5.9	Impulse response estimation	211
4.5.10	Applications in digital filter design	213
4.5.11	Realization structures for digital filters	213
4.6	Summary	218
	Problems	218
	References	218
	Bibliography	223
	Appendices	223
4A	Recursive algorithm for the inverse z-transform	223
4B	C program for evaluating the inverse z-transform and for cascade-to-parallel structure conversion	223
4C	C program for estimating frequency response	225
4D	z-transform operations with MATLAB	231
	References for Appendices	233
		241
5	Correlation and convolution	
5.1	Introduction	242
5.2	Correlation description	242
5.2.1	Cross- and autocorrelation	243
5.2.2	Applications of correlation	249
5.2.3	Fast correlation	257
5.3	Convolution description	267
5.3.1	Properties of convolution	273
5.3.2	Circular convolution	282
5.3.3	System identification	283
5.3.4	Deconvolution	283
5.3.5	Blind deconvolution	285
5.3.6	Fast linear convolution	286
5.3.7	Computational advantages of fast linear convolution	288
5.3.8	Convolution and correlation by sectioning	289
5.3.9	Overlap-add method	290
5.3.10	Overlap-save method	292
5.3.11	Computational advantages of fast convolution by sectioning	297
5.3.12	The relationship between convolution and correlation	300
		301

5.4	Implementation of correlation and convolution	301
5.5	Application examples	302
5.5.1	Correlation	302
5.5.2	Convolution	307
5.6	Summary	310
	Problems	311
	References	315
	Appendix	316
	5A C language program for computing cross- and autocorrelation	316
6	A framework for digital filter design	317
6.1	Introduction to digital filters	318
6.2	Types of digital filters: FIR and IIR filters	319
6.3	Choosing between FIR and IIR filters	321
6.4	Filter design steps	324
6.4.1	Specification of the filter requirements	324
6.4.2	Coefficient calculation	327
6.4.3	Representation of a filter by a suitable structure (realization)	328
6.4.4	Analysis of finite wordlength effects	332
6.4.5	Implementation of a filter	333
6.5	Illustrative examples	334
6.6	Summary	339
	Problems	339
	Reference	341
	Bibliography	341
7	Finite impulse response (FIR) filter design	342
7.1	Introduction	343
7.1.1	Summary of key characteristic features of FIR filters	343
7.1.2	Linear phase response and its implications	344
7.1.3	Types of linear phase FIR filters	347
7.2	FIR filter design	349
7.3	FIR filter specifications	350
7.4	FIR coefficient calculation methods	351
7.5	Window method	352
7.5.1	Some common window functions	354
7.5.2	Summary of the window method of calculating FIR filter coefficients	358
7.5.3	Advantages and disadvantages of the window method	366

7.6	The optimal method	367
7.6.1	Basic concepts	367
7.6.2	Parameters required to use the optimal program	370
7.6.3	Relationships for estimating filter length, N	371
7.6.4	Summary of procedure for calculating filter coefficients by the optimal method	372
7.6.5	Illustrative examples	373
7.7	Frequency sampling method	380
7.7.1	Nonrecursive frequency sampling filters	380
7.7.2	Recursive frequency sampling filters	389
7.7.3	Frequency sampling filters with simple coefficients	390
7.7.4	Summary of the frequency sampling method	398
7.8	Comparison of the window, optimum and frequency sampling methods	398
7.9	Special FIR filter design topics	402
7.9.1	Half-band FIR filters	402
7.9.2	Frequency transformation	404
7.9.3	Computationally efficient FIR filters	406
7.10	Realization structures for FIR filters	407
7.10.1	Transversal structure	407
7.10.2	Linear phase structure	408
7.10.3	Other structures	410
7.10.4	Choosing between structures	410
7.11	Finite wordlength effects in FIR digital filters	411
7.11.1	Coefficient quantization errors	412
7.11.2	Roundoff errors	419
7.11.3	Overflow errors	419
7.12	FIR implementation techniques	420
7.13	Design example	422
7.14	Summary	425
7.15	Application examples of FIR filters	425
	Problems	426
	References	435
	Bibliography	436
	Appendices	437
	7A C programs for FIR filter design	437
	7B FIR filter design with MATLAB	440
8	Design of infinite impulse response (IIR) digital filters	
8.1	Introduction: summary of the basic features of IIR filters	454
8.2	Design stages for digital IIR filters	455

8.3	Performance specification	457
8.4	Coefficient calculation methods for IIR filters	459
8.5	Pole-zero placement method of coefficient calculation	459
8.5.1	Basic concepts and illustrative design examples	459
8.6	Impulse invariant method of coefficient calculation	463
8.6.1	Basic concepts and illustrative design examples	463
8.6.2	Summary of the impulse invariant method	466
8.6.3	Remarks on the impulse invariant method	466
8.7	Matched z-transform (MZT) method of coefficient calculation	468
8.7.1	Basic concepts and illustrative design examples	468
8.7.2	Summary of the matched z-transform method	470
8.7.3	Remarks on the matched z-transform method	471
8.8	Bilinear z-transform (BZT) method of coefficient calculation	471
8.8.1	Basic concepts and illustrative design examples	471
8.8.2	Summary of the BZT method of coefficient calculation	473
8.8.3	Comments on the bilinear transformation method	478
8.9	Use of BZT and classical analog filters to design IIR filters	482
8.9.1	Characteristic features of classical analog filters	483
8.9.2	The BZT methodology using classical analog filters	485
8.9.3	Illustrative design examples (lowpass, highpass, bandpass and bandstop filters)	491
8.10	Calculating IIR filter coefficients by mapping s-plane poles and zeros	500
8.10.1	Basic concepts	500
8.10.2	Illustrative examples	505
8.11	Using IIR filter design programs	508
8.12	Choice of coefficient calculation methods for IIR filters	509
8.12.1	Nyquist effect	510
8.13	Realization structures for IIR digital filters	517
8.13.1	Practical building blocks for IIR filters	518
8.13.2	Cascade and parallel realization structures for higher-order IIR filters	520
8.14	Finite wordlength effects in IIR filters	524
8.14.1	Coefficient quantization errors	526
8.15	Implementation of IIR filters	529
8.16	A detailed design example of an IIR digital filter	530
8.17	Summary	535
8.18	Application examples in digital audio and instrumentation	536
8.18.1	Digital audio	536
8.18.2	Digital control	536
8.18.3	Digital frequency oscillators	536

8.19	Application examples in telecommunication	538
8.19.1	Touch-tone generation and reception for digital telephones	538
8.19.2	Digital telephony: dual tone multifrequency (DTMF) detection using the Goertzel algorithm	540
8.19.3	Clock recovery for data communication	546
	Problems	549
	References	554
	Bibliography	555
	Appendices	557
	8A C programs for IIR digital filter design	557
	8B IIR filter design with MATLAB	562
	8C Evaluation of complex square roots using real arithmetic	577
9	Multirate digital signal processing	579
9.1	Introduction	579
9.1.1	Some current uses of multirate processing in industry	580
9.2	Concepts of multirate signal processing	581
9.2.1	Sampling rate reduction: decimation by integer factors	582
9.2.2	Sampling rate increase: interpolation by integer factors	584
9.2.3	Sampling rate conversion by non-integer factors	586
9.2.4	Multistage approach to sampling rate conversion	589
9.3	Design of practical sampling rate converters	590
9.3.1	Filter specification	590
9.3.2	Filter requirements for individual stages	591
9.3.3	Determining the number of stages and decimation factors	592
9.3.4	Illustrative design examples	594
9.4	Software implementation of sampling rate converters—decimators	601
9.4.1	Program for multistage decimation	602
9.4.2	Test example for the decimation program	604
9.5	Software implementation of interpolators	606
9.5.1	Program for multistage interpolation	610
9.5.2	Test example	610
9.6	Sample rate conversion using polyphase filter structure	612
9.6.1	Polyphase implementation of interpolators	612
9.7	Application examples	617
9.7.1	High quality analog-to-digital conversion for digital audio	618
9.7.2	Efficient digital-to-analog conversion in compact hi-fi systems	618
9.7.3	Application in the acquisition of high quality data	620
9.7.4	Multirate narrowband digital filtering	626
9.7.5	High resolution narrowband spectral analysis	631
9.8	Summary	632
	Problems	633

References	637
Bibliography	638
Appendices	639
9A C programs for multirate processing and systems design	639
9B Multirate digital signal processing with MATLAB	640
10 Adaptive digital filters	645
10.1 When to use adaptive filters and where they have been used	646
10.2 Concepts of adaptive filtering	647
10.2.1 Adaptive filters as a noise canceller	647
10.2.2 Other configurations of the adaptive filter	648
10.2.3 Main components of the adaptive filter	648
10.2.4 Adaptive algorithms	648
10.3 Basic Wiener filter theory	651
10.4 The basic LMS adaptive algorithm	654
10.4.1 Implementation of the basic LMS algorithm	655
10.4.2 Practical limitations of the basic LMS algorithm	658
10.4.3 Other LMS-based algorithms	661
10.5 Recursive least squares algorithm	662
10.5.1 Recursive least squares algorithm	663
10.5.2 Limitations of the recursive least squares algorithm	664
10.5.3 Factorization algorithms	665
10.6 Application example 1 – adaptive filtering of ocular artefacts from the human EEG	666
10.6.1 The physiological problem	666
10.6.2 Artefact processing algorithm	667
10.6.3 Real-time implementation	668
10.7 Application example 2 – adaptive telephone echo cancellation	668
10.8 Other applications	670
10.8.1 Loudspeaking telephones	670
10.8.2 Multipath compensation	670
10.8.3 Adaptive jammer suppression	671
10.8.4 Radar signal processing	672
10.8.5 Separation of speech signals from background noise	672
10.8.6 Fetal monitoring – cancelling of maternal ECG during labour	673
Problems	674
References	674
Bibliography	675
Appendices	676
10A C language programs for adaptive filtering	676
10B MATLAB programs for adaptive filtering	680

11	Spectrum estimation and analysis	681
11.1	Introduction	682
11.2	Principles of spectrum estimation	684
11.3	Traditional methods	687
11.3.1	Pitfalls	687
11.3.2	Windowing	690
11.3.3	The periodogram method and periodogram properties	703
11.3.4	Modified periodogram methods	704
11.3.5	The Blackman–Tukey method	705
11.3.6	The fast correlation method	706
11.3.7	Comparison of the power spectral density estimation methods	706
11.4	Modern parametric estimation methods	707
11.5	Autoregressive spectrum estimation	708
11.5.1	Autoregressive model and filter	708
11.5.2	Power spectrum density of AR series	709
11.5.3	Computation of model parameters – Yule–Walker equations	710
11.5.4	Solution of the Yule–Walker equations	713
11.5.5	Model order	714
11.6	Comparison of estimation methods	715
11.7	Application examples	715
11.7.1	Use of spectral analysis by a DFT for differentiating between brain diseases	715
11.7.2	Spectral analysis of EEGs using autoregressive modelling	719
11.8	Summary	721
11.9	Worked example	721
	Problems	721
	References	722
	Appendix	724
	11A MATLAB programs for spectrum estimation and analysis	725
		725
12	General- and special-purpose digital signal processors	727
12.1	Introduction	727
12.2	Computer architectures for signal processing	728
12.2.1	Harvard architecture	728
12.2.2	Pipelining	730
12.2.3	Hardware multiplier–accumulator	732
12.2.4	Special instructions	737
12.2.5	Replication	738
12.2.6	On-chip memory/cache	741
		742

12.2.7	Extended parallelism – SIMD, VLIW and static superscalar processing	742
12.3	General-purpose digital signal processors	746
12.3.1	Fixed-point digital signal processors	747
12.3.2	Floating-point digital signal processors	756
12.4	Selecting digital signal processors	759
12.5	Implementation of DSP algorithms on general-purpose digital signal processors	761
12.5.1	FIR digital filtering	761
12.5.2	IIR digital filtering	770
12.5.3	FFT processing	777
12.5.4	Multirate processing	782
12.5.5	Adaptive filtering	786
12.6	Special-purpose DSP hardware	787
12.6.1	Hardware digital filters	789
12.6.2	Hardware FFT processors	790
12.7	Summary	792
	Problems	793
	References	796
	Bibliography	797
	Appendix	798
12A	TMS320 assembly language programs for real-time signal processing and a C language program for constant geometry radix-2 FFT	798
13	Analysis of finite wordlength effects in fixed-point DSP systems	805
13.1	Introduction	805
13.2	DSP arithmetic	806
13.2.1	Fixed-point arithmetic	808
13.2.2	Floating-point arithmetic	812
13.3	ADC quantization noise and signal quality	815
13.4	Finite wordlength effects in IIR digital filters	817
13.4.1	Influence of filter structure on finite wordlength effects	818
13.4.2	Coefficient quantization errors in IIR digital filters	822
13.4.3	Coefficient wordlength requirements for stability and desired frequency response	823
13.4.4	Addition overflow errors and their effects	828
13.4.5	Principles of scaling	829
13.4.6	Scaling in cascade realization	832
13.4.7	Scaling in parallel realization	834
13.4.8	Output overflow detection and prevention	835

13.4.9	Product roundoff errors in IIR digital filters	836
13.4.10	Effects of roundoff errors on filter performance	837
13.4.11	Roundoff noise in cascade and parallel realizations	841
13.4.12	Effects of product roundoff noise in modern DSP systems	845
13.4.13	Roundoff noise reduction schemes	846
13.4.14	Determining practical values for error feedback coefficients	853
13.4.15	Limit cycles due to product roundoff errors	857
13.4.16	Other nonlinear phenomena	859
13.5	Finite wordlength effects in FFT algorithms	860
13.5.1	Roundoff errors in FFT	860
13.5.2	Overflow errors and scaling in FFT	862
13.5.3	Coefficient quantization in FFT	864
13.6	Summary	864
	Problems	865
	References	868
	Bibliography	868
	Appendices	870
	13A Finite wordlength analysis program for IIR filters	870
	13B L_2 scaling factor equations	870
14	Applications and design studies	873
14.1	Evaluation boards for real-time signal processing	874
14.1.1	Background	874
14.1.2	TMS320C10 target board	874
14.1.3	DSP56002 evaluation module for real-time DSP	876
14.1.4	TMS320C54 and DSP56300 evaluation boards	876
14.2	DSP applications	877
14.2.1	Detection of fetal heartbeats during labour	877
14.2.2	Adaptive removal of ocular artefacts from human EEGs	885
14.2.3	Equalization of digital audio signals	901
14.3	Design studies	904
14.4	Computer-based multiple choice DSP questions	911
14.5	Summary	920
	Problems	921
	References	921
	Bibliography	923
	Appendix	923
	14A The modified UD factorization algorithm	923
	Index	925