Contents

Preface xv

1 Introduction to Real-Time Digital Signal Processing 1

1.1 Basic Elements of Real-Time DSP Systems 2
1.2 Analog Interface 3
1.2.1 Sampling 3
1.2.2 Quantization and Encoding 7
1.2.3 Smoothing Filters 8
1.2.4 Data Converters 9
1.3 DSP Hardware 10
1.3.1 DSP Hardware Options 10
1.3.2 DSP Processors 13
1.3.3 Fixed- and Floating-Point Processors 15
1.3.4 Real-Time Constraints 16
1.4 DSP System Design 17
1.4.1 Algorithm Development 18
1.4.2 Selection of DSP Processors 19
1.4.3 Software Development 20
1.4.4 High-Level Software Development Tools 21
1.5 Introduction to DSP Development Tools 22
1.5.1 C Compiler 22
1.5.2 Assembler 23
1.5.3 Linker 24
1.5.4 Other Development Tools 25
1.6 Experiments and Program Examples 25
1.6.1 Experiments of Using CCS and DSK 26
1.6.2 Debugging Program Using CCS and DSK 29
1.6.3 File I/O Using Probe Point 32
1.6.4 File I/O Using C File System Functions 35
1.6.5 Code Efficiency Analysis Using Profiler 37
1.6.6 Real-Time Experiments Using DSK 39
1.6.7 Sampling Theory 42
1.6.8 Quantization in ADCs 44
References 45
Exercises 45

2 Introduction to TMS320C55x Digital Signal Processor 49

2.1 Introduction 49
2.2 TMS320C55x Architecture 50
2.2.1 Architecture Overview 50
2.2.2 Buses 53
2.2.3 On-Chip Memories 53
2.2.4 Memory-Mapped Registers 55
2.2.5 Interrupts and Interrupt Vector 55
2.3 TMS320C55x Peripherals 58
2.3.1 External Memory Interface 60
2.3.2 Direct Memory Access 60
2.3.3 Enhanced Host-Port Interface 61
2.3.4 Multi-Channel Buffered Serial Ports 62
2.3.5 Clock Generator and Timers 65
2.3.6 General Purpose Input/Output Port 65
2.4 TMS320C55x Addressing Modes 65
2.4.1 Direct Addressing Modes 66
2.4.2 Indirect Addressing Modes 68
2.4.3 Absolute Addressing Modes 70
2.4.4 Memory-Mapped Register Addressing Mode 70
2.4.5 Register Bits Addressing Mode 71
2.4.6 Circular Addressing Mode 72
2.5 Pipeline and Parallelism 73
2.5.1 TMS320C55x Pipeline 73
2.5.2 Parallel Execution 74
2.6 TMS320C55x Instruction Set 76
2.6.1 Arithmetic Instructions 76
2.6.2 Logic and Bit Manipulation Instructions 77
2.6.3 Move Instruction 78
2.6.4 Program Flow Control Instructions 78
2.7 TMS320C55x Assembly Language Programming 82
2.7.1 Assembly Directives 82
2.7.2 Assembly Statement Syntax 84
2.8 C Language Programming for TMS320C55x 86
2.8.1 Data Types 86
2.8.2 Assembly Code Generation by C Compiler 87
2.8.3 Compiler Keywords and Pragma Directives 89
2.9 Mixed C-and-Assembly Language Programming 90
2.10 Experiments and Program Examples 93
2.10.1 Interfacing C with Assembly Code 93
2.10.2 Addressing Modes Using Assembly Programming 94
2.10.3 Phase-Locked Loop and Timers 97
2.10.4 EMIF Configuration for Using SDRAM 103
2.10.5 Programming Flash Memory Devices 105
2.10.6 Using McBSP 106
2.10.7 AIC23 Configurations 109
2.10.8 Direct Memory Access 111
References 115
Exercises 115

3 DSP Fundamentals and Implementation Considerations **121**

3.1 Digital Signals and Systems 121
3.1.1 Elementary Digital Signals 121
3.1.2 Block Diagram Representation of Digital Systems 123
3.2 System Concepts 126
3.2.1 Linear Time-Invariant Systems 126
3.2.2 The z-Transform 130
3.2.3 Transfer Functions 132
3.2.4 Poles and Zeros 135
3.2.5 Frequency Responses 138
3.2.6 Discrete Fourier Transform 141
3.3 Introduction to Random Variables 142
3.3.1 Review of Random Variables 142
3.3.2 Operations of Random Variables 144
3.4 Fixed-Point Representations and Quantization Effects 147
3.4.1 Fixed-Point Formats 147
3.4.2 Quantization Errors 151
3.4.3 Signal Quantization 151
3.4.4 Coefficient Quantization 153
3.4.5 Roundoff Noise 153
3.4.6 Fixed-Point Toolbox 154
3.5 Overflow and Solutions 157
3.5.1 Saturation Arithmetic 157
3.5.2 Overflow Handling 158
3.5.3 Scaling of Signals 158
3.5.4 Guard Bits 159
3.6 Experiments and Program Examples 159
3.6.1 Quantization of Sinusoidal Signals 160
3.6.2 Quantization of Audio Signals 161
3.6.3 Quantization of Coefficients 162
3.6.4 Overflow and Saturation Arithmetic 164
3.6.5 Function Approximations 167
3.6.6 Real-Time Digital Signal Generation Using DSK 175
References 180
Exercises 180

4 Design and Implementation of FIR Filters **185**

4.1 Introduction to FIR Filters 185
4.1.1 Filter Characteristics 185
4.1.2 Filter Types 187
4.1.3 Filter Specifications 189
4.1.4 Linear-Phase FIR Filters 191
4.1.5 Realization of FIR Filters 194
4.2 Design of FIR Filters 196
4.2.1 Fourier Series Method 197
4.2.2 Gibbs Phenomenon 198
4.2.3 Window Functions 201

4.2.4 Design of FIR Filters Using MATLAB 206
4.2.5 Design of FIR Filters Using FDATool 207
4.3 Implementation Considerations 213
4.3.1 Quantization Effects in FIR Filters 213
4.3.2 MATLAB Implementations 216
4.3.3 Floating-Point C Implementations 218
4.3.4 Fixed-Point C Implementations 219
4.4 Applications: Interpolation and Decimation Filters 220
4.4.1 Interpolation 220
4.4.2 Decimation 221
4.4.3 Sampling-Rate Conversion 221
4.4.4 MATLAB Implementations 224
4.5 Experiments and Program Examples 225
4.5.1 Implementation of FIR Filters Using Fixed-Point C 226
4.5.2 Implementation of FIR Filter Using C55x Assembly Language 226
4.5.3 Optimization for Symmetric FIR Filters 228
4.5.4 Optimization Using Dual MAC Architecture 230
4.5.5 Implementation of Decimation 232
4.5.6 Implementation of Interpolation 233
4.5.7 Sample Rate Conversion 234
4.5.8 Real-Time Sample Rate Conversion Using DSP/BIOS and DSK 235
References 245
Exercises 245

5 Design and Implementation of IIR Filters 249
5.1 Introduction 249
5.1.1 Analog Systems 249
5.1.2 Mapping Properties 251
5.1.3 Characteristics of Analog Filters 252
5.1.4 Frequency Transforms 254
5.2 Design of IIR Filters 255
5.2.1 Bilinear Transform 256
5.2.2 Filter Design Using Bilinear Transform 257
5.3 Realization of IIR Filters 258
5.3.1 Direct Forms 258
5.3.2 Cascade Forms 260
5.3.3 Parallel Forms 262
5.3.4 Realization of IIR Filters Using MATLAB 263
5.4 Design of IIR Filters Using MATLAB 264
5.4.1 Filter Design Using MATLAB 264
5.4.2 Frequency Transforms Using MATLAB 267
5.4.3 Design and Realization Using FDATool 268
5.5 Implementation Considerations 271
5.5.1 Stability 271
5.5.2 Finite-Precision Effects and Solutions 273
5.5.3 MATLAB Implementations 275

5.6 Practical Applications 279
5.6.1 Recursive Resonators 279
5.6.2 Recursive Quadrature Oscillators 282
5.6.3 Parametric Equalizers 284
5.7 Experiments and Program Examples 285
5.7.1 Floating-Point Direct-Form I IIR Filter 285
5.7.2 Fixed-Point Direct-Form I IIR Filter 286
5.7.3 Fixed-Point Direct-Form II Cascade IIR Filter 287
5.7.4 Implementation Using DSP Intrinsics 289
5.7.5 Implementation Using Assembly Language 290
5.7.6 Real-Time Experiments Using DSP/BIOS 293
5.7.7 Implementation of Parametric Equalizer 296
5.7.8 Real-Time Two-Band Equalizer Using DSP/BIOS 297
References 299
Exercises 299

6 Frequency Analysis and Fast Fourier Transform 303
6.1 Fourier Series and Transform 303
6.1.1 Fourier Series 303
6.1.2 Fourier Transform 304
6.2 Discrete Fourier Transform 305
6.2.1 Discrete-Time Fourier Transform 305
6.2.2 Discrete Fourier Transform 307
6.2.3 Important Properties 310
6.3 Fast Fourier Transforms 313
6.3.1 Decimation-in-Time 314
6.3.2 Decimation-in-Frequency 316
6.3.3 Inverse Fast Fourier Transform 317
6.4 Implementation Considerations 317
6.4.1 Computational Issues 317
6.4.2 Finite-Precision Effects 318
6.4.3 MATLAB Implementations 318
6.4.4 Fixed-Point Implementation Using MATLAB 320
6.5 Practical Applications 322
6.5.1 Spectral Analysis 322
6.5.2 Spectral Leakage and Resolution 323
6.5.3 Power Spectrum Density 325
6.5.4 Fast Convolution 328
6.6 Experiments and Program Examples 332
6.6.1 Floating-Point C Implementation of DFT 332
6.6.2 C55x Assembly Implementation of DFT 332
6.6.3 Floating-Point C Implementation of FFT 336
6.6.4 C55x Intrinsics Implementation of FFT 338
6.6.5 Assembly Implementation of FFT and Inverse FFT 339
6.6.6 Implementation of Fast Convolution 343
6.6.7 Real-Time FFT Using DSP/BIOS 345
6.6.8 Real-Time Fast Convolution 347
References 347
Exercises 348

7 Adaptive Filtering **351**
7.1 Introduction to Random Processes 351
7.2 Adaptive Filters 354
7.2.1 Introduction to Adaptive Filtering 354
7.2.2 Performance Function 355
7.2.3 Method of Steepest Descent 358
7.2.4 The LMS Algorithm 360
7.2.5 Modified LMS Algorithms 361
7.3 Performance Analysis 362
7.3.1 Stability Constraint 362
7.3.2 Convergence Speed 363
7.3.3 Excess Mean-Square Error 363
7.3.4 Normalized LMS Algorithm 364
7.4 Implementation Considerations 364
7.4.1 Computational Issues 365
7.4.2 Finite-Precision Effects 365
7.4.3 MATLAB Implementations 366
7.5 Practical Applications 368
7.5.1 Adaptive System Identification 368
7.5.2 Adaptive Linear Prediction 369
7.5.3 Adaptive Noise Cancelation 372
7.5.4 Adaptive Notch Filters 374
7.5.5 Adaptive Channel Equalization 375
7.6 Experiments and Program Examples 377
7.6.1 Floating-Point C Implementation 377
7.6.2 Fixed-Point C Implementation of Leaky LMS Algorithm 379
7.6.3 ETSI Implementation of NLMS Algorithm 380
7.6.4 Assembly Language Implementation of Delayed LMS Algorithm 383
7.6.5 Adaptive System Identification 387
7.6.6 Adaptive Prediction and Noise Cancelation 388
7.6.7 Adaptive Channel Equalizer 392
7.6.8 Real-Time Adaptive Line Enhancer Using DSK 394
References 396
Exercises 397

8 Digital Signal Generators **401**
8.1 Sinewave Generators 401
8.1.1 Lookup-Table Method 401
8.1.2 Linear Chirp Signal 404
8.2 Noise Generators 405
8.2.1 Linear Congruential Sequence Generator 405
8.2.2 Pseudo-Random Binary Sequence Generator 407
8.3 Practical Applications 409
8.3.1 Siren Generators 409
8.3.2 White Gaussian Noise 409
8.3.3 Dual-Tone Multifrequency Tone Generator 410
8.3.4 Comfort Noise in Voice Communication Systems 411
8.4 Experiments and Program Examples 412
8.4.1 Sinewave Generator Using C5510 DSK 412
8.4.2 White Noise Generator Using C5510 DSK 413

8.4.3 Wail Siren Generator Using C5510 DSK 414
8.4.4 DTMF Generator Using C5510 DSK 415
8.4.5 DTMF Generator Using MATLAB Graphical User Interface 416
References 418
Exercises 418

9 Dual-Tone Multifrequency Detection 421
9.1 Introduction 421
9.2 DTMF Tone Detection 422
9.2.1 DTMF Decode Specifications 422
9.2.2 Goertzel Algorithm 423
9.2.3 Other DTMF Detection Methods 426
9.2.4 Implementation Considerations 428
9.3 Internet Application Issues and Solutions 431
9.4 Experiments and Program Examples 432
9.4.1 Implementation of Goertzel Algorithm Using Fixed-Point C 432
9.4.2 Implementation of Goertzel Algorithm Using C55x Assembly Language 434
9.4.3 DTMF Detection Using C5510 DSK 435
9.4.4 DTMF Detection Using All-Pole Modeling 439
References 441
Exercises 442

10 Adaptive Echo Cancelation 443
10.1 Introduction to Line Echoes 443
10.2 Adaptive Echo Canceler 444
10.2.1 Principles of Adaptive Echo Cancelation 445
10.2.2 Performance Evaluation 446
10.3 Practical Considerations 447
10.3.1 Prewhitening of Signals 447
10.3.2 Delay Detection 448
10.4 Double-Talk Effects and Solutions 450
10.5 Nonlinear Processor 453
10.5.1 Center Clipper 453
10.5.2 Comfort Noise 453
10.6 Acoustic Echo Cancelation 454
10.6.1 Acoustic Echoes 454
10.6.2 Acoustic Echo Canceler 456
10.6.3 Subband Implementations 457
10.6.4 Delay-Free Structures 459
10.6.5 Implementation Considerations 459
10.6.6 Testing Standards 460
10.7 Experiments and Program Examples 461
10.7.1 MATLAB Implementation of AEC 461
10.7.2 Acoustic Echo Cancelation Using Floating-Point C 464
10.7.3 Acoustic Echo Canceler Using C55x Intrinsics 468
10.7.4 Experiment of Delay Estimation 469
References 472
Exercises 472

11 Speech-Coding Techniques **475**
11.1 Introduction to Speech-Coding 475
11.2 Overview of CELP Vocoders 476
11.2.1 Synthesis Filter 477
11.2.2 Long-Term Prediction Filter 481
11.2.3 Perceptual Based Minimization Procedure 481
11.2.4 Excitation Signal 482
11.2.5 Algebraic CELP 483
11.3 Overview of Some Popular CODECs 484
11.3.1 Overview of G.723.1 484
11.3.2 Overview of G.729 488
11.3.3 Overview of GSM AMR 490
11.4 Voice over Internet Protocol Applications 492
11.4.1 Overview of VoIP 492
11.4.2 Real-Time Transport Protocol and Payload Type 493
11.4.3 Example of Packing G.729 496
11.4.4 RTP Data Analysis Using Ethereal Trace 496
11.4.5 Factors Affecting the Overall Voice Quality 497
11.5 Experiments and Program Examples 497
11.5.1 Calculating LPC Coefficients Using Floating-Point C 497
11.5.2 Calculating LPC Coefficients Using C55x Intrinsics 499
11.5.3 MATLAB Implementation of Formant Perceptual Weighting Filter 504
11.5.4 Implementation of Perceptual Weighting Filter Using C55x Intrinsics 506
References 507
Exercises 508

12 Speech Enhancement Techniques **509**
12.1 Introduction to Noise Reduction Techniques 509
12.2 Spectral Subtraction Techniques 510
12.2.1 Short-Time Spectrum Estimation 511
12.2.2 Magnitude Subtraction 511
12.3 Voice Activity Detection 513
12.4 Implementation Considerations 515
12.4.1 Spectral Averaging 515
12.4.2 Half-Wave Rectification 515
12.4.3 Residual Noise Reduction 516
12.5 Combination of Acoustic Echo Cancelation with NR 516
12.6 Voice Enhancement and Automatic Level Control 518
12.6.1 Voice Enhancement Devices 518
12.6.2 Automatic Level Control 519
12.7 Experiments and Program Examples 519
12.7.1 Voice Activity Detection 519
12.7.2 MATLAB Implementation of NR Algorithm 522
12.7.3 Floating-Point C Implementation of NR 522
12.7.4 Mixed C55x Assembly and Intrinsics Implementations of VAD 522
12.7.5 Combining AEC with NR 526
References 529
Exercises 529

13 Audio Signal Processing **531**
13.1 Introduction 531
13.2 Basic Principles of Audio Coding 531
13.2.1 Auditory-Masking Effects for Perceptual Coding 533
13.2.2 Frequency-Domain Coding 536
13.2.3 Lossless Audio Coding 538
13.3 Multichannel Audio Coding 539
13.3.1 MP3 540
13.3.2 Dolby AC-3 541
13.3.3 MPEG-2 AAC 542
13.4 Connectivity Processing 544
13.5 Experiments and Program Examples 544
13.5.1 Floating-Point Implementation of MDCT 544
13.5.2 Implementation of MDCT Using C55x Intrinsics 547
13.5.3 Experiments of Preecho Effects 549
13.5.4 Floating-Point C Implementation of MP3 Decoding 549
References 553
Exercises 553

14 Channel Coding Techniques **555**
14.1 Introduction 555
14.2 Block Codes 556
14.2.1 Reed–Solomon Codes 558
14.2.2 Applications of Reed–Solomon Codes 562
14.2.3 Cyclic Redundant Codes 563
14.3 Convolutional Codes 564
14.3.1 Convolutional Encoding 564
14.3.2 Viterbi Decoding 564
14.3.3 Applications of Viterbi Decoding 566
14.4 Experiments and Program Examples 569
14.4.1 Reed–Solomon Coding Using MATALB 569
14.4.2 Reed–Solomon Coding Using Simulink 570
14.4.3 Verification of RS(255, 239) Generation Polynomial 571
14.4.4 Convolutional Codes 572
14.4.5 Implementation of Convolutional Codes Using C 573
14.4.6 Implementation of CRC-32 575
References 576
Exercises 577

15 Introduction to Digital Image Processing **579**
15.1 Digital Images and Systems 579
15.1.1 Digital Images 579
15.1.2 Digital Image Systems 580
15.2 RGB Color Spaces and Color Filter Array Interpolation 581
15.3 Color Spaces 584
15.3.1 YC_bC_r and YUV Color Spaces 584
15.3.2 CYMK Color Space 585

15.3.3 YIQ Color Space 585
15.3.4 HSV Color Space 585
15.4 YC_bC_r Subsampled Color Spaces 586
15.5 Color Balance and Correction 586
15.5.1 Color Balance 587
15.5.2 Color Adjustment 588
15.5.3 Gamma Correction 589
15.6 Image Histogram 590
15.7 Image Filtering 591
15.8 Image Filtering Using Fast Convolution 596
15.9 Practical Applications 597
15.9.1 JPEG Standard 597
15.9.2 2-D Discrete Cosine Transform 599
15.10 Experiments and Program Examples 601
15.10.1 YC_bC_r to RGB Conversion 601
15.10.2 Using CCS Link with DSK and Simulator 604
15.10.3 White Balance 607
15.10.4 Gamma Correction and Contrast Adjustment 610
15.10.5 Histogram and Histogram Equalization 611
15.10.6 2-D Image Filtering 613
15.10.7 Implementation of DCT and IDCT 617
15.10.8 TMS320C55x Image Accelerator for DCT and IDCT 621
15.10.9 TMS320C55x Hardware Accelerator Image/Video Processing Library 623
References 625
Exercises 625

Appendix A Some Useful Formulas and Definitions 627
A.1 Trigonometric Identities 627
A.2 Geometric Series 628
A.3 Complex Variables 628
A.4 Units of Power 630
References 631

Appendix B Software Organization and List of Experiments 633

Index 639