Contents

Preface xv

1 Introduction to Real-Time Digital Signal Processing 1
  1.1 Basic Elements of Real-Time DSP Systems 2
  1.2 Input and Output Channels 3
    1.2.1 Input Signal Conditioning 3
    1.2.2 A/D Conversion 4
    1.2.3 Sampling 5
    1.2.4 Quantizing and Encoding 7
    1.2.5 D/A Conversion 9
    1.2.6 Input/Output Devices 9
  1.3 DSP Hardware 11
    1.3.1 DSP Hardware Options 11
    1.3.2 Fixed- and Floating-Point Devices 13
    1.3.3 Real-Time Constraints 14
  1.4 DSP System Design 14
    1.4.1 Algorithm Development 14
    1.4.2 Selection of DSP Chips 16
    1.4.3 Software Development 17
    1.4.4 High-Level Software Development Tools 18
  1.5 Experiments Using Code Composer Studio 19
    1.5.1 Experiment 1A – Using the CCS and the TMS320C55x Simulator 20
    1.5.2 Experiment 1B – Debugging Program on the CCS 25
    1.5.3 Experiment 1C – File Input and Output 28
    1.5.4 Experiment 1D – Code Efficiency Analysis 29
    1.5.5 Experiment 1E – General Extension Language 32
References 33
Exercises 33

2 Introduction to TMS320C55x Digital Signal Processor 35
  2.1 Introduction 35
  2.2 TMS320C55x Architecture 36
    2.2.1 TMS320C55x Architecture Overview 36
    2.2.2 TMS320C55x Buses 39
    2.2.3 TMS320C55x Memory Map 40
CONTENTS

2.3 Software Development Tools 40
  2.3.1 C Compiler 42
  2.3.2 Assembler 44
  2.3.3 Linker 46
  2.3.4 Code Composer Studio 48
  2.3.5 Assembly Statement Syntax 49

2.4 TMS320C55x Addressing Modes 50
  2.4.1 Direct Addressing Mode 52
  2.4.2 Indirect Addressing Mode 53
  2.4.3 Absolute Addressing Mode 56
  2.4.4 Memory-Mapped Register Addressing Mode 56
  2.4.5 Register Bits Addressing Mode 57
  2.4.6 Circular Addressing Mode 58

2.5 Pipeline and Parallelism 59
  2.5.1 TMS320C55x Pipeline 59
  2.5.2 Parallel Execution 60

2.6 TMS320C55x Instruction Set 63
  2.6.1 Arithmetic Instructions 63
  2.6.2 Logic and Bits Manipulation Instructions 64
  2.6.3 Move Instruction 65
  2.6.4 Program Flow Control Instructions 66

2.7 Mixed C and Assembly Language Programming 68

2.8 Experiments – Assembly Programming Basics 70
  2.8.1 Experiment 2A – Interfacing C with Assembly Code 71
  2.8.2 Experiment 2B – Addressing Mode Experiments 72

References 75
Exercises 75

3 DSP Fundamentals and Implementation
Considerations 77

3.1 Digital Signals and Systems 77
  3.1.1 Elementary Digital Signals 77
  3.1.2 Block Diagram Representation of Digital Systems 79
  3.1.3 Impulse Response of Digital Systems 83

3.2 Introduction to Digital Filters 83
  3.2.1 FIR Filters and Power Estimators 84
  3.2.2 Response of Linear Systems 87
  3.2.3 IIR Filters 88

3.3 Introduction to Random Variables 90
  3.3.1 Review of Probability and Random Variables 90
  3.3.2 Operations on Random Variables 92

3.4 Fixed-Point Representation and Arithmetic 95

3.5 Quantization Errors 98
  3.5.1 Input Quantization Noise 98
  3.5.2 Coefficient Quantization Noise 101
  3.5.3 Roundoff Noise 102

3.6 Overflow and Solutions 103
  3.6.1 Saturation Arithmetic 103
  3.6.2 Overflow Handling 104
  3.6.3 Scaling of Signals 105

3.7 Implementation Procedure for Real-Time Applications 107
### CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.8 Experiments of Fixed-Point Implementations</td>
<td>108</td>
</tr>
<tr>
<td>3.8.1 Experiment 3A – Quantization of Sinusoidal Signals</td>
<td>109</td>
</tr>
<tr>
<td>3.8.2 Experiment 3B – Quantization of Speech Signals</td>
<td>111</td>
</tr>
<tr>
<td>3.8.3 Experiment 3C – Overflow and Saturation Arithmetic</td>
<td>112</td>
</tr>
<tr>
<td>3.8.4 Experiment 3D – Quantization of Coefficients</td>
<td>115</td>
</tr>
<tr>
<td>3.8.5 Experiment 3E – Synthesizing Sine Function</td>
<td>117</td>
</tr>
<tr>
<td>References</td>
<td>121</td>
</tr>
<tr>
<td>Exercises</td>
<td>122</td>
</tr>
</tbody>
</table>

### 4 Frequency Analysis

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Fourier Series and Transform</td>
<td>127</td>
</tr>
<tr>
<td>4.1.1 Fourier Series</td>
<td>127</td>
</tr>
<tr>
<td>4.1.2 Fourier Transform</td>
<td>130</td>
</tr>
<tr>
<td>4.2 The z-Transforms</td>
<td>133</td>
</tr>
<tr>
<td>4.2.1 Definitions and Basic Properties</td>
<td>133</td>
</tr>
<tr>
<td>4.2.2 Inverse z-Transform</td>
<td>136</td>
</tr>
<tr>
<td>4.3 System Concepts</td>
<td>141</td>
</tr>
<tr>
<td>4.3.1 Transfer Functions</td>
<td>141</td>
</tr>
<tr>
<td>4.3.2 Digital Filters</td>
<td>143</td>
</tr>
<tr>
<td>4.3.3 Poles and Zeros</td>
<td>144</td>
</tr>
<tr>
<td>4.3.4 Frequency Responses</td>
<td>148</td>
</tr>
<tr>
<td>4.4 Discrete Fourier Transform</td>
<td>152</td>
</tr>
<tr>
<td>4.4.1 Discrete-Time Fourier Series and Transform</td>
<td>152</td>
</tr>
<tr>
<td>4.4.2 Aliasing and Folding</td>
<td>154</td>
</tr>
<tr>
<td>4.4.3 Discrete Fourier Transform</td>
<td>157</td>
</tr>
<tr>
<td>4.4.4 Fast Fourier Transform</td>
<td>159</td>
</tr>
<tr>
<td>4.5 Applications</td>
<td>160</td>
</tr>
<tr>
<td>4.5.1 Design of Simple Notch Filters</td>
<td>160</td>
</tr>
<tr>
<td>4.5.2 Analysis of Room Acoustics</td>
<td>162</td>
</tr>
<tr>
<td>4.6 Experiments Using the TMS320C55x</td>
<td>165</td>
</tr>
<tr>
<td>4.6.1 Experiment 4A – Twiddle Factor Generation</td>
<td>167</td>
</tr>
<tr>
<td>4.6.2 Experiment 4B – Complex Data Operation</td>
<td>169</td>
</tr>
<tr>
<td>4.6.3 Experiment 4C – Implementation of DFT</td>
<td>171</td>
</tr>
<tr>
<td>4.6.4 Experiment 4D – Experiment Using Assembly Routines</td>
<td>173</td>
</tr>
<tr>
<td>References</td>
<td>176</td>
</tr>
<tr>
<td>Exercises</td>
<td>176</td>
</tr>
</tbody>
</table>

### 5 Design and Implementation of FIR Filters

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Introduction to Digital Filters</td>
<td>181</td>
</tr>
<tr>
<td>5.1.1 Filter Characteristics</td>
<td>182</td>
</tr>
<tr>
<td>5.1.2 Filter Types</td>
<td>183</td>
</tr>
<tr>
<td>5.1.3 Filter Specifications</td>
<td>185</td>
</tr>
<tr>
<td>5.2 FIR Filtering</td>
<td>189</td>
</tr>
<tr>
<td>5.2.1 Linear Convolution</td>
<td>189</td>
</tr>
<tr>
<td>5.2.2 Some Simple FIR Filters</td>
<td>192</td>
</tr>
<tr>
<td>5.2.3 Linear Phase FIR Filters</td>
<td>194</td>
</tr>
<tr>
<td>5.2.4 Realization of FIR Filters</td>
<td>198</td>
</tr>
<tr>
<td>5.3 Design of FIR Filters</td>
<td>201</td>
</tr>
<tr>
<td>5.3.1 Filter Design Procedure</td>
<td>201</td>
</tr>
<tr>
<td>5.3.2 Fourier Series Method</td>
<td>202</td>
</tr>
<tr>
<td>5.3.3 Gibbs Phenomenon</td>
<td>205</td>
</tr>
<tr>
<td>6</td>
<td>Design and Implementation of IIR Filters</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>6.1</td>
<td>Laplace Transform</td>
</tr>
<tr>
<td>6.1.1</td>
<td>Introduction to the Laplace Transform</td>
</tr>
<tr>
<td>6.1.2</td>
<td>Relationships between the Laplace and z-Transforms</td>
</tr>
<tr>
<td>6.1.3</td>
<td>Mapping Properties</td>
</tr>
<tr>
<td>6.2</td>
<td>Analog Filters</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Introduction to Analog Filters</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Characteristics of Analog Filters</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Frequency Transforms</td>
</tr>
<tr>
<td>6.3</td>
<td>Design of IIR Filters</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Review of IIR Filters</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Impulse-Invariant Method</td>
</tr>
<tr>
<td>6.3.3</td>
<td>Bilinear Transform</td>
</tr>
<tr>
<td>6.3.4</td>
<td>Filter Design Using Bilinear Transform</td>
</tr>
<tr>
<td>6.4</td>
<td>Realization of IIR Filters</td>
</tr>
<tr>
<td>6.4.1</td>
<td>Direct Forms</td>
</tr>
<tr>
<td>6.4.2</td>
<td>Cascade Form</td>
</tr>
<tr>
<td>6.4.3</td>
<td>Parallel Form</td>
</tr>
<tr>
<td>6.4.4</td>
<td>Realization Using MATLAB</td>
</tr>
<tr>
<td>6.5</td>
<td>Design of IIR Filters Using MATLAB</td>
</tr>
<tr>
<td>6.6</td>
<td>Implementation Considerations</td>
</tr>
<tr>
<td>6.6.1</td>
<td>Stability</td>
</tr>
<tr>
<td>6.6.2</td>
<td>Finite-Precision Effects and Solutions</td>
</tr>
<tr>
<td>6.6.3</td>
<td>Software Implementations</td>
</tr>
<tr>
<td>6.6.4</td>
<td>Practical Applications</td>
</tr>
<tr>
<td>6.7</td>
<td>Software Developments and Experiments Using the TMS320C55x</td>
</tr>
<tr>
<td>6.7.1</td>
<td>Design of IIR Filter</td>
</tr>
<tr>
<td>6.7.2</td>
<td>Experiment 6A – Floating-Point C Implementation</td>
</tr>
<tr>
<td>6.7.3</td>
<td>Experiment 6B – Fixed-Point C Implementation Using Intrinsics</td>
</tr>
<tr>
<td>6.7.4</td>
<td>Experiment 6C – Fixed-Point C Programming Considerations</td>
</tr>
<tr>
<td>6.7.5</td>
<td>Experiment 6D – Assembly Language Implementations</td>
</tr>
</tbody>
</table>

References

Exercises

7 Fast Fourier Transform and Its Applications

| 7.1 | Discrete Fourier Transform | 303 |
| 7.1.1 | Definitions | 304 |
| 7.1.2 | Important Properties of DFT | 308 |
| 7.1.3 | Circular Convolution | 311 |
CONTENTS

7.2 Fast Fourier Transforms 314
7.2.1 Decimation-in-Time 315
7.2.2 Decimation-in-Frequency 319
7.2.3 Inverse Fast Fourier Transform 320
7.2.4 MATLAB Implementations 321

7.3 Applications 322
7.3.1 Spectrum Estimation and Analysis 322
7.3.2 Spectral Leakage and Resolution 324
7.3.3 Power Density Spectrum 328
7.3.4 Fast Convolution 330
7.3.5 Spectrogram 332

7.4 Implementation Considerations 333
7.4.1 Computational Issues 334
7.4.2 Finite-Precision Effects 334

7.5 Experiments Using the TMS320C55x 336
7.5.1 Experiment 7A – Radix-2 Complex FFT 336
7.5.2 Experiment 7B – Radix-2 Complex FFT Using Assembly Language 341
7.5.3 Experiment 7C – FFT and IFFT 344
7.5.4 Experiment 7D – Fast Convolution 344

References 346
Exercises 347

8 Adaptive Filtering 351
8.1 Introduction to Random Processes 351
8.1.1 Correlation Functions 352
8.1.2 Frequency-Domain Representations 356

8.2 Adaptive Filters 359
8.2.1 Introduction to Adaptive Filtering 359
8.2.2 Performance Function 361
8.2.3 Method of Steepest Descent 365
8.2.4 The LMS Algorithm 366

8.3 Performance Analysis 367
8.3.1 Stability Constraint 367
8.3.2 Convergence Speed 368
8.3.3 Excess Mean-Square Error 369

8.4 Modified LMS Algorithms 370
8.4.1 Normalized LMS Algorithm 370
8.4.2 Leaky LMS Algorithm 371

8.5 Applications 372
8.5.1 Adaptive System Identification 372
8.5.2 Adaptive Linear Prediction 373
8.5.3 Adaptive Noise Cancellation 375
8.5.4 Adaptive Notch Filters 377
8.5.5 Adaptive Channel Equalization 379

8.6 Implementation Considerations 381
8.6.1 Computational Issues 381
8.6.2 Finite-Precision Effects 382

8.7 Experiments Using the TMS320C55x 385
8.7.1 Experiment 8A – Adaptive System Identification 385
8.7.2 Experiment 8B – Adaptive Predictor Using the Leaky LMS Algorithm 390

References 396
Exercises 396
9 Practical DSP Applications in Communications 399

9.1 Sinewave Generators and Applications 400
  9.1.1 Lookup-Table Method 400
  9.1.2 Linear Chirp Signal 402
  9.1.3 DTMF Tone Generator 403

9.2 Noise Generators and Applications 404
  9.2.1 Linear Congruential Sequence Generator 404
  9.2.2 Pseudo-Random Binary Sequence Generator 406
  9.2.3 Comfort Noise in Communication Systems 408
  9.2.4 Off-Line System Modeling 409

9.3 DTMF Tone Detection 410
  9.3.1 Specifications 410
  9.3.2 Goertzel Algorithm 411
  9.3.3 Implementation Considerations 414

9.4 Adaptive Echo Cancellation 417
  9.4.1 Line Echoes 417
  9.4.2 Adaptive Echo Canceler 418
  9.4.3 Practical Considerations 422
  9.4.4 Double-Talk Effects and Solutions 423
  9.4.5 Residual Echo Suppressor 425

9.5 Acoustic Echo Cancellation 426
  9.5.1 Introduction 426
  9.5.2 Acoustic Echo Canceler 427
  9.5.3 Implementation Considerations 428

9.6 Speech Enhancement Techniques 429
  9.6.1 Noise Reduction Techniques 429
  9.6.2 Spectral Subtraction Techniques 431
  9.6.3 Implementation Considerations 433

9.7 Projects Using the TMS320C55x 435
  9.7.1 Project Suggestions 435
  9.7.2 A Project Example – Wireless Application 437

References 442

Appendix A Some Useful Formulas 445

A.1 Trigonometric Identities 445
A.2 Geometric Series 446
A.3 Complex Variables 447
A.4 Impulse Functions 449
A.5 Vector Concepts 449
A.6 Units of Power 450
Reference 451

Appendix B Introduction of MATLAB for DSP Applications 453

B.1 Elementary Operations 453
  B.1.1 Initializing Variables and Vectors 453
  B.1.2 Graphics 455
  B.1.3 Basic Operators 457
  B.1.4 Files 459
B.2 Generation and Processing of Digital Signals 460
B.3 DSP Applications 463
B.4 User-Written Functions 465
Appendix C  Introduction of C Programming for DSP Applications  469
C.1  A Simple C Program  470
   C.1.1  Variables and Assignment Operators  472
   C.1.2  Numeric Data Types and Conversion  473
   C.1.3  Arrays  474
C.2  Arithmetic and Bitwise Operators  475
   C.2.1  Arithmetic Operators  475
   C.2.2  Bitwise Operators  476
C.3  An FIR Filter Program  476
   C.3.1  Command-Line Arguments  477
   C.3.2  Pointers  477
   C.3.3  C Functions  478
   C.3.4  Files and I/O Operations  480
C.4  Control Structures and Loops  481
   C.4.1  Control Structures  481
   C.4.2  Logical Operators  483
   C.4.3  Loops  484
C.5  Data Types Used by the TMS320C55x  485
   References  486

Appendix D  About the Software  487
Index  489