BULETINUL INSTITUTULUI POLITEHNIC DIN IAŞI
Publicat de
Universitatea Tehnică „Gheorghe Asachi” din Iaşi
Tomul LVIII (LXII), Fasc. 4, 2012
Secția
ELECTROTEHNICĂ. ENERGETICĂ. ELECTRONICĂ

A NEW SYSTOLIC ALGORITHM FOR 1-D DST USING PARALLEL PSEUDO-CIRCULAR CORRELATIONS

BY

DORU-FLORIN CHIPER*

"Gheorghe Asachi" Technical University of Iaşi,
Faculty of Electronics, Telecommunications
and Information Technology

Received, December 12, 2012
Accepted for publication: December 27, 2012

Abstract. Using a new restructuring input sequence and appropriate index mappings we have reformulated 1-D discrete sine transform (DST) in such a way that a new efficient VLSI algorithm for a prime-length DST is obtained. The proposed algorithm uses some modular and regular computational structures called *pseudo-circular correlations* that can be computed in parallel, thus resulting a high throughput and efficient VLSI implementation. The proposed algorithm can be mapped into a linear systolic array with high performances. High computing speed with low I/O cost characterized by a small number of I/O channels placed at the two ends of the linear array have been obtained. The resulting VLSI architecture is highly regular and modular with local connections and have a small hardware complexity of the pre-processing and post-processing stages.

Key words: discrete transforms; DST; systolic arrays; VLSI architectures.

1. Introduction

The discrete sine transform (DST) together with discrete cosine transform (DCT) (Ahmed *et al.*, 1974; Jain, 1976, 1980) are important

* *e-mail*: chiper@etc.tuiasi.ro

transforms used in many digital signal processing applications. Being good approximations to the statistically-optimal Karhunen-Loeve transform, they are used especially in speech and image transform coding (Jain, 1980; Zhang *et al*., 2007), but also in other transform coding applications as DCT based subband decomposition in speech and image compression (Chen, 2007), or video transcoding (Fung & Siu, 2006). Some other important applications are: block filtering (Martucci & Mersereau, 1993), transform-adaptive filtering (Pei & Tseng, 1996; Mayyas, 2005) and filter banks (Bergen, 2008).

The DST represents a good approximation of the statistically optimal Karhunen-Loeve transform and for low correlated images generates better results as compared with DCT. Although the usual transform length in block transform coding is 8 or 16 we can use a prime transform length of 11 or 17 and an overlapping technique in order to reduce blocks artifacts. Also , it is useful to have a prime DST length or a composite transform length were the factors are mutually prime. A prime factor could be a more suitable transform length for some specific applications than a power of two (Tasaki *et al*., 1995) and there are in the literature efficient algorithms for an efficient implementation of a prime-factor DST length (Chiper *et al*., 2002). Also, there is the possibility to combine prime-factor algorithms for an efficient implementation of the DST transform for composite-lengths (Kar & Rao, 1994).

Due to the fact that DST is computational intensive it is necessary to design new algorithms or to restructure the existing ones in order to accelerate the execution of this transform through VLSI implementation. The efficiency of a VLSI implementation is greatly due to the data movement and transfer existing in the algorithm. This has been already proved by several implementation solutions that have been proposed for a VLSI implementation using cyclic convolution (Chiper *et al*., 2005) or circular correlation (Chiper *et al*., 2002). The circular correlation is well suited for a VLSI implementation using distributed arithmetic (White, 1989) or systolic arrays (Kung, 1982) due to the fact that they avoid complicated data routing and management leading thus to efficient VLSI implementation with a regular and modular structure.

Systolic arrays can exploit the advantages offered by the VLSI technology, especially through modularity, regularity and short and local interconnections representing thus an interesting architectural paradigm for the VLSI implementation of our proposed VLSI algorithm. The I/O bottleneck has been reduced by repeatedly using of the input data in the systolic array. In the same time systolic arrays are well suited for real time implementations due to the fact that they can efficiently exploit the concurrency existing in the VLSI algorithm. All the above mentioned advantages of the circular correlation can be extended to *pseudo-circular correlation*, where the differences in sign are managed using the control-tag scheme.

In this paper we introduce a new restructuring sequence that is used to appropriately reformulate 1-D DST into an efficient VLSI algorithm. The new systolic array algorithm is based on two regular and modular computational

structures called *pseudo-circular correlations*, that can be computed in parallel resulting thus a high throughput VLSI implementation. We have a single input restructuring sequence as opposed to a previous paper (Chiper *et al.*, 2005), where two such auxiliary sequences are used. This results into a significantly reduction of the chip area occupied by the pre-processing stage that for relatively short DST transforms represents an important overhead. The differences in sign involved by the pseudo-circular correlations can be efficiently managed using the tag control scheme. The resulting VLSI architecture that can be obtained with the proposed VLSI algorithm has some important advantages as regularity, modularity, low I/O cost and a reduced data management scheme.

The rest of the paper is organized as follows: in Section **2** a low complexity reformulation of the DST transform based on pseudo circular correlation structures is presented. În Section **3** it is presented an example of the new VLSI algorithm for 1-D DST of length N = 11. In Section **4** we discuss some implementation aspects of the proposed VLSI algorithm using the systolic array architectural paradigm. Conclusions are presented in Section **5**.

2. Systolic Algorithm for 1-D DST

1-D DST is defined as follows:

$$Y(k) = \sqrt{\frac{2}{N}} \sum_{i=0}^{N-1} x(i) \sin[(2i+1)k\alpha], \ (k = 1,\ldots,N), \tag{1}$$

with

$$\alpha = \frac{\pi}{2N}, \tag{2}$$

where the input sequence $\{x(i),\ (i = 0,1,...,N-1)\}$ is a real input sequence.

In order to simplify the presentation, the constant $\sqrt{2/N}$ from the eq. (1) can be dropped and a multiplier will be added at the end of the VLSI systolic array to scale the output sequence with this constant.

As opposed to a previous paper (Chiper *et al.*, 2005), where have been used two input restructuring sequences to reformulate relation (1) as a parallel decomposition, we will introduce a single new input restructuring sequence to do a such decomposition while preserving the regularity and the modularity of the computational structures. Further, we'll use the symmetry of DST kernel and the properties of the Galois Field of indexes to obtain the desired computational structures that have been called pseudo-circular correlations.

The new input restructuring sequence, $\{x_s(i),\ (i = 0,1,...,N-1)\}$, is defined as follows:

$$x_s(N-1) = x(N-1), \tag{3}$$

$$x_s(i) = (-1)^i x(i) + x_s(i+1), \ (i = N-2,...,0). \tag{4}$$

Further on, we can reformulate (1) namely

$$Y(N) = x_s(0), \tag{5}$$

$$Y(k) = x_s(0)\sin(k\alpha) + T(k)\cos(k\alpha), \ (k = 1,\ldots,N-1). \tag{6}$$

We have introduced a new auxiliary output sequence, $\{T(k), (k = 1,2,...,N-1)\}$, that can be computed in parallel as two pseudo-circular correlations, if the transform length, *N*, is a prime number, as following:

$$T(\delta(k)) = \sum_{i=1}^{(N-1)/2} (-1)^{\mu(k,i)} \{x_a\varphi(i) + x_a[\varphi(i+(N-1)/2]\} \times 2\sin(\sigma(k+i)\times 2\alpha), \tag{7}$$

$$T(\gamma(k)) = \sum_{i=1}^{(N-1)/2} (-1)^{\nu(k,i)} \{x_a\varphi(i) - x_a(\varphi(i+(N-1)/2]\} \times 2\sin(\sigma(k+i)\times 2\alpha), \quad (k = 0,1,...,(N-1)/2), \tag{8}$$

where

$$\sigma(k) = \begin{cases} \varphi(k) \text{ if } \varphi(k) \le (N-1)/2; \\ \varphi(N-1+k), \text{ otherwise,} \end{cases} \tag{9}$$

with

$$\varphi(k) = \left\langle g^k \right\rangle_N \text{; here } < x >_N \text{ denotes the result of } x \text{ modulo } N. \tag{10}$$

Relations (9) and (10) define some index mappings based on the properties of the Galois Field of indexes.

3. Example

In the followings we will consider an example for 1-D DST with the length N = 11 and the primitive root g = 2, to illustrate the proposed VLSI algorithm. Firstly, we compute the restructuring input sequence, $\{x_s(i), \ (i = 0,...,N-1)\}$, as following:

$$x_s(10) = x(10), \tag{12}$$

$$x_s(i) = (-1)^i x(i) + x_s(i+1), \ (i = 9,...,0). \tag{13}$$

Using the sequence $\{x_s(i), \ (i = 0,...,N-1)\}$,we can write (7) and (8) in a matrix-vector product form as

$$\begin{bmatrix} T(2) \\ T(4) \\ T(8) \\ T(6) \\ T(10) \end{bmatrix} = \begin{bmatrix} s(4) & s(3) & -s(5) & -s(1) & -s(2) \\ s(3) & -s(5) & s(1) & s(2) & -s(4) \\ -s(5) & s(1) & -s(2) & s(4) & -s(3) \\ -s(1) & s(2) & s(4) & -s(3) & -s(5) \\ -s(2) & -s(4) & -s(3) & -s(5) & -s(1) \end{bmatrix} \begin{bmatrix} x_s(2) + x_s(9) \\ x_s(4) + x_s(7) \\ x_s(8) + x_s(3) \\ x_s(5) + x_s(6) \\ x_s(10) + x_s(1) \end{bmatrix}, \tag{14}$$

$$\begin{bmatrix} T(9) \\ T(7) \\ T(3) \\ T(5) \\ T(1) \end{bmatrix} = \begin{bmatrix} -s(4) & -s(3) & s(5) & -s(1) & s(2) \\ -s(3) & s(5) & -s(1) & s(2) & s(4) \\ s(5) & -s(1) & s(2) & s(4) & s(3) \\ s(1) & -s(2) & -s(4) & -s(3) & s(5) \\ s(2) & s(4) & s(3) & -s(5) & s(1) \end{bmatrix} \begin{bmatrix} x_s(2) - x_s(9) \\ x_s(4) - x_s(7) \\ x_s(8) - x_s(3) \\ x_s(5) - x_s(6) \\ x_s(10) - x_s(1) \end{bmatrix}, \tag{15}$$

where we have noted $s(k)$ for $2\sin(2k\alpha)$.

The index mappings, $\delta(i)$ and $\gamma(i)$, in eqs. (7) and (8), realize a partition into two groups of the permutation of indexes {1,2,3,4,5,6,7,8,9,10}. They are defined as follows:

$$\{\delta(i): 1 \to 2, 2 \to 4, 3 \to 8, 4 \to 6, 5 \to 10\}, \tag{16}$$

$$\{\gamma(i): 1 \to 9, 2 \to 7, 3 \to 3, 4 \to 5, 5 \to 1\}. \tag{17}$$

The functions $\mu(k,i)$ and $\nu(k,i)$ define the sign of terms in eqs. (14) and (15), respectively. They are defined as follows: a) $\mu(k,i)$ is defined by the matrix

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

and b) $v(k,i)$ is defined by the matrix

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Finally, the output sequence, $\{Y(k),\ (k=1,2,...,N-1)\}$, can be computed as follows:

$$\begin{aligned} Y(1) &= x_s(0)\sin[\alpha] + T(1)\cos[\alpha], \\ Y(2) &= x_s(0)\sin[2\alpha] + T(2)\cos[2\alpha], \\ Y(3) &= x_s(0)\sin[3\alpha] + T(3)\cos[3\alpha], \\ Y(4) &= x_s(0)\sin[4\alpha] + T(4)\cos[4\alpha], \\ Y(5) &= x_s(0)\sin[5\alpha] + T(5)\cos[5\alpha], \\ Y(6) &= x_s(0)\sin[6\alpha] + T(6)\cos[6\alpha], \\ Y(7) &= x_s(0)\sin[7\alpha] + T(7)\cos[7\alpha], \\ Y(8) &= x_s(0)\sin[8\alpha] + T(8)\cos[8\alpha], \\ Y(9) &= x_s(0)\sin[9\alpha] + T(9)\cos[9\alpha], \\ Y(10) &= x_s(0)\sin[10\alpha] + T(10)\cos[10\alpha]. \end{aligned} \tag{18}$$

$$Y(11) = x_s(0). \tag{19}$$

4. An Analyse of the VLSI Implementation

Using the proposed algorithm and a data dependence-graph based synthesis procedure we can obtain a linear systolic array as will be seen in the following. First, we can map eqs. (14) and (15) into two linear systolic arrays that represent the hardware-core of the architecture used for the VLSI implementation of the proposed algorithm. In eqs. (14) and (15) there are some differences in sign that can be managed using the tag-control mechanism (Jen & Hsu, 1988) well suited for the systolic array architectural paradigm. The obtained processing elements consists of a multiplier, an adder and some multiplexers used to manage the differences in sign in eqs. (14), (15), respectively. The multipliers for the processing elements located in the same position in the two systolic arrays use the same operand. This aspect can be further used to obtain an important hardware reduction complexity using an appropriate hardware sharing method.

The proposed VLSI architecture can be implemented using a two level pipelining by internally pipelining the adders and multipliers. Thus, the throughput can be significantly increased as compared with the proceeding used in a previous paper (Chiper *et al*., 2005), where such a mechanism can not be incorporated. Using the tag-control mechanism it is possible to place all I/O channels at the two extreme ends of each systolic array as shown by Guo *et al*. (1993), where the tag bits used in this mechanism can control the internal registers using only I/O channels placed at the two ends of the linear systolic array. Due to the fact that each input data is used in each processing element the I/O cost has been significantly reduced. This aspect is very important in designing systolic arrays as the so called *I/O bottleneck*, can seriously limit the speed performances of the design.

The input sequence is computed in the pre-processing stage using eqs. (3) and (4). In the same module we have to appropriately permute this input sequence. First, we have to reverse the order of the input data in order to compute eq. (4) and then we have to change the order of the obtained sequence with the view to obtain the desired auxiliary input sequence in the desired order, as shown in (10). These operations are obtained using a RAM with *N* words. In the paper published by Chiper *et al*. (2005) the number of RAMs and adders is doubled as compared with the proposed solution in this paper.

In the post-processing stage we have to reorder the auxiliary output sequence $\{T(k),\ (k=1,2,...,N-1)\}$ using the index mappings (16) and (17). Then, we will compute the output sequence $Y(k)$ as shown in eqs. (18) using two multipliers. In a previous paper (Chiper *et al*., 2002) there where used four such multipliers to compute the final output sequence.

5. Conclusions

In this paper it has been introduced a new restructuring input sequence that together with appropriate index mappings have been used to reformulate 1-D DST into an efficient VLSI algorithm. The proposed algorithm is based on some regular and modular computational structures called pseudo-circular correlations. These computational structures are computed in parallel resulting a high throughput. The proposed VLSI algorithm can be mapped on a linear systolic array with a high processing speed at a low I/O cost. Moreover, using the proposed VLSI algorithm, the hardware complexity of the pre-processing and post-processing stages can be substantially reduced.

REFERENCES

Ahmed N., Natarajan T., Rao K.R., *Discrete Cosine Transform*. IEEE Trans. Comput., **C-23**, *1*, 90-94 (1974).

Bergen S.W., *A Design Method for Cosine-Modulated Filter Banks Using Weighted Constrained-Least-Squares Filters*. Dig. Signal Process., **18**, *3*, 282-290 (2008).

Chen Y.-Y., *Medical Image Compression Using DCT-Based Subband Decomposition and Modified SPIHT Data Organization.* Internat. J. of Med. Inf., **76**, 717-725 (2007).

Chiper D.F., Swamy M.N.S., Ahmad M.O., Stouraitis T., *Systolic Algorithms and a Memory-Based Design Approach for a Unified Architecture for the Computation of DCT/DST/IDCT/IDST*. IEEE Trans. on Circ. a. Syst._I: Regular papers, **52**, *6*, 1125-1137 (2005).

Chiper D.F., Swamy M.N.S., Ahmad M.O., Stouraits T., *A Systolic Array Architecture for the Discrete Sine Transform.* IEEE Trans. on Signal Process., **50**, *9*, 2347-2354 (2002).

Fung K.-T., Siu W.-C., *On Re-Composition of Motion Compensated Macroblocks for DCT-Based Video Transcoding.* Signal Process.: Image Commun., **21**, 44-58 (2006).

Guo J., Liu C.M., Jen C.W., *A New Array Architecture for Primelength Discrete Cosine Transform*. IEEE Trans. Signal Process., **41**, 436–442 (1993).

Jain A.K., *A Fast Karhunen-Loeve Transform for a Class of Random Processes.* IEEE Trans. Comm., **COM-24**, *10*, 1023-1029 (1976).

Jain A.K., *Fundamentals of Digital Image Processing*. Englewood Cliffs,Prentice-Hall, NJ, 1989.

Jen C.W., Hsu H. Y., *The Design of Systolic Arrays with Tag Input*. Proc. IEEE Internat. Symp. Circ. Syst., Helsinki, Finland, 1988, 2263–2266.

Kar D., Rao V.V.B., *On Prime Factor Decomposition Algorithm for the Discrete Sine Transform*. IEEE Trans. on Signal Process., **42**, *11*, 3258-3260 (1994).

Kung H.T., *Why Systolic Architectures.* IEEE Comp., **15**, 37-46 (1982).

Martucci S.A., Mersereau R., *New Approaches to Block Filtering of Images Using Symmetric Convolution and the DST or DCT*. Proc. IEEE Internat. Symp. Circ. Syst. (ISCAS'93), Chicago, May 1993, 259-262.

Mayyas K., *A Note on Performance Analysis of the DCT-LMS Adaptive Filtering Algorithm.* Signal Process., **85**, 1465-1467 (2005).

Pei S.-C., Tseng C.-C., *Transform-Domain Adaptive Filter*. IEEE Trans. Signal Process., **44**, *12*, 3142-3146 (1996).

Tatsaki A., Dre C., Stouraitis T., Goutis C., *Prime-Factor DCT Algorithms.* IEEE Trans. on Signal Process., **43**, *3*, 772-776 (1995).

White S.A., *Applications of Distributed Arithmetic to Digital Signal Processing: A Tutorial Review.* IEEE ASSP Mag., **6**, *3*, 5-19 (1989).

Zhang D., Lin S., Zhang Y., Yu L., *Complexity Controllable DCT for Real-Time H.264 Encoder*. J. of Visual Commun. a. Image Repr., **18**, 59-67 (2007).

UN NOU ALGORITM SISTOLIC PENTRU TRANSFORMATA 1-D DST FOLOSIND CORELATII CIRCULARE DE TIP PSEUDO

(Rezumat)

Utilizând o nouă secvență de restructurare la intrare şi operații adecvate de permutare a indicilor s-a reformulat transformata 1-D DST într-un astfel de mod încât s-a obținut un nou algoritm VLSI eficient. Algoritmul propus utilizează anumite structuri computaționale regulate şi modulare, numite *corelații circulare*, care pot fi calculate în paralel, rezultând astfel o implementare VLSI, eficientă, cu o mare productivitate în procesarea datelor. Algoritmul propus poate fi mapat pe o arie sistolică liniară având performanțe ridicate.