

Contents

Preface	xv
1 Introduction to Digital Signal Processing Systems	1
1.1 Introduction	1
1.2 Typical DSP Algorithms	2
1.3 DSP Application Demands and Scaled CMOS Technologies	27
1.4 Representations of DSP Algorithms	31
1.5 Book Outline	40
References	41
2 Iteration Bound	43
2.1 Introduction	43
2.2 Data-Flow Graph Representations	43
2.3 Loop Bound and Iteration Bound	45
2.4 Algorithms for Computing Iteration Bound	47
2.5 Iteration Bound of Multirate Data-Flow Graphs	55
2.6 Conclusions	57
2.7 Problems	58
References	61

3	Pipelining and Parallel Processing	63
3.1	Introduction	63
3.2	Pipelining of FIR Digital Filters	64
3.3	Parallel Processing	69
3.4	Pipelining and Parallel Processing for Low Power	74
3.5	Conclusions	82
3.6	Problems	83
	References	88
4	Retiming	91
4.1	Introduction	91
4.2	Definitions and Properties	93
4.3	Solving Systems of Inequalities	95
4.4	Retiming Techniques	97
4.5	Conclusions	112
4.6	Problems	112
	References	118
5	Unfolding	119
5.1	Introduction	119
5.2	An Algorithm for Unfolding	121
5.3	Properties of Unfolding	124
5.4	Critical Path, Unfolding, and Retiming	127
5.5	Applications of Unfolding	128
5.6	Conclusions	140
5.7	Problems	140
	References	147
6	Folding	149
6.1	Introduction	149
6.2	Folding Transformation	151
6.3	Register Minimization Techniques	157
6.4	Register Minimization in Folded Architectures	163
6.5	Folding of Multirate Systems	170
6.6	Conclusions	174
6.7	Problems	174
	References	186
7	Systolic Architecture Design	189

7.1	Introduction	189
7.2	Systolic Array Design Methodology	190
7.3	FIR Systolic Arrays	192
7.4	Selection of Scheduling Vector	201
7.5	Matrix-Matrix Multiplication and 2D Systolic Array Design	205
7.6	Systolic Design for Space Representations Containing Delays	210
7.7	Conclusions	213
7.8	Problems	213
	References	223
8	Fast Convolution	227
8.1	Introduction	227
8.2	Cook-Toom Algorithm	228
8.3	Winograd Algorithm	237
8.4	Iterated Convolution	244
8.5	Cyclic Convolution	246
8.6	Design of Fast Convolution Algorithm by Inspection	250
8.7	Conclusions	251
8.8	Problems	251
	References	253
9	Algorithmic Strength Reduction in Filters and Transforms	255
9.1	Introduction	255
9.2	Parallel FIR Filters	256
9.3	Discrete Cosine Transform and Inverse DCT	275
9.4	Parallel Architectures for Rank-Order Filters	285
9.5	Conclusions	297
9.6	Problems	297
	References	310
10	Pipelined and Parallel Recursive and Adaptive Filters	313
10.1	Introduction	313
10.2	Pipeline Interleaving in Digital Filters	314
10.3	Pipelining in 1st-Order IIR Digital Filters	320
10.4	Pipelining in Higher-Order IIR Digital Filters	325
10.5	Parallel Processing for IIR filters	339
10.6	Combined Pipelining and Parallel Processing for IIR Filters	345

10.7	Low-Power IIR Filter Design Using Pipelining and Parallel Processing	348
10.8	Pipelined Adaptive Digital Filters	351
10.9	Conclusions	366
10.10	Problems	367
	References	374
11	Scaling and Roundoff Noise	377
11.1	Introduction	377
11.2	Scaling and Roundoff Noise	378
11.3	State Variable Description of Digital Filters	382
11.4	Scaling and Roundoff Noise Computation	386
11.5	Roundoff Noise in Pipelined IIR Filters	391
11.6	Roundoff Noise Computation Using State Variable Description	403
11.7	Slow-Down, Retiming, and Pipelining	405
11.8	Conclusions	410
11.9	Problems	410
	References	419
12	Digital Lattice Filter Structures	421
12.1	Introduction	421
12.2	Schur Algorithm	422
12.3	Digital Basic Lattice Filters	429
12.4	Derivation of One-Multiplier Lattice Filter	437
12.5	Derivation of Normalized Lattice Filter	444
12.6	Derivation of Scaled-Normalized Lattice Filter	447
12.7	Roundoff Noise Calculation in Lattice Filters	454
12.8	Pipelining of Lattice IIR Digital Filters	458
12.9	Design Examples of Pipelined Lattice Filters	464
12.10	Low-Power CMOS Lattice IIR Filters	469
12.11	Conclusions	470
12.12	Problems	470
	References	474
13	Bit-Level Arithmetic Architectures	477
13.1	Introduction	477
13.2	Parallel Multipliers	478
13.3	Interleaved Floor-plan and Bit-Plane-Based Digital Filters	489

13.4	Bit-Serial Multipliers	490
13.5	Bit-Serial Filter Design and Implementation	499
13.6	Canonic Signed Digit Arithmetic	505
13.7	Distributed Arithmetic	511
13.8	Conclusions	518
13.9	Problems	518
	References	527
14	Redundant Arithmetic	529
14.1	Introduction	529
14.2	Redundant Number Representations	530
14.3	Carry-Free Radix-2 Addition and Subtraction	531
14.4	Hybrid Radix-4 Addition	536
14.5	Radix-2 Hybrid Redundant Multiplication Architectures	540
14.6	Data Format Conversion	545
14.7	Redundant to Nonredundant Converter	547
14.8	Conclusions	551
14.9	Problems	552
	References	555
15	Numerical Strength Reduction	559
15.1	Introduction	559
15.2	Subexpression Elimination	560
15.3	Multiple Constant Multiplication	560
15.4	Subexpression Sharing in Digital Filters	566
15.5	Additive and Multiplicative Number Splitting	574
15.6	Conclusions	583
15.7	Problems	583
	References	589
16	Synchronous, Wave, and Asynchronous Pipelines	591
16.1	Introduction	591
16.2	Synchronous Pipelining and Clocking Styles	593
16.3	Clock Skew and Clock Distribution in Bit-Level Pipelined VLSI Designs	601
16.4	Wave Pipelining	606
16.5	Constraint Space Diagram and Degree of Wave Pipelining	612
16.6	Implementation of Wave-Pipelined Systems	614
16.7	Asynchronous Pipelining	619

16.8	Signal Transition Graphs	622
16.9	Use of STG to Design Interconnection Circuits	626
16.10	Implementation of Computational Units	631
16.11	Conclusions	640
16.12	Problems	640
	References	643
17	Low-Power Design	645
17.1	Introduction	645
17.2	Theoretical Background	648
17.3	Scaling Versus Power Consumption	650
17.4	Power Analysis	652
17.5	Power Reduction Techniques	662
17.6	Power Estimation Approaches	671
17.7	Conclusions	688
17.8	Problems	688
	References	692
18	Programmable Digital Signal Processors	695
18.1	Introduction	695
18.2	Evolution of Programmable Digital Signal Processors	696
18.3	Important Features of DSP Processors	697
18.4	DSP Processors for Mobile and Wireless Communications	703
18.5	Processors for Multimedia Signal Processing	704
18.6	Conclusions	714
	References	714
	Appendix A: Shortest Path Algorithms	717
A.1	Introduction	717
A.2	The Bellman-Ford Algorithm	718
A.3	The Floyd-Warshall Algorithm	720
A.4	Computational Complexities	721
	References	722
	Appendix B: Scheduling and Allocation Techniques	723
B.1	Introduction	723
B.2	Iterative/Constructive Scheduling Algorithms	725
B.3	Transformational Scheduling Algorithms	729
B.4	Integer Linear Programming Models	738

References	741
Appendix C: Euclidean GCD Algorithm	743
C.1 Introduction	743
C.2 Euclidean GCD Algorithm for Integers	743
C.3 Euclidean GCD Algorithm for Polynomials	745
Appendix D: Orthonormality of Schur Polynomials	747
D.1 Orthogonality of Schur Polynomials	747
D.2 Orthonormality of Schur Polynomials	749
Appendix E: Fast Binary Adders and Multipliers	753
E.1 Introduction	753
E.2 Multiplexer-Based Fast Binary Adders	753
E.3 Wallace Tree and Dadda Multiplier	758
References	761
Appendix F: Scheduling in Bit-Serial Systems	763
F.1 Introduction	763
F.2 Outline of the Scheduling Algorithm	764
F.3 Minimum Cost Solution	766
F.4 Scheduling of Edges with Delays	768
References	769
Appendix G: Coefficient Quantization in FIR Filters	771
G.1 Introduction	771
G.2 NUS Quantization Algorithm	771
References	774

Preface

Digital signal processing (DSP) is used in numerous applications such as video compression, digital set-top box, cable modems, digital versatile disk, portable video systems/computers, digital audio, multimedia and wireless communications, digital radio, digital still and network cameras, speech processing, transmission systems, radar imaging, acoustic beamformers, global positioning systems, and biomedical signal processing. The field of DSP has always been driven by the advances in DSP applications and in scaled very-large-scale-integrated (VLSI) technologies. Therefore, at any given time, DSP applications impose several challenges on the implementations of the DSP systems. These implementations must satisfy the enforced sampling rate constraints of the real-time DSP applications and must require less space and power consumption.

This book addresses the methodologies needed to design custom or semi-custom VLSI circuits for these applications. Many of the techniques presented in the book are also applicable for faster implementations using off-the-shelf programmable digital signal processors. This book is intended to be used as a textbook for first-year graduate or senior courses on VLSI DSP architectures, or DSP structures for VLSI or High-Performance VLSI system design. This book is also an excellent reference for those involved in algorithm or architecture or circuit design for DSP applications.

This book brings together the distinct fields of computer architecture theory and DSP. DSP computation is different from general-purpose computation in the sense that the DSP programs are nonterminating programs. In DSP

computation, the same program is executed repetitively on an infinite time series. The nonterminating nature can be exploited to design more efficient DSP systems by exploiting the dependency of tasks both within an iteration and among multiple iterations. Furthermore, long critical paths in DSP algorithms limit the performance of DSP systems. These algorithms need to be transformed for design of high-speed or low-area or low-power implementations. The emphasis of this book is on design of efficient architectures, algorithms, and circuits, which can be operated with either less area or power consumption or with higher speed or lower roundoff noise. The actual VLSI design of the circuits is not covered in this book.

DSP algorithms are used in various real-time applications with different sampling rate requirements that can vary from about 20 KHz in speech applications to over 500 MHz in radar and high-definition television applications. The computation requirement of a video compression system for high-definition TV (HDTV) can range from 10 to 100 gigaoperations per second. The dramatically different sample rate and computation requirements necessitate different architecture considerations for implementations of DSP algorithms. For example, in a speech application a time-multiplexed architecture may be preferred where many algorithm operations are mapped to the same hardware. However, the high-speed requirement in video applications can be met by one-to-one mapping of algorithm operations to processors. Thus it is important to study techniques to design not just a single architecture but a family of architectures out of which an appropriate architecture can be selected for a specified application.

The first part of the book (chapters 2 to 7) addresses several high-level architectural transformations that can be used to design families of architectures for a given algorithm. These transformations include pipelining, retiming, unfolding, folding, and systolic array design methodology. The second part of the book (chapters 8 to 12) deals with high-level algorithm transformations such as strength reduction, look-ahead and relaxed look-ahead. Strength reduction transformations are applied to reduce the number of multiplications in convolution, parallel finite impulse response (FIR) digital filters, discrete cosine transforms (DCTs), and parallel rank-order filters. Look-ahead and relaxed look-ahead transformations are applied to design pipelined direct-form and lattice recursive digital filters and adaptive digital filters, and parallel recursive digital filters. This part of the book exploits the interplay between algorithm design and integrated circuit implementations. The third part of the book (chapters 13 to 18) addresses architectures for VLSI addition, multiplication, and digital filters, and issues related to high-performance VLSI system design such as pipelining styles, low-power design, and architectures for programmable digital signal processors.

Chapter 1 of the book reviews various DSP algorithms and addresses their representation using block diagrams, signal flow graphs, and data-flow graphs. Chapter 2 addresses the iteration bound, which is a fundamental lower bound

on the iteration period of any recursive signal processing algorithm. Two algorithms are described for determining this bound. The next 5 chapters address various transformations for improving performance of digital signal processing implementations. In Chapter 3, the basic concepts of pipelining and parallel processing are reviewed and the use of these techniques in design of high-speed or low-power applications is demonstrated. Chapter 4 addresses the retiming transformation, which is a generalization of the pipelining approach. Chapter 5 addresses unfolding, which can be used to design parallel architectures. Chapters 6 and 7 address folding techniques used to design time-multiplexed architectures where area reduction is important. While Chapter 6 addresses folding of arbitrary data-flow graphs, Chapter 7 addresses folding of regular data-flow graphs based on systolic design methodology.

Chapters 8 to 12 address design of algorithm structures for various DSP algorithms based on algorithm transformations such as strength reduction, look-ahead and relaxed look-ahead, and scaling and roundoff noise in digital filters. Chapter 8 addresses fast convolution based on Cook-Toom and Winograd convolution algorithms. In Chapter 9, algorithmic strength reduction is exploited to reduce the number of multiplication operations in parallel FIR filters, discrete cosine transforms, and parallel rank-order filters. Design of fast Fourier transform (FFT) structures is also based on strength reduction transformations but is not covered in this book since it is covered in many introductory DSP textbooks. While it is easy to achieve pipelining and parallel processing in nonrecursive computations, recursive and adaptive digital filters cannot be easily pipelined or processed in parallel due to the presence of feedback loops. In Chapter 10, the look-ahead technique is discussed and is used to pipeline first-order infinite impulse response (IIR) digital filters. For higher order filters, two types of look-ahead techniques, clustered and scattered look-ahead, are discussed. It is shown that the scattered look-ahead technique guarantees stability in pipelined IIR filters. The parallel implementations of IIR digital filters and how to combine pipelining and parallel processing in these digital filters are also addressed. Adaptive digital filters are pipelined based on relaxed look-ahead, which are based on certain approximations or relaxations of look-ahead. Chapter 11 addresses scaling and roundoff noise, which are important for VLSI implementations of DSP systems using fixed-point arithmetic. Roundoff noise computation techniques cannot be applied to many digital filters. These filters are preprocessed using slowdown, pipelining and/or retiming so that every roundoff noise node can be expressed as a state variable. The direct-form IIR digital filters cannot meet the filter requirements in certain applications. Lattice digital filters may be better suited for these applications due to their excellent roundoff noise property. Chapter 12 presents Schur polynomials, orthonormality of Schur polynomials, and use of these polynomials to design basic (two multiplier and one multiplier), normalized, and scaled-normalized lattice digital filters. Pipelined implementation of these lattice digital filters is also discussed.

Chapters 13 to 18 address VLSI implementations of arithmetic operations

such as addition and multiplication and digital filters, high-performance VLSI system design issues such as pipelining styles and low-power design, and programmable digital signal processors. Design of adders and multipliers using various implementation styles, such as bit-parallel, bit-serial, and digit-serial, and various number systems such as two's complement, canonic signed digit, and carry-save are discussed in Chapter 13. This chapter also addresses distributed arithmetic. Chapter 14 addresses arithmetic architectures based on redundant or signed-digit implementations. The main advantage of redundant arithmetic lies in its carry-free property, which enables computation in both least significant bit and most significant bit first modes. Conversion from redundant to nonredundant and vice versa is also addressed. In these chapters, bit-serial multipliers are derived from bit-parallel designs by systolic design methodology. Residue arithmetic, which can be used for implementation of FIR digital filters and transforms, is not studied in this book. Chapter 15 presents strength reduction at numerical level to reduce the area and power consumption of two's complement and canonic signed digit number based digital filters. Chapter 16 discusses various pipelining styles, such as synchronous, wave, and asynchronous pipelining. Approaches to reduction of clock skew in synchronous systems and synthesis of interface circuits in asynchronous systems are also addressed. Chapter 17 on low-power design presents various approaches for reduction of power consumption at architectural and technology levels and for estimation of power consumption. Chapter 18 addresses various architectures used in programmable digital signal processors.

Seven appendixes in the book cover shortest path algorithms used for determining the iteration bound and for retiming, scheduling, and allocation techniques used for determining the folding sets for design of folded architectures; Euclid's GCD algorithm, which is used for Winograd's convolution; orthonormality of Schur polynomials used for design of lattice digital filters; fast bit-parallel addition and multiplication; scheduling techniques for bit-serial systems; and coefficient quantization in FIR filters.

The concepts in this book have been described in a technology-independent manner. The examples in this book are based on digital filters and transforms. Many real-time DSP systems make use of control flow constructs such as conditionals, interrupts, and jump. Design of control-dominated DSP systems is beyond the scope of this book. The exercises can be completed using any programming language such as MATLAB or C. Many application-driven problems have been included at the end of the chapters. For example, the problems at the end of the algorithmic strength reduction chapter address the use of fast filters in design of equalizers in communications systems, wavelets, two-dimensional FIR digital filters, and motion estimation. These problems introduce the reader to different applications where the concepts covered in the chapter can be applied.

This book is based on the material taught at the University of Minnesota in two current semester courses: EE 5329: VLSI Digital Signal Processing

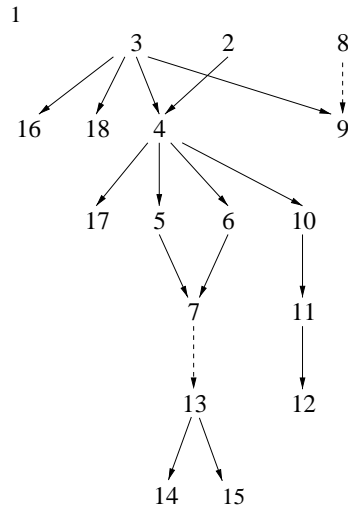


Fig. 0.1 Precedence constraints among different chapters.

Systems and EE 5549: Digital Signal Processing Structures for VLSI. EE 5329 (with a basic course on VLSI Design as prerequisite) covers chapters 2 through 7 and parts of chapters 13 through 18 (in that order). EE 5549 (with a basic course on digital signal processing as prerequisite) covers parts of chapters 2, 3, and 4, chapters 8 through 12, and some architectures for video compression based on journal and conference papers. These two semester courses were taught as three-quarter courses in the past. For a single semester course on VLSI Digital Signal Processing, chapters 2 through 7, parts of chapters 9, 10, 13 and 15, and an overview of topics in chapters 17 and 18 are recommended. However, the instructors can select the chapters that suit their needs.

The chapters need not be followed in the order they are presented. Many chapters can be taught independently. The precedence graph in Fig. 0.1 shows the dependencies among chapters. The dashed lines represent weak dependencies where a section of the current chapter is dependent on the preceding chapter.

The author has been fortunate to receive valuable help, support, and suggestions from numerous colleagues, students, and friends. The author is grateful to Leilei Song for her constant and enthusiastic help during the writing of this book. He is also grateful to Jin-Gyun Chung, Tracy Denk, David Parker, Janardhan Satyanarayana, and Ching-Yi Wang for their help during the early part of the writing of this book. The author is thankful to Wayne Burleson, Francky Catthoor, Ed F. Deprettere, Graham Jullien, and Naresh R. Shanbhag for their thorough and constructive reviews of the first draft; their comments have resulted in reorganization of several chapters in the book. Ed F. Deprettere and Scott Douglas used the preliminary versions

of the book at Delft University of Technology and at the University of Utah, respectively, and provided numerous suggestions.

The author appreciates the constant support and encouragement he has received from David G. Messerschmitt and Mos Kaveh. The author's research included in this book has been supported by the National Science Foundation, the Army Research Office, the Office of Naval Research, the Defense Advanced Research Projects Agency, Texas Instruments, Lucent Technologies, and NEC Corporation. The author is thankful to John Cozzens, Wanda Gass, Arup Gupta, Clifford Lau, Jose Munoz, Takao Nishitani, and Bill Sander for their encouragement.

Several chapters in the book are based on the joint research work of the author with his colleagues Jin-Gyun Chung, Tracy Denk, Kazuhito Ito, Lori Lucke, David G. Messerschmitt, Luis Montalvo, David Parker, Janardhan Satyanarayana, Naresh Shanbhag, H. R. Srinivas, and Ching-Yi Wang. The author also thanks many of his colleagues: Bryan Ackland, Jonathan Allen, Magdy Bayoumi, Don Boudlin, Robert W. Brodersen, Peter Cappello, Anantha Chandrakasan, Liang-Gee Chen, Gerhard Fettweis, Eby Friedman, Richard Hartley, Mehdi Hatamian, Sonia Heemstra, Yu Hen Hu, M. K. Ibrahim, Mary Irwin, Rajeev Jain, Leah Jamieson, Chein-Wei Jen, S.Y. Kung, Ichiro Kuroda, Edward Lee, K. J. R. Liu, Vijay Madisetti, John McCanny, Teresa Meng, Takao Nishitani, Tobias Noll, Robert Owens, Peter Pirsch, Miodrag Potkonjak, Jan Rabaey, Takayasu Sakurai, Edwin Sha, Bing Sheu, Michael Soderstrand, Mani Srivastava, Thanos Stouraitis, Earl Swartzlander, P. P. Vaidyanathan, Ingrid Verbauwhede, and Kung Yao. He has enjoyed numerous interactions with them. This book has been directly or indirectly influenced by these interactions. Thanks are also due to Carl Harris of Kluwer Academic Publishers for his permitting the author to reprint several parts of chapters 11 and 12 from an earlier monograph.

The author thanks Andrew Smith of John Wiley & Sons for his personal interest in this topic and for having invited the author to write this book. He also thanks Angioline Loredó, associate managing editor at Wiley, for her help in production of this book. It was truly a pleasure to work with them.

KESHAB K. PARHI

Minneapolis, MN