# Contents

**Preface** xv  
**Preface to the First Edition** xvii  
**List of Examples** xxi  
**Programs/Files on Accompanying CD** xxvii  

### 1  DSP Development System 1  
1.1 Introduction 1  
1.2 DSK Support Tools 3  

1.2.1 C6713 and C6416 DSK Boards 4  
1.2.2 TMS320C6713 Digital Signal Processor 4  
1.2.3 TMS320C6416 Digital Signal Processor 5  

1.3 Code Composer Studio 6  

1.3.1 CCS Version 3.1 Installation and Support 6  
1.3.2 Installation of Files Supplied with This Book 7  
1.3.3 File Types 7  

1.4 Quick Tests of the DSK (On Power On and Using CCS) 7  

1.5 Programming Examples to Test the DSK Tools 9  

1.6 Support Files 30  

1.6.1 Initialization/Communication File (c6713dskinit.c) 30  
1.6.2 Header File (c6713dskinit.h) 33  
1.6.3 Vector Files (vectors_intr.asm, vectors_poll.asm) 34  
1.6.4 Linker Command File (c6713dsk.cmd) 34  

1.7 Assignments 38  

References 41  

vii
# Contents

## 2 Input and Output with the DSK

2.1 Introduction 45  
2.2 TLV320AIC23 (AIC23) Onboard Stereo Codec for Input and Output 46  
2.3 Programming Examples Using C Code 48  
2.3.1 Real-Time Sine Wave Generation 60  
2.4 Assignments 101  
References 101

## 3 Architecture and Instruction Set of the C6x Processor

3.1 Introduction 102  
3.2 TMS320C6x Architecture 104  
3.3 Functional Units 105  
3.4 Fetch and Execute Packets 106  
3.5 Pipelining 108  
3.6 Registers 110  
3.7 Linear and Circular Addressing Modes 110  
3.7.1 Indirect Addressing 110  
3.7.2 Circular Addressing 111  
3.8 TMS320C6x Instruction Set 112  
3.8.1 Assembly Code Format 112  
3.8.2 Types of Instructions 113  
3.9 Assembler Directives 115  
3.10 Linear Assembly 116  
3.11 ASM Statement Within C 117  
3.12 C-Callable Assembly Function 117  
3.13 Timers 118  
3.14 Interrupts 118  
3.14.1 Interrupt Control Registers 118  
3.14.2 Interrupt Acknowledgment 120  
3.15 Multichannel Buffered Serial Ports 121  
3.16 Direct Memory Access 122  
3.17 Memory Considerations 122  
3.17.1 Data Allocation 122  
3.17.2 Data Alignment 123  
3.17.3 Pragma Directives 123  
3.17.4 Memory Models 124  
3.18 Fixed- and Floating-Point Format 124  
3.18.1 Data Types 124  
3.18.2 Floating-Point Format 125  
3.18.3 Division 126  
3.19 Code Improvement 126  
3.19.1 Intrinsics 126  
3.19.2 Trip Directive for Loop Count 127
3.19.3 Cross-Paths 127
3.19.4 Software Pipelining 127

3.20 Constraints 128
3.20.1 Memory Constraints 128
3.20.2 Cross-Path Constraints 128
3.20.3 Load/Store Constraints 129
3.20.4 Pipelining Effects with More Than One EP Within an FP 129

3.21 Programming Examples Using C, Assembly, and Linear Assembly 130
3.22 Assignments 142
References 145

4 Finite Impulse Response Filters 146

4.1 Introduction to the $z$-Transform 146
4.1.1 Mapping from $s$-Plane to $z$-Plane 149
4.1.2 Difference Equations 150

4.2 Discrete Signals 151
4.3 FIR Filters 152
4.4 FIR Lattice Structure 154
4.5 FIR Implementation Using Fourier Series 158
4.6 Window Functions 162
4.6.1 Hamming Window 163
4.6.2 Hanning Window 163
4.6.3 Blackman Window 164
4.6.4 Kaiser Window 164
4.6.5 Computer-Aided Approximation 164

4.7 Programming Examples Using C and ASM Code 165
4.8 Assignments 207
References 207

5 Infinite Impulse Response Filters 210

5.1 Introduction 210
5.2 IIR Filter Structures 211
5.2.1 Direct Form I Structure 212
5.2.2 Direct Form II Structure 212
5.2.3 Direct Form II Transpose 214
5.2.4 Cascade Structure 215
5.2.5 Parallel Form Structure 216

5.3 Bilinear Transformation 217
5.3.1 BLT Design Procedure 219

5.4 Programming Examples Using C and ASM Code 220
5.5 Assignments 252
References 253
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Fast Fourier Transform</td>
<td>255</td>
</tr>
<tr>
<td>6.1</td>
<td>Introduction</td>
<td>255</td>
</tr>
<tr>
<td>6.2</td>
<td>Development of the FFT Algorithm with Radix-2</td>
<td>256</td>
</tr>
<tr>
<td>6.3</td>
<td>Decimation-in-Frequency FFT Algorithm with Radix-2</td>
<td>257</td>
</tr>
<tr>
<td>6.4</td>
<td>Decimation-in-Time FFT Algorithm with Radix-2</td>
<td>263</td>
</tr>
<tr>
<td>6.5</td>
<td>Bit Reversal for Unscrambling</td>
<td>268</td>
</tr>
<tr>
<td>6.6</td>
<td>Development of the FFT Algorithm with Radix-4</td>
<td>269</td>
</tr>
<tr>
<td>6.7</td>
<td>Inverse Fast Fourier Transform</td>
<td>272</td>
</tr>
<tr>
<td>6.8</td>
<td>Programming Examples</td>
<td>273</td>
</tr>
<tr>
<td>6.8.1</td>
<td>Frame-Based Processing</td>
<td>280</td>
</tr>
<tr>
<td>6.8.2</td>
<td>Fast Convolution</td>
<td>297</td>
</tr>
<tr>
<td></td>
<td>References</td>
<td>318</td>
</tr>
<tr>
<td>7</td>
<td>Adaptive Filters</td>
<td>319</td>
</tr>
<tr>
<td>7.1</td>
<td>Introduction</td>
<td>319</td>
</tr>
<tr>
<td>7.2</td>
<td>Adaptive Structures</td>
<td>321</td>
</tr>
<tr>
<td>7.3</td>
<td>Adaptive Linear Combiner</td>
<td>324</td>
</tr>
<tr>
<td>7.4</td>
<td>Performance Function</td>
<td>327</td>
</tr>
<tr>
<td>7.5</td>
<td>Searching for the Minimum</td>
<td>329</td>
</tr>
<tr>
<td>7.6</td>
<td>Programming Examples for Noise Cancellation and System Identification</td>
<td>332</td>
</tr>
<tr>
<td></td>
<td>References</td>
<td>352</td>
</tr>
<tr>
<td>8</td>
<td>Code Optimization</td>
<td>354</td>
</tr>
<tr>
<td>8.1</td>
<td>Introduction</td>
<td>354</td>
</tr>
<tr>
<td>8.2</td>
<td>Optimization Steps</td>
<td>355</td>
</tr>
<tr>
<td>8.2.1</td>
<td>Compiler Options</td>
<td>355</td>
</tr>
<tr>
<td>8.2.2</td>
<td>Intrinsic C Functions</td>
<td>356</td>
</tr>
<tr>
<td>8.3</td>
<td>Procedure for Code Optimization</td>
<td>356</td>
</tr>
<tr>
<td>8.4</td>
<td>Programming Examples Using Code Optimization Techniques</td>
<td>356</td>
</tr>
<tr>
<td>8.5</td>
<td>Software Pipelining for Code Optimization</td>
<td>363</td>
</tr>
<tr>
<td>8.5.1</td>
<td>Procedure for Hand-Coded Software Pipelining</td>
<td>363</td>
</tr>
<tr>
<td>8.5.2</td>
<td>Dependency Graph</td>
<td>364</td>
</tr>
<tr>
<td>8.5.3</td>
<td>Scheduling Table</td>
<td>365</td>
</tr>
<tr>
<td>8.6</td>
<td>Execution Cycles for Different Optimization Schemes</td>
<td>372</td>
</tr>
<tr>
<td></td>
<td>References</td>
<td>373</td>
</tr>
<tr>
<td>9</td>
<td>DSP/BIOS and RTDX Using MATLAB, Visual C++, Visual Basic, and LabVIEW</td>
<td>374</td>
</tr>
<tr>
<td>9.1</td>
<td>Introduction to DSP/BIOS</td>
<td>374</td>
</tr>
<tr>
<td>9.1.1</td>
<td>Periodic Functions</td>
<td>376</td>
</tr>
<tr>
<td>9.1.2</td>
<td>Hardware Interrupts</td>
<td>378</td>
</tr>
</tbody>
</table>
## 10 DSP Applications and Student Projects

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1</td>
<td>DTMF Signal Detection Using Correlation, FFT, and Goertzel Algorithm</td>
<td>422</td>
</tr>
<tr>
<td>10.1.1</td>
<td>Using a Correlation Scheme and Onboard LEDs for Verifying Detection</td>
<td>424</td>
</tr>
<tr>
<td>10.1.2</td>
<td>Using RTDX with Visual C++ to Display Detected DTMF Signals on the PC</td>
<td>427</td>
</tr>
<tr>
<td>10.1.3</td>
<td>Using FFT and Onboard LEDs for Verifying Detection</td>
<td>429</td>
</tr>
<tr>
<td>10.1.4</td>
<td>Using Goertzel Algorithm</td>
<td>429</td>
</tr>
<tr>
<td>10.2</td>
<td>Beat Detection Using Onboard LEDs</td>
<td>429</td>
</tr>
<tr>
<td>10.3</td>
<td>FIR with RTDX Using Visual C++ for Transfer of Filter Coefficients</td>
<td>434</td>
</tr>
<tr>
<td>10.4</td>
<td>Radix-4 FFT with RTDX Using Visual C++ and MATLAB for Plotting</td>
<td>435</td>
</tr>
<tr>
<td>10.5</td>
<td>Spectrum Display Through EMIF Using a Bank of 32 LEDs</td>
<td>438</td>
</tr>
<tr>
<td>10.6</td>
<td>Spectrum Display Through EMIF Using LCDs</td>
<td>440</td>
</tr>
<tr>
<td>10.7</td>
<td>Time–Frequency Analysis of Signals with Spectrogram</td>
<td>445</td>
</tr>
<tr>
<td>10.7.1</td>
<td>Simulation Using MATLAB</td>
<td>446</td>
</tr>
<tr>
<td>10.7.2</td>
<td>Spectrogram with RTDX Using MATLAB</td>
<td>446</td>
</tr>
<tr>
<td>10.7.3</td>
<td>Spectrogram with RTDX Using Visual C++</td>
<td>450</td>
</tr>
<tr>
<td>10.8</td>
<td>Audio Effects (Echo and Reverb, Harmonics, and Distortion)</td>
<td>451</td>
</tr>
<tr>
<td>10.9</td>
<td>Voice Detection and Reverse Playback</td>
<td>453</td>
</tr>
<tr>
<td>10.10</td>
<td>Phase Shift Keying—BPSK Encoding and Decoding with PLL</td>
<td>454</td>
</tr>
<tr>
<td>10.10.1</td>
<td>BPSK Single-Board Transmitter/Receiver Simulation</td>
<td>455</td>
</tr>
<tr>
<td>10.10.2</td>
<td>BPSK Transmitter/Voice Encoder with Real-Time Input</td>
<td>459</td>
</tr>
<tr>
<td>10.10.3</td>
<td>Phase-Locked Loop</td>
<td>460</td>
</tr>
<tr>
<td>10.10.4</td>
<td>BPSK Transmitter and Receiver with PLL</td>
<td>465</td>
</tr>
<tr>
<td>10.11</td>
<td>Binary Phase Shift Keying</td>
<td>468</td>
</tr>
</tbody>
</table>
10.12 Modulation Schemes—PAM and PSK 470
  10.12.1 Pulse Amplitude Modulation 470
  10.12.2 Phase Shift Keying 474
10.13 Selectable IIR Filter and Scrambling Scheme Using Onboard Switches 479
10.14 Convolutional Encoding and Viterbi Decoding 482
10.15 Speech Synthesis Using Linear Prediction of Speech Signals 493
10.16 Automatic Speaker Recognition 496
10.17 μ-Law for Speech Companding 500
10.18 SB-ADPCM Encoder/Decoder: Implementation of G.722 Audio Coding 501
10.19 Encryption Using the Data Encryption Standard Algorithm 503
10.20 Phase-Locked Loop 506
10.21 Miscellaneous Projects 508
  10.21.1 Multirate Filter 508
  10.21.2 Acoustic Direction Tracker 513
  10.21.3 Neural Network for Signal Recognition 515
  10.21.4 Adaptive Temporal Attenuator 519
  10.21.5 FSK Modem 521
  10.21.6 Image Processing 521
  10.21.7 Filter Design and Implementation Using a Modified Prony’s Method 521
  10.21.8 PID Controller 522
  10.21.9 Four-Channel Multiplexer for Fast Data Acquisition 522
  10.21.10 Video Line Rate Analysis 522
Acknowledgments 522
References 523

Appendix A TMS320C6x Instruction Set 528
A.1 Instructions for Fixed- and Floating-Point Operations 528
A.2 Instructions for Floating-Point Operations 528
References 528

Appendix B Registers for Circular Addressing and Interrupts 530
Reference 530

Appendix C Fixed-Point Considerations 533
C.1 Binary and Two’s-Complement Representation 533
C.2 Fractional Fixed-Point Representation 536
C.3 Multiplication 536
Reference 539
Appendix D  MATLAB and Goldwave Support Tools  
D.1  fdatool for FIR Filter Design  
D.2  fdatool for IIR Filter Design  
D.3  MATLAB for FIR Filter Design Using the Student Version  
D.4  MATLAB for IIR Filter Design Using the Student Version  
D.5  Using the Goldwave Shareware Utility as a Virtual Instrument  
References  

Appendix E  Fast Hartley Transform  
References  

Appendix F  Goertzel Algorithm  
F.1  Design Considerations  
References  

Appendix G  TMS320C6416 DSK  
G.1  TMS320C64x Processor  
G.2  Programming Examples Using the C6416 DSK  
References  

Index  