FPGA-based Implementation of Signal Processing Systems

Roger Woods  
Queen’s University, Belfast, UK

John McAllister  
Queen’s University, Belfast, UK

Gaye Lightbody  
University of Ulster, UK

Ying Yi  
University of Edinburgh, UK
# Contents

**About the Authors** xv

**Preface** xviii

1 Introduction to Field-programmable Gate Arrays 1

1.1 Introduction 1

1.1.1 Field-programmable Gate Arrays 1

1.1.2 Programmability and DSP 3

1.2 A Short History of the Microchip 4

1.2.1 Technology Offerings 6

1.3 Influence of Programmability 7

1.4 Challenges of FPGAs 9

References 10

2 DSP Fundamentals 11

2.1 Introduction 11

2.2 DSP System Basics 12

2.3 DSP System Definitions 12

2.3.1 Sampling Rate 14

2.3.2 Latency and Pipelining 15

2.4 DSP Transforms 16

2.4.1 Fast Fourier Transform 16

2.4.2 Discrete Cosine Transform (DCT) 18

2.4.3 Wavelet Transform 19

2.4.4 Discrete Wavelet Transform 19

2.5 Filter Structures 20

2.5.1 Finite Impulse Response Filter 20

2.5.2 Correlation 23

2.5.3 Infinite Impulse Response Filter 23

2.5.4 Wave Digital Filters 24

2.6 Adaptive Filtering 27

2.7 Basics of Adaptive Filtering 27

2.7.1 Applications of Adaptive Filters 28

2.7.2 Adaptive Algorithms 30
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.7.3</td>
<td>LMS Algorithm</td>
</tr>
<tr>
<td>2.7.4</td>
<td>RLS Algorithm</td>
</tr>
<tr>
<td>2.8</td>
<td>Conclusions</td>
</tr>
<tr>
<td>References</td>
<td></td>
</tr>
</tbody>
</table>

### 3 Arithmetic Basics

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Introduction</td>
</tr>
<tr>
<td>3.2</td>
<td>Number Systems</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Number Representations</td>
</tr>
<tr>
<td>3.3</td>
<td>Fixed-point and Floating-point</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Floating-point Representations</td>
</tr>
<tr>
<td>3.4</td>
<td>Arithmetic Operations</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Adders and Subtracters</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Multipliers</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Division</td>
</tr>
<tr>
<td>3.4.4</td>
<td>Square Root</td>
</tr>
<tr>
<td>3.5</td>
<td>Fixed-point versus Floating-point</td>
</tr>
<tr>
<td>3.6</td>
<td>Conclusions</td>
</tr>
<tr>
<td>References</td>
<td></td>
</tr>
</tbody>
</table>

### 4 Technology Review

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Introduction</td>
</tr>
<tr>
<td>4.2</td>
<td>Architecture and Programmability</td>
</tr>
<tr>
<td>4.3</td>
<td>DSP Functionality Characteristics</td>
</tr>
<tr>
<td>4.4</td>
<td>Processor Classification</td>
</tr>
<tr>
<td>4.5</td>
<td>Microprocessors</td>
</tr>
<tr>
<td>4.5.1</td>
<td>The ARM Microprocessor Architecture Family</td>
</tr>
<tr>
<td>4.6</td>
<td>DSP Microprocessors (DSP(\mu)s)</td>
</tr>
<tr>
<td>4.6.1</td>
<td>DSP Micro-operation</td>
</tr>
<tr>
<td>4.7</td>
<td>Parallel Machines</td>
</tr>
<tr>
<td>4.7.1</td>
<td>Systolic Arrays</td>
</tr>
<tr>
<td>4.7.2</td>
<td>SIMD Architectures</td>
</tr>
<tr>
<td>4.7.3</td>
<td>MIMD Architectures</td>
</tr>
<tr>
<td>4.8</td>
<td>Dedicated ASIC and FPGA Solutions</td>
</tr>
<tr>
<td>4.9</td>
<td>Conclusions</td>
</tr>
<tr>
<td>References</td>
<td></td>
</tr>
</tbody>
</table>

### 5 Current FPGA Technologies

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Introduction</td>
</tr>
<tr>
<td>5.2</td>
<td>Toward FPGAs</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Early FPGA Architectures</td>
</tr>
<tr>
<td>5.3</td>
<td>Altera FPGA Technologies</td>
</tr>
<tr>
<td>5.3.1</td>
<td>MAX(^\text{\textregistered})7000 FPGA Technology</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Stratix(^\text{\textregistered}) I(\text{II}) FPGA Family</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Hardcopy(^\text{\textregistered}) Structured ASIC Family</td>
</tr>
<tr>
<td>5.4</td>
<td>Xilinx FPGA Technologies</td>
</tr>
<tr>
<td>5.4.1</td>
<td>Xilinx Virtex(^\text{\textTM})-5 FPGA Technologies</td>
</tr>
</tbody>
</table>
5.5 Lattice® FPGA Families
5.5.1 Lattice® ispXPLD 5000MX Family 103
5.6 Actel® FPGA Technologies
5.6.1 Actel® ProASIC® PLUS FPGA Technology 105
5.6.2 Actel® Antifuse SX FPGA Technology 106
5.7 Atmel® FPGA Technologies
5.7.1 Atmel® AT40K FPGA Technologies 108
5.7.2 Reconfiguration of the Atmel® AT40K FPGA Technologies 109
5.8 General Thoughts on FPGA Technologies 110
References 110

6 Detailed FPGA Implementation Issues 111

6.1 Introduction 111
6.2 Various Forms of the LUT 112
6.3 Memory Availability 115
6.4 Fixed Coefficient Design Techniques 116
6.5 Distributed Arithmetic 117
6.6 Reduced Coefficient Multiplier
6.6.1 RCM Design Procedure 122
6.6.2 FPGA Multiplier Summary 125
6.7 Final Statements 125
References 125

7 Rapid DSP System Design Tools and Processes for FPGA 127

7.1 Introduction 127
7.2 The Evolution of FPGA System Design
7.2.1 Age 1: Custom Glue Logic 128
7.2.2 Age 2: Mid-density Logic 128
7.2.3 Age 3: Heterogeneous System-on-chip 129
7.3 Design Methodology Requirements for FPGA DSP 129
7.4 System Specification
7.4.1 Petri Nets 129
7.4.2 Process Networks (PN) and Dataflow 131
7.4.3 Embedded Multiprocessor Software Synthesis 132
7.4.4 GEDAE 132
7.5 IP Core Generation Tools for FPGA
7.5.1 Graphical IP Core Development Approaches 133
7.5.2 Synplify DSP 134
7.5.3 C-based Rapid IP Core Design 134
7.5.4 MATLAB®-based Rapid IP Core Design 136
7.5.5 Other Rapid IP Core Design 136
7.6 System-level Design Tools for FPGA
7.6.1 Compaan 137
7.6.2 ESPAM 137
7.6.3 Daedalus 138
7.6.4 Koski 140
7.7 Conclusion 140
References 141
8 Architecture Derivation for FPGA-based DSP Systems 143

8.1 Introduction 143
8.2 DSP Algorithm Characteristics 144
8.2.1 Further Characterization 145
8.3 DSP Algorithm Representations 148
8.3.1 SFG Descriptions 148
8.3.2 DFG Descriptions 149
8.4 Basics of Mapping DSP Systems onto FPGAs 149
8.4.1 Retiming 150
8.4.2 Cut-set Theorem 154
8.4.3 Application of Delay Scaling 155
8.4.4 Calculation of Pipelining Period 158
8.5 Parallel Operation 161
8.6 Hardware Sharing 163
8.6.1 Unfolding 163
8.6.2 Folding 165
8.7 Application to FPGA 169
8.8 Conclusions 169
References 169

9 The IRIS Behavioural Synthesis Tool 171

9.1 Introduction of Behavioural Synthesis Tools 172
9.2 IRIS Behavioural Synthesis Tool 173
9.2.1 Modular Design Procedure 174
9.3 IRIS Retiming 176
9.3.1 Realization of Retiming Routine in IRIS 177
9.4 Hierarchical Design Methodology 179
9.4.1 White Box Hierarchical Design Methodology 180
9.4.2 Automatic Implementation of Extracting Processor Models from Previously Synthesized Architecture 181
9.4.3 Hierarchical Circuit Implementation in IRIS 184
9.4.4 Calculation of Pipelining Period in Hierarchical Circuits 185
9.4.5 Retiming Technique in Hierarchical Circuits 188
9.5 Hardware Sharing Implementation (Scheduling Algorithm) for IRIS 190
9.6 Case Study: Adaptive Delayed Least-mean-squares Realization 199
9.6.1 High-speed Implementation 200
9.6.2 Hardware-shared Designs for Specific Performance 205
9.7 Conclusions 207
References 207

10 Complex DSP Core Design for FPGA 211

10.1 Motivation for Design for Reuse 212
10.2 Intellectual Property (IP) Cores 213
10.3 Evolution of IP Cores 215
10.3.1 Arithmetic Libraries 216
10.3.2 Fundamental DSP Functions 218
10.3.3 Complex DSP Functions 219
10.3.4 Future of IP Cores 219
References 219
12.4.3 RLS Solved by QR Decomposition 278
12.4.4 Givens Rotations Used for QR Factorization 280
12.5 Algorithm to Architecture 282
12.5.1 Dependence Graph 283
12.5.2 Signal Flow Graph 283
12.5.3 Systolic Implementation of Givens Rotations 285
12.5.4 Squared Givens Rotations 287
12.6 Efficient Architecture Design 287
12.6.1 Scheduling the QR Operations 290
12.7 Generic QR Architecture 292
12.7.1 Processor Array 293
12.8 Retiming the Generic Architecture 301
12.8.1 Retiming QR Architectures 305
12.9 Parameterizable QR Architecture 307
12.9.1 Choice of Architecture 307
12.9.2 Parameterizable Control 309
12.9.3 Linear Architecture 310
12.9.4 Sparse Linear Architecture 310
12.9.5 Rectangular Architecture 316
12.9.6 Sparse Rectangular Architecture 316
12.9.7 Generic QR Cells 319
12.10 Generic Control 319
12.10.1 Generic Input Control for Linear and Sparse Linear Arrays 320
12.10.2 Generic Input Control for Rectangular and Sparse Rectangular Arrays 321
12.10.3 Effect of Latency on the Control Seeds 321
12.11 Beamformer Design Example 323
12.12 Summary 325
References 325

13 Low Power FPGA Implementation 329
13.1 Introduction 329
13.2 Sources of Power Consumption 330
13.2.1 Dynamic Power Consumption 331
13.2.2 Static Power Consumption 332
13.3 Power Consumption Reduction Techniques 335
13.4 Voltage Scaling in FPGAs 335
13.5 Reduction in Switched Capacitance 337
13.6 Data Reordering 337
13.7 Fixed Coefficient Operation 338
13.8 Pipelining 339
13.9 Locality 343
13.10 Application to an FFT Implementation 344
13.11 Conclusions 348
References 348
## 14 Final Statements

14.1 Introduction 351
14.2 Reconfigurable Systems 351
   14.2.1 Relevance of FPGA Programmability 352
   14.2.2 Existing Reconfigurable Computing 353
   14.2.3 Realization of Reconfiguration 354
   14.2.4 Reconfiguration Models 355
14.3 Memory Architectures 357
14.4 Support for Floating-point Arithmetic 358
14.5 Future Challenges for FPGAs 359
References 359

Index 361