

AD-A270 976

## Systolic Design with Asynchronous Controls for Digital-Signal Processing

**Final Report** 

Author:

Dr. Kamran Reihani

July 15, 1993

U. S. Army Research Office Grant: DAAL03-90-G-0211

Physical Science Laboratory New Mexico State University Box 30002 Las Cruces, New Mexico 88003-0002

The views, opinions, and/or findings contained in this report are those of the author and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

Public reporting purgen for this collection of in	OCUMENTATION PA	GE	OMB No. 0704-0188
Jathering and maintaining the data needed, an collection of information, including suggestion Davis Histway, Suite 1204, Arlington, VA, 2220.	formation is estimated to average 1 hour per re dicompleting and reviewing the collection of in i for reducing this ourden to Washington Head 24302, and to the Office of Management and B	sponse, including the time for r formation - Send comments reg- quarters Services, Directorate fo udget, Paperwork Reduction Pro	eviewing instructions, searching existing data source arding this burden estimate or any other aspect of th or information Operations and Reports, 1215 jeffersi ject (0704-0188), Washington - DC 20503
1. AGENCY USE ONLY (Leave blar	07/15/93	3. REPORT TYPE AN Final: 08/0	D DATES COVERED 08/91 - 07/15/93
4. TITLE AND SUBTITLE Systolic Design with Signal Processing	Asynchronous Controls	for Digital-	5. FUNDING NUMBERS DAAL03-90-G-0211
6 AUTHOR(S) Dr. Kamran Reihani	······		
PERFORMING ORGANIZATION N Physical Science Labo New Mexico State Univ Box 30002 Las Cruces, NM 88003	AME(S) AND ADDRESS(ES) Dratory Versity 3-0002		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING / MONITORING AGI U.S. Army Research	ENCY NAME(S) AND ADDRESS(ES) Office		10. SPONSORING / MONITORING AGENCY REPORT NUMBER
Research Triangle F	Park, NC 27709-2211		ARO 28,76.9-MA SA
Approved for public	release; distribution	unlimited.	
<ul> <li>13. ABSTRACT (Maximum 200 word The research sponsore and technological bas processing algorithms</li> <li>Data reduction tec</li> <li>Conversion of sequence systolic mesh that for enabling variod</li> <li>Devising a hybrid of computations ex- communication (PE) merging of the spin</li> </ul>	(s) ed by this grant is foc sis for designing effic a. The major contribut chniques via the utiliz tential input signal in t is suitable for paral ous array dimensions of SA and data-flow ap tecuted in each process of protocols resolve the tral and asynchronous s	cused on the deviation of algor: ations of the res ation of algor: to input blocks lel processing oproach, making sor, thus reduct data-flow cont systolic array a	velopment of a theoretica arrays for digital-signal search are the following: ithm properties s by means of a spiral and that is flexible data streams independent ing waiting time. The flicts created by the architecture
<ul> <li>13. ABSTRACT (Maximum 200 word The research sponsore and technological bas processing algorithms</li> <li>Data reduction ted</li> <li>Conversion of sequence systolic mesh that for enabling variod</li> <li>Devising a hybrid of computations ex- communication (PE) merging of the spin</li> <li>14. SUBJECT TERMS Asynchronic cation protocol, Compu- propagation, Digital-Selement, Spinal mesh</li> </ul>	(s) (d) by this grant is foc (a) for designing effic (a). The major contribut (chniques via the utiliz (chniques via	cused on the dev cient systolic a cions of the res cation of algor: nto input blocks lel processing oproach, making sor, thus reduct e data-flow cont systolic array a cernel cycle, Co ction matrices, cithms, Process	velopment of a theoretica arrays for digital-signal search are the following: ithm properties s by means of a spiral and that is flexible data streams independent ing waiting time. The flicts created by the architecture ommuni 15. NUMBER OF PAGES 31 pages 16. PRICE CODE

GENERAL INSTRUCTIONS	FOR COMPLETING SF 298		
The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to stay within the lines to meet optical scanning requirements.			
Block 1 Agency Use Only (Leave blank)	Block 12a. <u>Distribution/Availability Statement</u> . Denotes public availability or limitations. Cite any		
Block 2. <u>Report_Date</u> . Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.	availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).		
Block 3. <u>Type of Report and Dates Covered</u> . State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).	DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."		
Block 4. <u>Title and Subtitle</u> . A title is taken from the part of the report that provides the most meaningful and complete information. When a	NASA - See Handbook NHB 2200.2. NTIS - Leave blank.		
report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification	Block 12b. <u>Distribution Code</u> . DOD - Leave blank.		
Block 5. Funding Numbers. To include contract	DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical		
and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:	Reports. NASA - Leave blank. NTIS - Leave blank.		
C- ContractPR- ProjectG- GrantTA- TaskPE- ProgramWU- Work UnitElementAccession No.	Block 13. <u>Abstract</u> . Include a brief (Maximum 200 words) factual summary of the most significant information contained in the report.		
Block 6. <u>Author(s)</u> Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the	Block 14. <u>Subject Terms</u> . Keywords or phrases identifying major subjects in the report.		
the name(s).	Block 15. <u>Number of Pages</u> . Enter the total number of pages.		
BIOCK 7. Performing Organization Name(s) and Address(es). Self-explanatory. Block 8. Performing Organization Report	Block 16. Price Code. Enter appropriate price code (NTIS only).		
number(s) assigned by the organization performing the report.	Blocks 17 19. Security Classifications. Self- explanatory Enter U.S. Security Classification in		
Block 9. Sponsoring/Monitoring Agency_Name(s) and_Address(es)_Self-explanatory.	accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and		
Block 10. Sponsoring/Monitoring Agency Report Number. (If known)	bottom of the page.		
<b>Block 11.</b> <u>Supplementary Notes</u> . Enter information not included elsewhere such as: Prepared in cooperation with; Trans. of; To be published in When a report is revised, include a statement whether the new report supersedes or supplements the older report	Block 20. <u>Limitation of Abstract</u> . This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be uplimited		

Standard Form 298 Back (Rev. 2-89)

# TABLE OF CONTENTS

.

# Page No.

LIST OF FIGURES	2
1. STATEMENT OF RESEARCH PROBLEM	3
2. SUMMARY OF RESULTS         2.1 Theoretical Aspect         2.2 System Aspect         2.2.1 Spiral Architecture         2.2.2 Asynchronous Architecture         2.3 Performance Aspect         2.4 References	4 6 8 3 8
3. PUBLICATIONS AND REPORTS 2	!1
4. PERSONNEL SUPPORTED AND DEGREES AWARDED	22
APPENDIX A	23
APPENDIX B	!5
APPENDIX C	27
APPENDIX D	<u>!9</u>

DTIC QUALITY INCOME LOS

Accest	ion for	1
NTIS	GRALI	
DTIC 3	TAB	
Unannounced		
Just1	firstion.	
Disi.r Avel	instion/ Jatiiity	todas
Titat	sheatt	ajor Al
0.80		
A-1		r

## LIST OF FIGURES

		Pa	ge	No	5.
Fig. 1.	One-step adaptive lattice predictor, L = 3		•••	•	5
Fig. 2.	Computational graph for $\ldots$ and $\varepsilon_{b,k}$			•	7
Fig. 3.	Computational scheme for $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\varepsilon_{\ell,k}$ and $\varepsilon_{b,k}$			•	7
Fig. 4.	Interconnection scheme for $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\varepsilon_{t,k}$ and $\varepsilon_{b,k}$			•	7
Fig. 5.	Proposed PE <sub>k</sub> for $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\varepsilon_{t,k}$ and $\varepsilon_{b,k}$				7
Fig. 6.	Propagation of data between array PEs			•	8
Fig. 7.	Spiral systolic array architecture for		•••		9
Fig. 8.	Inputs and outputs of a PE	• • •		1	0
Fig. 9.	PE input/output data distribution forkernel cycle t = 9			1	2
Fig. 10	Proposed protocol in (type I)	<b></b>		1	7

### 1. STATEMENT OF RESEARCH PROBLEM

The systolic algorithm approach is currently the most effective design procedure for minimizing computing time and the number of processors. Systolic arrays (SAs) are examples of VLSI special purpose processor networks that realize computationally expensive algorithms in hardware, and can achieve massive concurrency. Such arrays are numerously used for the real-time implementation of digital-signal processing algorithms, because of the properties that these algorithms possess for their effective mapping onto high-speed SA architectures. The main problem in a systolic approach is that extra delays are needed to assure proper timing and synchronization, which will collectively slow down computation and, thereby, reduce the throughput rate. For a large-scale array, this synchronization can become particularly tedious. Moreover, algorithms for asynchronous SAs (i) have not been fully utilized for data reduction, and (ii) the conversion of sequential input signals into input blocks that will alter the organization of the systolic mesh has not been performed.

The research sponsored by this grant focused on the following important technical issues that result in the development of efficient SAs.

- Data reduction techniques via the utilization of algorithm properties.
- Conversion of sequential input signals into input blocks by means of a spiral systolic mesh that is suitable for parallel processing and that is flexible for enabling various array dimensions.
- Making data streams independent of computations executed in each processor, thus reducing waiting time.

The main objective of this research grant is to develop a theoretical and technological basis for designing an asynchronous SA — that is a hybrid of SA and data-flow approach — one capable of converting input signals into input blocks for producing a faster and more flexible system.' The key to this design is a spiral SA structure, self-timed processors, and communication protocols to get control of data streams so that each computation can start if all its data are available. Moreover, the communication protocols resolve the data-flow conflicts created by the merging of the spiral and asynchronous SA architectures. The SA processes the input signal efficiently and eliminates the complex shift register organization of traditional filter realizations. Incorporated in this design are maximum parallelism and pipelinability. trade-off among computations, communications, and memory. Furthermore, the systolic array will use simple local interconnections without undesirable properties, such as preloading input data or global broadcasting.

For most digital signal processing algorithms.

## 2. SUMMARY OF RESULTS

The research performed during this grant focused on the improvement of present methods of designing systolic arrays [1-7] for the following digital signal processing algorithms [8-15]:

- discrete Fourier transform (DFT)
- inverse discrete Fourier transform (IDFT)
- convolution
- correlation
- linear phase filter (LPF)
- discrete Hadamard transform (DHT)
- one-step adaptive lattice predictors
- arbitrarily large LMS adaptive filters

This required a comprehensive research effort that addresses theoretical, system, and performance aspects.

**Theoretical Aspect**: Describe useful mathematical representations for digital-signal processing (DSP) algorithms.

**System Aspect**: Develop a technique for tailoring the algorithms into forms suitable for mapping onto SA architectures and to develop array systems for implementing the architectures efficiently.

**Performance Aspect**: Evaluate the performance of the new algorithms developed especially for the asynchronous array.

## 2.1 Theoretical Aspect

The mathematical representations for DSP algorithms include important properties such as local and regular data broadcasting. On the basis of these representations, we have arrived at a design procedure for creating a systolic array and a systolic algorithm for the realization of DSP algorithms efficiently. For example, lattice filters offer the following features:

- 1. They are very efficient structures, because of their simultaneous production of the forward and backward prediction errors.
- 2. The various stages of the lattice predictors are decoupled from each other. Moreover, the backward prediction errors generated by the stages of a lattice predictor are orthogonal to each other for wide-sense stationary input data.
- 3. They are modular in structure; an order increase is induced by adding one or more stages without affecting previous computations.
- 4. The structure of each of the stages is identical, making lattice filters viable candidates for VLSI implementation.





Fig. 1. One-step adaptive lattice predictor, L = 3.

The forward and backward signals are labeled  $s_{l,k}$  and  $s'_{l,k}$ , respectively. The relations describing a **one-step adaptive lattice predictor** are

$$s_{0, k} = s'_{0, k} = x_{k}$$

$$s_{l+1, k} = s_{l, k} + K_{l, k} s'_{l, k-1} \qquad 0 \le l \le L-1$$

$$s'_{l+1, k} = K_{l, k} s_{l, k} + s'_{l, k-1} \qquad 0 \le l \le L-1$$

$$\varepsilon_{f, k} = s_{L, k}$$

$$\varepsilon_{b, k} = s'_{L, k}$$
(1)

where,  $\varepsilon_{t,k}$  and  $\varepsilon_{b,k}$  are the respective forward and backward predicted errors. The LMS algorithm for this lattice is

$$K_{I, k+1} = K_{I, k} - 2\mu_{I}s_{I+1, k}s'_{I, k-1} \qquad 0 \le I \le L - 1$$
(2)

Here,  $\mu_i$  is the so-called convergence parameter, which is time invariant. This parameter is set to different values from stage to stage [13]. The **data reduction** technique can be demonstrated via the theoretical aspect of the LPF digital filters. The one-dimensional FIR filter is mathematically described as [11]

$$y(k) = \sum_{n=1}^{N} \omega(n) x(k-n+1) \quad k = 1, 2, ..., N+L-1$$
(3)

where y(k) is the filter output, x(m) is the filter's input signal, m = 1, 2, ..., L, and  $\omega(n)$  is the filter's impulse response.

For LPF digital filters, the following four cases are considered.

Case 1, 2: Even filter order (+ for even and - for odd symmetry):

$$y(k) = \sum_{n=1}^{\frac{N}{2}} \omega(n) [x (k - n + 1) \pm x(k - N + n)]$$
  

$$\omega(n) = \pm \omega(N + 1 - n)$$
(4)

Case 3, 4: Odd filter order (+ for even and - for odd symmetry):

A/ 1

$$y(k) = \sum_{\substack{n=1\\ n=1}}^{N-1} \omega(n) \left[ x \left( k - n + 1 \right) \pm x \left( k - N + n \right) \right] + u(k)$$

$$\omega(n) = \pm \omega(N + 1 - n)$$

$$u(k) = \begin{cases} \omega \left( \frac{N+1}{2} \right) x \left( k - \frac{N-1}{2} \right) & \text{for even symmetry} \\ \text{for odd symmetry} \end{cases}$$
(5)

Because of the similarities between the first two and last two cases, case 1 and case 2 are combined into one class, and case 3 and case 4 are combined into another class, with the respective reduced filter orders of N/2, N/2, (N + 1)/2, and (N - 1)/2 for SA implementation [16].

A summary of the theoretical aspect for the remaining DSP algorithms is listed in Appendix A.<sup>2</sup>

## 2.2 System Aspect

A fundamental problem in the design of SA structures is to obtain localized algorithms, namely, localization of data dependencies. To achieve maximum parallelism in an algorithm, data dependencies during the computations must be determined. The following notations can be useful for the system aspect of SA design:

**Definition 1**: A dependency graph shows the dependency of the computations that occur in an algorithm. An algorithm is computable if, and only if, its dependency graph contains no loops or cycles.

**Definition 2**: A localized dependency graph has only local dependencies, i.e., the length of each dependency arc is independent of the problem size.

**Definition 3**: A computational graph is a localized dependency graph with each node in the graph labeled by the indices of the terms it computes.

For SA architecture, determining the computational graph for a given algorithm is a suitable starting point for non-structurally altered arrays. For example, spiral SAs have been structurally altered to conform to an architecture capable of processing blocks of input signals, rather than sequential input signals. The computational graph for the one-step adaptive lattice predictor is shown in Fig. 2. The large circles represent processing elements (PEs), and the first and second digits of the small circles represent one of nine operations, shown in Fig. 3, and the PE number, respectively.

<sup>&</sup>lt;sup>2</sup> This list is presented to maintain continuity with the system and performance design aspects of the DSP algorithms.



Fig. 2. Computational graph for  $\varepsilon_{\ell,k}$  and  $\varepsilon_{b,k}$ .

The computational scheme illustrates identical sets of operations in various stages. This enables us to determine the interconnection scheme which reveals the PE organization and PEs' input and output. The computational and interconnection scheme for the one-step adaptive lattice predictor are illustrated in Figs. 3 and 4, and the internal operations of a PE are shown in Fig. 5 [17]. The system aspect of the SA design for the **DSP algorithms** is listed in Appendix B [18-19].



Fig. 3. Computational scheme for  $\varepsilon_{t,k}$  and  $\varepsilon_{b,k}$ .



Fig. 4. Interconnection scheme for  $\varepsilon_{t,k}$  and  $\varepsilon_{b,k}$ .



Fig. 5. Proposed PE<sub>k</sub> for  $\varepsilon_{t,k}$  and  $\varepsilon_{b,k}$ .

#### 2.2.1 Spiral Architecture

To illustrate this method, we describe herein the process involved in designing a spiral SA for an LPF digital filter: An input signal x(m) is passed from the input  $x_{a}$ of a PE at (i, j) to its output  $x_{aut}$  (see equations 4 and 5). Using zero or one delay, this input is transferred to the input  $x_{in}$  of the neighboring cell  $(i + 1, j^*)$ . The index  $j^*$  can be identified as j + 1 or j - 1 using an analogy from optics. Input data x(k) moves between the PEs in the direction  $d_1 = 5$  or  $d_2 = 3$  until it reaches the left or right boundary of the array, respectively (see Fig. 6(a) and (b)). Then the input is "reflected" back in the respective direction  $d_1 = 3$  or  $d_2 = 5$ , with the intermediate direction  $d_3 = 4$ . The intercell propagation of intermediate values y(k), of input data y(k) can be described by

$$y(k)_{i} = y(k)_{i-1} + \omega(i) \cdot x(k-i+1)$$
(6)

where,  $y(k)_{o} = 0$  and  $y(k)_{i} = y(k)$ . A delay element is positioned immediately following a PE at (i, j), provided that

$$j = \begin{cases} (i+1)_{\text{mod } M} & \text{for } \begin{cases} (r-1) M \leq i < rM & r \text{ odd} \\ i = rM \\ (M-i)_{\text{mod } M} & \text{for } \end{cases} \begin{cases} (r-1) M \leq i < rM & r \text{ odd} \\ i = rM \end{cases}$$
(7)

where, M is the number of input data blocks.



(a) Directional encoded values

(b) Data propagation between array PEs

## Fig. 6. Propagation of data between array PEs.

The two systolic array architectures with spiral structure of intercell connections, proposed to compute equation (4) for class 1 and equation (5) for class 2 (for N = 12) are depicted in Fig. 7(a) and (b), respectively [16]. The secondary input

signal x(k - N + n) is shown integrated into the structure. In this structure, the PE performs the following operation, as illustrated in Fig. 8:

$$\begin{aligned} x_{out} &= x_{in} \\ y_{out} &= y_{in} + \omega(i) \left[ x_{in} \pm x'_{in} \right] \end{aligned} \tag{8}$$

Here x(k - N + n) is the past input data, symmetric to x(k - n + 1), and N - 2i + 1 sampling periods apart.



#### (a) Class 1

(b) Class 2

Fig. 7. Spiral systolic array architecture for LPF filter of order 12. (Notations: + for even symmetry and - for odd symmetry.)



Fig. 8. Inputs and outputs of a PE.

Let U, the reduced filter order, and M, the number of input data blocks, designate the spiral SA's number of rows and columns, respectively. Denote indx, (i, j, t) and indx<sub>o</sub> (i, j, t) to be the respective indices of input and output signals of a PE, where i, j, and t are the row, column and block kernel cycle. Here i = 1, ..., U, j = 1, ..., M, and  $t = 1, ..., T_{max} (T_{max} = (K/M + .5)_{integer})$ , where K is the number of input signals. Note that the indices of input and output signals are related by the following:

$$\operatorname{indx}_{o}(i, j, t) = \operatorname{indx}_{i}(i, j, t) - M$$
(9)

The following algorithm demonstrates the positioning of the input signals in the PEs of the spiral SA, at kernel cycle t.

Algorithm 1. { \* Input: spiral SA's row and column size, and input data size **Output:** input signals' indices,  $indx_i$  (*i*, *j*, *t*). \* }

#### Begin

 $T_{\text{max}}$ : =  $(K/M + 0.5)_{\text{integer}}$ for t = 1 to  $T_{max}$  do begin  $R:=\begin{cases} M/2, & \text{if } M \text{ even} \\ (M-1)/2, & \text{if } M \text{ odd} \end{cases}$ i = 1;for j: = 1 to M do { 1st row computations } begin  $\operatorname{indx}_{i}[i, j, t] := \begin{cases} (t-1) M + (j+1)/2, & \text{if } j \text{ odd} \\ tM - (j-2)/2, & \text{if } j \text{ even} \end{cases}$ end: for i = 2 to N do { rows 2 to N computations } begin  $p:=i \mod 2;$ for j := 1 to M do  $indx_i[i, j, t] := indx_i[i - 1, j, t] - M - 1;$ for q: = 1 to  $\begin{cases} R-1, & \text{if } i \text{ even} \\ R, & \text{if } i \text{ odd} \end{cases}$ do begin  $j1: = 2q + p - 2; \quad j2: = j1 + 1;$ if j2 < = M then begin mid: = indx, [i, j1, t]; { swap of indices }  $indx_i [i, j1, t]$ : =  $indx_i [i, j2, t]$ ; indx, [i, j2, t] := mid :end:  $\{j2 < = M\}$ end; **end**;  $\{i: = 2, N\}$ end:  $\{t = 1, T_{max}\}$ 

End.

The index of some feedback input,  $\ell$ , is related to the index of its destination PE, k by (see eq. 4 and Fig. 7).

$$\ell = k + \Delta k \ell \Delta k \ell = (k - N + n) - (k - n + 1) = -N + 2n - 1$$
(10)  
$$\ell = k + 2n - N - 1$$

An example for the PE input/output data distribution in the spiral SA, with N/2 = 0 and M = 4 dimensions (at kernel cycle t = 9) is shown in Fig. 9:

PE <u>input</u> output data distr.	j = 1	j = 2	<i>j</i> = 3	<i>j</i> = 4
<i>i</i> – 1	<i>x</i> (33)	x(36)	x(34)	x(35)
	x(29)	x(32)	x(30)	x(31)
i _ 2	x(28)	x(29)	<i>x</i> (31)	x(30)
1 = 2	x(24)	x(25)	x(27)	x(26)
i _ 2	x(24)	x(23)	x(25)	x(26)
7 = 5	x(20)	x(19)	x(21)	x(22)
i – 1	<i>x</i> (19)	<i>x</i> (20)	<i>x</i> (18)	x(21)
/ = 4	x(15)	x(16)	x(14)	x(17)
i 5	<i>x</i> (15)	<i>x</i> (14)	<i>x</i> (16)	x(13)
1 = 5	x(11)	x(10)	x(12)	x(9)
i _ 6	<i>x</i> (10)	x(11)	<i>x</i> (9)	x(8)
1 = 0	x(6)	x(7)	$\overline{x(5)}$	$\overline{x(4)}$

Fig. 9. PE input/output data distribution for kernel cycle t = 9.

The connection matrices for the feedback input data of the class 1 and class 2 spiral SA (LPF of order 12) are shown here. (Note that  $\frac{N}{2} = 6$ , M = 4 for class 1 and  $\frac{(N-1)}{2} = 5$ , M = 4 for class 2.)

class 1: 
$$\begin{array}{c} \pm 0 (3,4) & \pm 0 (2,2) & \pm i (3,2) & \pm 0 (2,1) \\ \pm 0 (3,2) & \pm 0 (3,1) & \pm 0 (3,4) & \pm 0 (3,3) \\ \pm 0 (4,4) & \pm 0 (4,2) & \pm i (4,3) & \pm 0 (3,2) \\ \pm 0 (4,3) & \pm 0 (4,1) & \pm i (5,4) & \pm 0 (4,2) \\ \pm 0 (5,3) & \pm 0 (5,1) & \pm i (5,4) & \pm 0 (5,2) \\ \pm 0 (5,4) & \pm 0 (5,2) & \pm i (6,4) & \pm 0 (6,2) \end{array}$$
(11)

class 2: 
$$\begin{bmatrix} \pm i (3,2) & \pm 0 (2,4) & \pm 0 (2,1) & \pm 0 (2,2) \\ \pm 0 (3,1) & \pm 0 (3,3) & \pm i (3,2) & \pm 0 (3,4) \\ \pm i (4,3) & \pm 0 (4,4) & \pm 0 (3,2) & \pm 0 (3,1) \\ \pm 0 (4,1) & \pm 0 (4,2) & \pm 0 (4,3) & \pm 0 (4,4) \\ \pm i (5,4) & \pm 0 (5,3) & \pm i (5,2) & \pm 0 (5,1) \end{bmatrix}$$
(12)

where 0(i, j) and l(i, j) at the (m, n) location of the connection matrix refer to the connection made from the respective output and input of the PE at location (i, j) to the input of the PE that makes up the processor's feedback input data at the location (m, n) of the spiral SA.

The spiral SA structure for computing eqs. A4 and A5 for the DFT and IDFT algorithms and for computing eqs. A2 and A3 for the convolution and correlation algorithms are illustrated in Appendix C [20]. Note that contrary to the LPF spiral SA structure neither structure requires a feedback input, and in addition, the DFT and IDFT spiral SA structures consist of no delay elements.

#### 2.2.2 Asynchronous Architecture

The performance of a systolic array can be improved further if an asynchronous approach is adopted. The idea is to design self-timed processors and communication protocols to gain control of data streams, so that each computation can start if all its data are available. The proposed array is a hybrid of a systolic array and a data-flow machine.

In the asynchronous design, instead of using a global clock, self-timing PEs and a communication protocol are employed. The advantage is the following: The whole period of a clock unit for addition/subtraction, multiplication, addition, and data routing can be separated into several small steps, and some of these steps can be executed simultaneously. The concept of asynchronous computations can be specified as shown below, with steps 2 and 3 executed simultaneously.

Step 1

- send/receive the respective acknowledge signal and data to the previous processor
- send a request signal that simultaneously forwards data to the next processor

Step 2

transfer data to the next processor

Step 3

add/subtract, multiply, and accumulate the results

The basic features of the proposed array remain the same as in the systolic array, with the exception that the data routing and computing in each PE can be operated simultaneously. Moreover, a PE does not wait for data until the previous PE completes its computations. The following algorithm reflects this new feature for an LPF digital filter [16].

#### Algorithm 2: Linear Phase FIR Digital Filter (Asynchronous)

Begin

While there are data entering PE, do

Begin

•**Receive** input data  $x_{in}$ ,  $x'_{in}$ ,  $y_{in}$ ;

**Send**  $x_i$  to the next processing element :

$$\boldsymbol{u}_{i,j} \leftarrow \omega_i \, \boldsymbol{v}_{i,j};$$

$$\mathbf{y}_{out} \leftarrow \mathbf{u}_{i,j} + \mathbf{y}_{in};$$

**End While** 

End

## Protocol realization for Algorithm 2

A protocol for realization of Algorithm 2 must control the flow of data and make the data flow independent of the internal operations in each PE, such that the values of input variables are not overwritten during their computing time intervals. Because in spiral SAs timing is crucial for the synchronization of the PEs' input data, and in asynchronous machines operations are triggered by the availability of the data, the system protocol for Algorithm 2 (1) triggers PEs' operations by the fullness of their  $x_{in}, x'_{in}$ , and  $y_{in}$  input ports, and (2) allows data routing and computing to be performed simultaneously. In the proposed protocol for an LPF digital filter, five types of signals — REQ, ACK, FLG, EMP, FUL — are introduced, two of which are external signals and three of which are internal signals. The REQ signal reports to the next PE that the data in its output port are ready for transmission. The ACK signal reports to the previous PE that its input port is ready to receive new data. The FLG, EMP, and FUL signals indicate the completeness of the add/subtract-multiply-and-add computations, and the emptiness and fullness of the input port(s), respectively. The protocol can be described formally for data  $x_{in}$  as shown below, while noting that the specifications of the protocols for  $x'_{in}$  and  $y_{in}$  are similar.

1. PE<sub>*i*, *j*</sub> receives a request, REQ<sup>xin</sup> from a source PE when the data in its output port are ready to be transmitted.

- 2.  $PE_{i,j}$  receives an acknowledge ACK<sup>xin</sup> from a destined PE when its input port is ready to accept new data.
- 3.  $PE_{i,j}$  has three internal signals,  $FLG^{x_{in}}$ ,  $EMP^{x_{in}}$ , and  $FUL^{x_{in}}$ , which indicate the completeness of the add/subtract-multiply-and-add operations, and the emptiness and fullness of the input port(s).
- 4. **PE**<sub>*i*,*i*</sub> contains the following transition latches:

Latch 1 activates the signal ACK<sup>xin</sup>, which is fired toward a source PE only after both signals REQ<sup>xin</sup> and EMP<sup>xin</sup> are received.

Latch 2 controls the data flow from the input port to the buffer-for-add/subtract and to the output port, and it also activates the signals EMP<sup>x</sup><sup>in</sup> and FUL<sup>x</sup><sup>in</sup>. It is fired only after the signal FLG<sup>x</sup><sup>in</sup> is received.

Latch 3 controls the data flow from the output port of the PE to the input port of a destined PE. It is fired only after the signal ACK<sup> $x_m$ </sup> is received from the destined PE.

A detailed configuration of this protocol for type I cell (case 1 or 3 FIR linear phase digital filters) is depicted in Figure 10. In this configuration, there are two lines that intersect each PE. One is a bidirectional control line for the transmission of the ACK and REQ bit-signals, and the other is a unidirectional data line. The linear phase FIR digital filter operations in PE, are shown in Algorithm 3 [16].

The PE protocols for some of the most important **DSP algorithms** are shown in Appendix D [17-19].

Algorithm 3: Linear Phase FIR Digital Filte	r Operation
(A Completed Asynchronous Versi	on)

Begin				
While there are data ente	ring PE <sub>i, j</sub> do			
Begin				
For all PE <sub>i. j</sub> s asynchronously <b>do</b>				
Begin	Begin			
Wait for R	Wait for REQ <sub>input</sub> from a source PE ;			
<b>Receive</b> REQ <sub>input</sub> ;				
While input port is empty do				
Begin				
Activa	te REQ <sub>input</sub> & EMP ;			
Send	ACK <sub>input</sub> ;			
Recei	ve data x <sub>in</sub> , x' <sub>in</sub> , y <sub>in</sub> ;			
Store	data into the input port ;			
Activa	ite FUL signal ;			
& Send	REQ <sub>output</sub> to a destined PE ;			
Wait f	or ACK $_{output}$ from the destined PE ;			
Send	<i>x<sub>in</sub></i> to output port & buffer-for-add :			
& Add (	$\mathbf{v}_{i,j} \leftarrow \mathbf{x}_{in} + \mathbf{x}'_{in}$ ):			
Store	<i>v<sub>i, j</sub></i> into buffer-for-mult :			
Read	$\omega_i$ from buffer-for-mult ;			
&	$v_{i,j}$ from buffer-for-mult ;			
Mult (	$\boldsymbol{u}_{i,j} \leftarrow \omega_i  \boldsymbol{v}_{i,j}$ :			
Store	u <sub>i.j</sub> into buffer-for-add ;			
Read	<i>u<sub>i.j</sub></i> from buffer-for-add :			
&	y <sub>in</sub> from buffer-for-add :			
Add (	$\mathbf{y}_{out} \leftarrow \mathbf{u}_{i,j} + \mathbf{y}_{in}$ :			
Store	y <sub>out</sub> into buffer-for-W :			
& Activa	ate FLG signal :			
End While	•			
End For				
End While				
End				



Fig. 10. Proposed protocol in (type I)  $PE_{i,j}$ .

#### 2.3 Performance Aspect

The execution time modeling of an asynchronous system can represent quantitative evaluation of the developed algorithms. We developed three performance measures, throughput, spaced up, and efficiency, for evaluating the asynchronous spiral SA realizations. The performance analysis technique is next demonstrated for the LPF digital filter asynchronous SA realization [16]. The following notations will be used to describe the time models of the arrays:

- $T_{in}$  = time for reading data from input port
- $T_{D_i}$  = time for internal data transfer
- $T_{D_{o}}$  = time for external data transfer
- $T_R$  = signal REQ propagation time
- $T_{A_k}$  = signal ACK propagation time
- $T_{Fire}$  = time for firing data via latch
- $T_c$  = clock time for synchronous systolic array realization
- $T_{\varepsilon}$  = signal EMP propagation time
- $T_{A_{d}}$  = computing time for addition
- $T_{M}$  = computing time for multiplication
- $T_{\rm w}$  = time for fetching data from buffer-for-add/multiply/etc.
- $T_F$  = signal FLG/FUL propagation time
- $T_{out}$  = time for reading data from output port

The time delays, starting with the reading of data from input port, for the respective asynchronous spiral SA and the corresponding synchronous systolic array can be described by the following relationships (see Figure 10):

$$T_{\mathsf{PE}_{(asyn)}} = \max\{(T_{In} + T_{D_i} + (T_R + T_{Fire}) + (T_{A_k} + T_{Fire}) + T_{Out} + T_{D_2}).$$

$$((T_{In} + T_{D_i} + T_W + T_{A_d} + T_W + T_M + T_W + T_{A_d} + T_W) + (T_R + T_{Fire}) + (13)$$

$$(T_{A_k} + T_{Fire}) + T_{Out} + T_{D_2}\}$$

$$T_{total_{(asyn)}} = (U) (T_{ln} + T_{D_l} + T_{Out} + T_{D_o} + 4T_W + 2T_{A_d} + T_M + T_{Com})$$
(14)

where,

$$T_{Com} = T_{A_k} + 2T_{Fire} + T_R$$

and,

$$T_{total_{(syn)}} = M'(T_{ln} + T_{D_i} + T_{D_o} + 3 (\max\{T_{A_d}, T_M\} + T_W)) = M'(T_{ln} + T_{D_i} + T_{D_o} + 3 (T_M + T_W)) = M'T_c$$
(15)

where, M' refers to the number of PEs in the synchronous systolic array, and

$$M' = \begin{cases} M+1 & \text{if } N \text{ odd, even symmetry} \\ M & \text{otherwise} \end{cases}$$

The throughput can be obtained by dividing the number of processors by the execution time per sample. Thus, the throughput using  $M \times U$  processors,  $R(M_0)$  is defined by the following equation:

$$R(M_n) = \frac{M}{T_{ln} + T_{D_l} + T_{Out} + T_{D_o} + 4T_W + 2T_{A_d} + T_M + T_{Com}}$$
(16)

The speedup is the ratio of the throughput in the asynchronous spiral SA system to that in a reference synchronous system. From above, the speedup via an asynchronous spiral SA realization is represented as

$$S(M_n) = \frac{M(T_{ln} + T_{D_i} + T_{D_o} + 3(T_M + T_W))}{M'(T_{ln} + T_{D_i} + T_{Out} + T_{D_o} + 4T_W + 2T_{A_q} + T_M + T_{Com})}$$
(17)

The system efficiency is defined as the ratio of the speedup and the number of processors

$$E(M_n) = \frac{T_{ln} + T_{D_i} + T_{D_o} + 3(T_M + T_W)}{U(M')(T_{ln} + T_{D_i} + T_{Out} + T_{D_o} + 4T_W + 2T_{A_d} + T_M + T_{Com})}$$
(18)

The above equation indicates that the reduction of interprocessor communication time,  $T_{Com}$ , greatly improves the system's efficiency [16].

#### 2.4 References

- [1] Kung, H. T., and C. E. Leiserson, "Systolic Arrays (for VLSI)." In Sparse Matrix Proc.; Duff et al., Eds., 1979, pp. 256-282.
- [2] Kung, H. T., and C. E. Leiserson, "Algorithms for VLSI Processor Arrays." In Introduction to VLSI System; C. Mead and L. Conway, Eds. Reading, MA: Addison-Wesley, 1980, pp. 263-332.
- [3] Melkemi, L., and M. Tchuente, "Algebraic and Combinatorial Aspects of Systolic Algorithms." In Proc. of Int. Workshop on Parallel Algorithms and Arch., 1986, pp. 269-279.
- [4] Cytron, R. G., Compile-Time Scheduling and Optimization for Asynchronous Machines. Ph.D. Dissertation, University of Illinois at Urbana-Champaign. 1985.
- [5] Peng, S., and M. Jun, "A New VLSI 2-D Systolic Array for Matrix Multiplication and Its Applications." In Proc. of Int. Conf. on Parallel Processing, 1988, pp 169-172.
- [6] Kung, H. T. "Why Systolic Architectures?" Computer, Jan. 1982, Vol. 15, No. 1, pp. 37-46.

- [7] Kung, S. Y., and S. C. Lo, "Supercomputing with Systolic/Wavefront Array Processors." In *IEEE Proc.*, July 1984, Vol. 72, No. 7.
- [8] Mills, P. L., "The Design of Bit Parallel Systolic Filter Algorithms." In Proc. ACM Computer Science Conference, 1984, pp. 121-130.
- [9] Cooley, J. W., et al., "The Fast Fourier Transformation Algorithm: Programming Considerations in the Calculation of Sine, Cosine, and Laplace Transformations." Digital Signal Processing; L. R. Rabiner and C. M. Rader, Eds. New York, NY: IEEE Press, 1972, pp. 271-293.
- [10] Oppenheim, A. V., and R. W. Schafer, *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.
- [11] Kung, S. Y., VLSI Array Processors. Englewood Cliffs, NJ: Prentice-Hall, 1988.
- [12] Kaminski, P. G., A. E. Bryson, and S. F. Schmidt, "Discrete Square-Root Filtering: A Survey of Current Techniques." In IEEE Trans. on Automatic Control, Dec. 1971, Vol. AC-16, No. 6, pp. 727-735.
- [13] Griffiths, L. J., "A Continuously-Adaptive Filter Implemented as a Lattice Structure." In Proc. ICASSP-77, May 1977, pp. 683-686.
- [14] Haykin, S., Adaptive Filter Theory. Second Edition, Prentice-Hall. 1991
- [15] Widrow, B., and S. Sterns, Adaptive Signal Processing. Prentice-Hall, 1985.
- [16] Reihani, K, and Y. Fan, "Spiral Systolic Design with Asynchronous Controls for LPF Digital Filters." Under review by the IEEE Transactions on Signal Processing.
- [17] Reihani, K., and M. Mayorga, "Asynchronous Systolic Realization for One-Step Adaptive Lattice Predictors." To appear in the Proc. of the Fourth Int. Conf. on Signal Processing Applications and Technology - ICSPAT '93, Santa Clara, CA, Sept. 1993.
- [18] Reihani, K., and M. Mayorga, "Asynchronous Systolic Realization for Arbitrarily Large LMS Adaptive Filters." Under Review by the *IEEE Transactions on Computers*.
- [19] Reihani, K., and Z. Zhang, "Asynchronous and Concurrent Systolic Array Design for Discrete Hadamard Transforms." PSL Internal Report, June 1992.
- [20] Reihani, K., and Y. Fan, "Spiral Systolic Design with Asynchronous Controls for DFT and IDFT." PSL Internal Report, May 1993.

## 3. PUBLICATIONS AND REPORTS

- [1] Reihani, K, and Y. Fan, "Spiral Systolic Design with Asynchronous Controls for LPF Digital Filters." Under review by the *IEEE Transactions on Signal Processing*.
- [2] Reihani, K., and M. Mayorga, "Asynchronous Systolic Realization for One-Step Adaptive Lattice Predictors." To appear in the Proc. of the Fourth Int. Conf. on Signal Processing Applications and Technology - ICSPAT '93, Santa Clara. CA, Sept. 1993.
- [3] Reihani, K., and M. Mayorga. "Asynchronous Systolic Realization for Arbitrarily Large LMS Adaptive Filters." Under Review by the *IEEE Transactions on Computers*.
- [4] Reihani, K., and Z. Zhang, "Asynchronous and Concurrent Systolic Array Design for Discrete Hadamard Transforms." PSL Internal Report, June 1992.
- [5] Reihani, K., and Y. Fan, "Spiral Systolic Design with Asynchronous Controls for DFT and IDFT." PSL Internal Report, May 1993.

## 4. PERSONNEL SUPPORTED AND DEGREES AWARDED

Major Investigator	Research Area
Reihani, K.	Signal Processing and Image Understanding
Doctoral Students	Dissertation
Fan, Yiping	Current Ph.D. Student: Signal Processing
Zhang, Zhi	Current Ph.D. Student: Signal Processing

**APPENDIX A** 

#### THEORETICAL ASPECT FOR DSP ALGORITHMS

Arbitrarily Large LMS Adaptive Filters

A standard form for the LMS algorithm when real discrete time signals,  $x_p$  are used is described by

$$W_{i+1} = W_i + 2 \mu \xi_i X_i$$
 (A1)

where  $W_j$  and  $X_j$  are vectors representing, respectively, the filter weights and the outputs of the transversal filter line at time j

$$W_{j}^{T} = [w_{1,j}, w_{2,j}, ..., W_{n,j}]$$
$$X_{j}^{T} = [x_{j}, x_{j-1}, ..., x_{j-n+1}]$$

and  $\mu$  is the gain constant that regulates the speed and stability of adaptation. The error,  $\xi_i$ , is the difference between the desired response  $d_i$ , an externally supplied training signal, and the filter output  $y_i$ 

 $\xi_j = d_j - y_j$ 

where the filter output is given by

 $y_j = X_j^T W_j$ 

- Convolution and Correlation

Convolution and correlation are closely related operations that are basic to many areas of digital signal processing. The convolution of two time series is equivalent to the product of the Fourier transforms of the two time series. They can be expressed for a finite length N as

Convolution: 
$$c_{xy}(n) = conv\{[x(k), y(k)]\} = \sum_{k=0}^{N-1} x(k)y(n-k) = \sum_{k=0}^{N-1} x(n-k)y(k)$$
 (A2)

Correlation: 
$$r_{xy}(n) = corr\{[x(k), y(k)]\} = \sum_{k=0}^{N-1} x(k)y(n+k) = \sum_{k=0}^{N-1} x(n+k)y(k)$$
 (A3)

- DFT and IDFT

The discrete Fourier transform (DFT) is used to transform an ordered sequence of data samples, usually from the time domain into the frequency domain, so that spectral information about the sequence can become known explicitly. The inverse discrete Fourier Transform (IDFT) is used to obtain a data sequence in time domain from its complex spectrum. If we denote x(k) (k = 0, 1, ..., N - 1) as a time series and X(k) as the output of an N point Fourier transform, then

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi n}{N} k} = \sum_{n=0}^{N-1} x(n) w(nk) \quad k = 0, 1, ..., N-1$$
 (A4)

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j \frac{2\pi n}{N} k} = \frac{1}{N} \sum_{k=0}^{N-1} X(k) w(-nk) \quad n = 0, 1, ..., N-1 \quad (A5)$$

where

$$w(nk) = e^{-j \frac{2\pi n}{N} k}$$
$$w(-nk) = e^{j \frac{2\pi n}{N} k}$$

- DHT

A Hadamard matrix is a matrix whose columns are orthogonal and comprised of elements whose value is either -1 or +1. Given a Hadamard matrix of order N. Hadamard matrices of higher order may be generated by the simple recursive formulation.

$$H_{2N} = \begin{bmatrix} -\frac{H_N}{-} & \frac{H_N}{-} \\ -\frac{H_N}{-} & -\frac{H_N}{-} \end{bmatrix}$$
(A6)

The Hadamard matrix of the lowest order becomes

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$
(A7)

and the Hadamard matrix of order N = 4 is given as

**APPENDIX B** 

# SYSTEM ASPECT I(A)

- Arbitrarily Large LMS Adaptive Filters

- Computational graph for  $y_i$ 



— Computational scheme for  $y_i$ 



- Interconnection scheme for  $y_i$ 



- Proposed  $PE_k$  for  $y_j$ 



## - DHT

- Systolic array scheme for a 4 x 4 DHT



APPENDIX C



DFT

IDFT

--- Spiral SA Structure to Compute Equations A2 and A3 (N = 6)

- Convolution and correlation





Convolution

Correlation

2

## SYSTEM ASPECT II

- One-Step Adaptive Lattice Predictors' PE<sub>m</sub> Protocol





- Arbitrarily Large LMS Adaptive Filters' PE, Protocol



