Systolic Signal Processing Systems

edited by
Earl E. Swartzlander, Jr.

Systolic Signal Processing Systems

edited by

Earl E. Swartzlander, Jr.
TRW Inc.
Redondo Beach, California

Marcel Dekker, Inc. New York • Basel

Library of Congress Cataloging-in-Publication Data

Systolic signal processing systems.

Includes index.
1. Signal processing--Digital techniques.
I. Swartzlander, Earl E.
TK5102.5.S663 1987 621.38'043 87-3467
ISBN 0-8247-7717-4

MARCEL DEKKER, INC.
270 Madison Avenue, New York, New York 10016

Current printing (last digit):
10 9 8 7 6 5 4 3 2 1

PRINTED IN THE UNITED STATES OF AMERICA

Systolic Signal Processing Systems

Preface

The field of special purpose digital processor development is evolving rapidly. This results in part from advances in VLSI circuit design and fabrication technology. Specifically, through computer aided design (CAD), silicon compilation, and VLSI fabrication advances it has become practical to create custom digital processors for a wide range of military, industrial, and commercial applications. The processor architecture must respect the constraints imposed by the design and fabrication technologies. The repetitive use of common structures (i.e., modules) and the minimization of "random" interconnections are crucial to reducing the design effort and to creating producable integrated circuits.

This book is about systolic signal processing systems: networks of signal processors with extremely efficient data flow between the processors. The terminology is new: systolic systems were first proposed about a decade ago by Professor H. T. Kung and his students, including Charles Leiserson. This concept represents a significant evolution from earlier pipelined digital signal processors.

Pipeline architectures have been employed for several decades to implement analog filters and high speed radar signal processors. The basic concept is that a cascade chain of processors can be provided with storage between the processors so that multiple data enter "into the pipeline" while previous data are being processed. The resulting data flow is similar to a firefighters' bucket brigade; in both cases the rate of flow is determined by the speed of the individual processors and not by the number of stages in the pipeline.

Systolic systems are so named because data flows through them in a rhythmic fashion similar to the flow of blood through a body. They feature data connections only between neighboring processors, use

multiple copies of a few types of processors, and avoid the use of global memory. These attributes are well matched to the constraints of VLSI technology. Localized interconnection (i.e., data flow only between neighboring processors and no global memory) reduces the need for exotic high-pin-count packaging, and allows increasing numbers of processors to be placed on a chip (or wafer) as technology evolves. Use of a few types of processing elements greatly simplifies the VLSI design challenge.

Although much attention has been paid to the architecture of general-purpose digital computers, relatively little has been devoted to special purpose signal processing systems where the benefit from the use of systolic architectures is greatest. This book represents a joint effort by about a dozen internationally recognized experts to improve the understanding of such systems. Chapters are included describing the basic concepts, current systems, new approaches, and extensions to the future. Our ultimate purpose is to foster a greater awareness of the techniques, technologies, and benefits of systolic processor architectures for signal processing applications.

This book is written for advanced students, engineers, and managers who wish a concise introduction to the key concepts and future directions of systolic processor architectures. Working engineers and managers may find it useful to attend workshops and symposia, such as those sponsored by the IEEE, to extend their understanding.

The first chapter provides an introduction to the use of mathematical modeling for the design of pipelined signal processors. Dr. Danny Cohen shows the relationship between the functions performed by circuit elements (and their interconnection) and the resulting processor's operation, allowing the designer to better understand what a given processor will do prior to its construction.

In chapter 2, Professor Per E. Danielsson describes systolic convolvers. Since the convolution operation is the cornerstone of modern signal processing, the concepts are of great importance. A wide variety of serial/parallel multiplier cells (which are cascaded to perform convolution) are described. This chapter introduces many of the concepts of circuit level pipelining and systolic processing.

Professor H. T. Kung is credited with developing the concept of systolic processing. In chapter 3, he describes the CMU Warp processor (the term Warp denoting ultra high speed is derived from the "Star Trek" television series). Warp is a 32-bit floating point systolic array capable of computing at rates of 10 million floating point operations per cell. This chapter shows how various algorithms are implemented on it.

In chapter 4, Professor S. Y. Kung describes the wavefront array concept. This represents an extension to systolic architecture in that data move in self-timed "waves" across the array. This chapter

develops the concept and shows how algorithms are implemented on arrays using data flow graphs.

In the next chapter, Professor Peter Cappello provides a methodology for making trades between space and time in processor arrays. In general, adding more processors (i.e., increasing space) does not produce a proportionate reduction in processing time. This chapter shows ways to design arrays that achieve more nearly optimum performance.

The development of a methodology to optimize the array implementation for a given algorithm is extended in chapter 6 by Dr. Sailesh Rao and Professor Thomas Kailath. Their approach is based on expressing the problem as a Regular Iterative Algorithm (RIA) and developing procedures to implement the RIA efficiently. Their work provides a mathematical framework for architecture development and comparison.

In chapter 7, Professor Tom Leighton and Charles E. Leiserson show how systolic algorithms apply to the development of defect tolerant architectures for Wafer Scale Integration (WSI). In WSI, circuits are implemented on integrated circuit wafers (which may contain many defective circuit elements). As shown in this chapter, systolic architectures facilitate reconfiguration to bypass defective circuits.

Finally, in chapter 8, Professors Renato Stefanelli and Mariagiovanna Sami examine fault tolerance in the context of a two-dimensional array. This work leads to a protocol for dynamic fault circumvention. Although developed for a two-dimensional array, it should also be applicable to other regular structures.

Some users may be troubled by variations in the style of presentation; this is one of the disadvantages of a multiauthor book. Exposure to the pioneers who are developing this field should help to compensate for the inconsistencies.

This book was developed with significant support from the staff at Marcel Dekker, Inc., and my secretary, Lauren Hall. Thanks are due to all of them for their help. On behalf of the technical community, it is a pleasure to extend special thanks to the authors who made time in their already busy schedules to document their work, thereby sharing it with all of us.

EARL E. SWARTZLANDER, JR.

Contributors

PETER R. CAPPELLO *Computer Science Department, University of California, Santa Barbara, California*

DANNY COHEN *University of Southern California Information Sciences Institute, Marina del Rey, California*

PER E. DANIELSSON *Department of Electrical Engineering, Linköping University, Linköping, Sweden*

THOMAS KAILATH *Information Systems Laboratory, Stanford University, Stanford, California*

H. T. KUNG *Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania*

SUN-YUAN KUNG *Signal & Image Processing Institute, Department of Electrical Engineering, University of Southern California, Los Angeles, California*

TOM LEIGHTON *Mathematics Department and Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts*

CHARLES E. LEISERSON *Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts*

SAILESH K. RAO *AT&T Bell Laboratories, Holmdel, New Jersey*

MARIAGIOVANNA SAMI *Department of Electronics, Politecnico di Milano, Milano, Italy*

RENATO STEFANELLI *Department of Electronics, Politecnico di Milano, Milano Italy*

Contents

ELECTRICAL ENGINEERING AND ELECTRONICS

A Series of Reference Books and Textbooks

Editors

Marlin O. Thurston
Department of Electrical Engineering
The Ohio State University
Columbus, Ohio

William Middendorf
Department of Electrical and Computer Engineering
University of Cincinnati
Cincinnati, Ohio

1. Rational Fault Analysis, *edited by Richard Saeks and S. R. Liberty*
2. Nonparametric Methods in Communications, *edited by P. Papantoni-Kazakos and Dimitri Kazakos*
3. Interactive Pattern Recognition, *Yi-tzuu Chien*
4. Solid-State Electronics, *Lawrence E. Murr*
5. Electronic, Magnetic, and Thermal Properties of Solid Materials, *Klaus Schröder*
6. Magnetic-Bubble Memory Technology, *Hsu Chang*
7. Transformer and Inductor Design Handbook, *Colonel Wm. T. McLyman*
8. Electromagnetics: Classical and Modern Theory and Applications, *Samuel Seely and Alexander D. Poularikas*
9. One-Dimensional Digital Signal Processing, *Chi-Tsong Chen*
10. Interconnected Dynamical Systems, *Raymond A. DeCarlo and Richard Saeks*
11. Modern Digital Control Systems, *Raymond G. Jacquot*
12. Hybrid Circuit Design and Manufacture, *Roydn D. Jones*
13. Magnetic Core Selection for Transformers and Inductors: A User's Guide to Practice and Specification, *Colonel Wm. T. McLyman*
14. Static and Rotating Electromagnetic Devices, *Richard H. Engelmann*
15. Energy-Efficient Electric Motors: Selection and Application, *John C. Andreas*
16. Electromagnetic Compossibility, *Heinz M. Schlicke*
17. Electronics: Models, Analysis, and Systems, *James G. Gottling*

18. Digital Filter Design Handbook, *Fred J. Taylor*

19. Multivariable Control: An Introduction, *P. K. Sinha*

20. Flexible Circuits: Design and Applications, *Steve Gurley, with contributions by Carl A. Edstrom, Jr., Ray D. Greenway, and William P. Kelly*

21. Circuit Interruption: Theory and Techniques, *Thomas E. Browne, Jr.*

22. Switch Mode Power Conversion: Basic Theory and Design, *K. Kit Sum*

23. Pattern Recognition: Applications to Large Data-Set Problems, *Sing-Tze Bow*

24. Custom-Specific Integrated Circuits: Design and Fabrication, *Stanley L. Hurst*

25. Digital Circuits: Logic and Design, *Ronald C. Emery*

26. Large-Scale Control Systems: Theories and Techniques, *Magdi S. Mahmoud, Mohamed F. Hassan, and Mohamed G. Darwish*

27. Microprocessor Software Project Management, *Eli T. Fathi and Cedric V. W. Armstrong (Sponsored by Ontario Centre for Microelectronics)*

28. Low Frequency Electromagnetic Design, *Michael P. Perry*

29. Multidimensional Systems: Techniques and Applications, *edited by Spyros G. Tzafestas*

30. AC Motors for High-Performance Applications: Analysis and Control, *Sakae Yamamura*

31. Ceramic Materials for Electronics: Processing, Properties, and Applications, *edited by Relva C. Buchanan*

32. Microcomputer Bus Structures and Bus Interface Design, *Arthur L. Dexter*

33. End User's Guide to Innovative Flexible Circuit Packaging, *Jay J. Miniet*

34. Reliability Engineering for Electronic Design, *Norman B. Fuqua*

35. Design Fundamentals for Low-Voltage Distribution and Control, *Frank W. Kussy and Jack L. Warren*

36. Encapsulation of Electronic Devices and Components, *Edward R. Salmon*

37. Protective Relaying: Principles and Applications, *J. Lewis Blackburn*

38. Testing Active and Passive Electronic Components, *Richard F. Powell*

39. Adaptive Control Systems: Techniques and Applications, *V. V. Chalam*
40. Computer-Aided Analysis of Power Electronic Systems, *Venkatachari Rajagopalan*
41. Integrated Circuit Quality and Reliability, *Eugene R. Hnatek*
42. Systolic Signal Processing Systems, *edited by Earl Swartzlander*
43. Adaptive Digital Filters and Signal Analysis, *Maurice Bellanger*

Additional Volumes in Preparation

Computer Systems Engineering Management, *Robert S. Alford*

Electronic Ceramics: Properties, Configuration, and Applications, *edited by Lionel M. Levinson*

Electrical Engineering-Electronics Software

1. Transformer and Inductor Design Software for the IBM PC, *Colonel Wm. T. McLyman*
2. Transformer and Inductor Design Software for the Macintosh, *Colonel Wm. T. McLyman*
3. Digital Filter Design Software for the IBM PC, *Fred J. Taylor and Thanos Stouraitis*

1

A Mathematical Approach to Computational Network Design

DANNY COHEN *University of Southern California Information Sciences Institute, Marina del Rey, California*

1 INTRODUCTION

In this chapter we demonstrate the application of a precise mathematical notation to the design and manipulation of computational networks. This notation captures the concepts of arithmetic operations (such as addition and multiplication) and of timing (e.g., delay). Once a design is expressed by means of such a mathematical notation, it can be evaluated objectively against a predefined set of design objectives, such as performance and cost. Throughout this chapter the term "design" means the structure/architecture of the computational network. This term is the hardware equivalent of the software term "algorithm."

In Section 2 we define the design objectives that guide the examples in this chapter. Obviously, other sets of design objectives may be used without deviating from the spirit of the chapter.

In Section 3 we deal with the implementation of a finite impulse response (FIR) filter, a typical signal processing problem. In that section, several designs are suggested and evaluated objectively, and the mathematical notation to express them is developed, in parallel. Here we consider first a design that follows closely the mathematical definition of the FIR filter. Later this design is transformed several times in order to improve it with respect to the predefined design objectives. Here the graphic representations of the designs are the source of the intuition for the synthesis, and their mathematical representations are mainly a means for the analysis, such as verifying the correctness of the various transformations of the design.

In Section 4 the same technique is applied to synthetic aperture radar (SAR) processing. We consider and evaluate several designs that result directly from the mathematical definition and the notation.

In Section 5 the same techniques and notation are applied to polynomial multiplication, division, and simultaneous multiplication and division. In that section the mathematical representation is the guiding force for the synthesis, and the graphic representations are used only for demonstration.

In Section 6 we discuss a special type of filter, the spectral filter. Operational calculus is used on the mathematical notation for deriving the most efficient implementation network for that filter. This mathematical notation is a very powerful tool, complementing the intuition based on conventional graphic representation. It could be used for both the synthesis and analysis of computational networks.

2 THE DESIGN GOALS

To achieve an optimal design, it is necessary to define the design objectives. The following objectives are typically considered to be important:

1. Correctness and accuracy
2. High computation rate
3. Low delay
4. Low parts count
5. Modularity, simplicity, etc.
6. Low power
7. Small size
8. Low cost

Obviously, this is only a partial list. For different applications the relative weights of these objectives may vary. It is generally accepted that (1) is the most important, even though there is evidence that this is not always the case. In some cases (8) is the dominant factor; in others, it is (6) and (7). Here (1) through (5) are considered to be the most important, in that order.

3 FIR-FILTER EXAMPLE

Consider the finite impulse response (FIR) filter, defined by

$$y_n = \sum_{i=1}^{N} a_i x_{n-i} \qquad (1.1)$$

This is a nonrecursive filter of the Nth order. Each output (Y) is a weighted average of the previous N inputs (X). Typically, the X sequence is a time series, and the x's are available sequentially, with x_i preceding x_{i+1} by one "cycle."

3.1 Z Operator

Let Z be the "delay-by-one-cycle" operator such that $Zx_i = x_{i-1}$. In systems controlled by a central master clock, this Z operator may be implemented by a simple register. Z^0 is defined to be the unit operator and Z^n is defined by $Z^n = ZZ^{n-1}$. It may be implemented by an n-stage first-in, first-out (FIFO) shift register.

We use the following properties of the Z operator:

1. $Z^n x_i = x_{i-n}$.
2. $ZF(x) = F(Zx)$.
3. If C is time invariant, $ZC = C$.

Z^n with $n < 0$ is a prediction by $|n|$ steps into the future. Since prediction of external input is not easy to implement with conventional logic, it is advisable to use only $n \geqslant 0$ when applying the Z^n operator to external input.

3.2 FIR-Filter Implementation

The expression

$$y_n = \sum_{i=1}^{N} a_i x_{n-i} \tag{1.1}$$

may also be written as

$$y_n = \sum_{i=1}^{N} a_i Z^i x_n$$

Using operator-calculus notation, this may be written as

$$Y = \left(\sum_{i=1}^{N} a_i Z^i\right) X$$

For N = 4 this yields