

BULETINUL INSTITUTULUI POLITEHNIC DIN IAȘI  
Publicat de  
Universitatea Tehnică „Gheorghe Asachi” din Iași  
Tomul LX (LXIV), Fasc. 2, 2014  
Secția  
ELECTROTEHNICĂ. ENERGETICĂ. ELECTRONICĂ

## A NOVEL SYSTOLIC ALGORITHM FOR TYPE IV DST TRANSFORM

BY

**DORU-FLORIN CHIPER\*** and **PAUL UNGUREANU**

“Gheorghe Asachi” Technical University of Iași  
Faculty of Electronics, Telecommunications and Information Technology

Received: May 15, 2014

Accepted for publication: June 26, 2014

**Abstract.** Using a new VLSI algorithm for 1-D type IV discrete sine transform (DST-IV) an efficient VLSI architecture with appealing topological features and high performances can be obtained. The new algorithm has a modular and regular computational structure and can be computed in parallel thus resulting a high throughput VLSI implementation. The proposed algorithm can be mapped into two linear systolic arrays having a high computing speed and low I/O cost with a small number of I/O channels placed at the two ends of the linear array. By combining two such linear systolic arrays we can obtain an efficient VLSI architecture for type IV DST. The architecture that can be obtained has a highly regular and modular structure and local connections specific to the systolic array architectural paradigm.

**Key words:** Type IV discrete sine transform; systolic array; VLSI algorithm; VLSI architecture.

### 1. Introduction

Discrete Sinus Transform (DST) like DCT (Ahmed *et al.*, 1974) expresses a signal as a sum, but of sinus, not of cosines functions. They can be found in digital signal and image processing applications and particularly data compression/ decompression.

---

\*Correspondin author: *e-mail*: chiper@etc.tuiasi.ro

Existence of such transforms was discovered by A. Jain in 1979, and Wang and Hunt defined later a complete set of orthogonal functions (Rao & Hwang, 1996). DCT and DST have two important properties. First these functions constitute an orthogonal basis of real functions and more important is the existence of fast algorithms for their computation. This are the main reasons for choosing this transforms as tools in the JPEG (Joint Photographic Experts Group) and MPEG (Moving Picture Experts Group) 1-4 standards (Rao & Hwang, 1996). Others examples of transforms used in practice are: Karhunen–Loève transform, discrete Fourier transform, Hartley transform (Hwang, 1995).

There are several DCT and DST transforms type. For example, DCT-II was been used for image and video compression standards such as: JPEG, H.26x-series, and MPEG 1-4. Also, the DCT and DST of type IV has been used in design of Lapped Orthogonal Transforms (Malvar, 1992), for implementing filter-banks in speech and audio coding algorithms, such as G.722.1, G.718, MPEG-4 AAC (Chivukula & Reznik, 2008).

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 circular correlation (Chiper *et al.*, 2012) or cyclic convolution (Chiper *et al.*, 2005). 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.

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 propose a new systolic array algorithm for type IV DST using some regular and modular computational structures. These structures can be computed in parallel resulting thus a high throughput VLSI implementation. We have used a new restructuring method of the type IV DST into such regular structures. All the advantages of a circular correlation or cycle convolution based implementation as regularity, modularity, low I/O cost and a reduced data management scheme can be obtained with the proposed computational structures.

The rest of the paper is organized as follows:

in Section 2 a low complexity formulation is presented for the computation of the type IV DST transform with an example for a DST IV of length  $N = 11$ . In Section 3 we discuss some details about a VLSI implementation of the proposed algorithm using the systolic array architectural paradigm. Conclusions are presented in Section 4.

## 2. Systolic Algorithm for 1-D Type IV DST

The output sequence  $\{Y(k)k=1,2,\dots,N-1\}$  of type IV DST can be computed as follows:

$$\begin{bmatrix} Y(0) \\ Y(1) \\ Y(2) \\ Y(3) \\ Y(4) \\ Y(5) \\ Y(6) \\ Y(7) \\ Y(8) \\ Y(9) \\ Y(10) \end{bmatrix} = \begin{bmatrix} \sin(\alpha) & \sin(3\alpha) & \sin(5\alpha) & \dots & \sin(19\alpha) & \sin(21\alpha) \\ \sin(3\alpha) & \sin(9\alpha) & \sin(15\alpha) & \dots & \sin(57\alpha) & \sin(63\alpha) \\ \sin(5\alpha) & \sin(15\alpha) & \sin(25\alpha) & \dots & \sin(95\alpha) & \sin(105\alpha) \\ \sin(7\alpha) & \sin(21\alpha) & \sin(35\alpha) & \dots & \sin(133\alpha) & \sin(147\alpha) \\ \sin(9\alpha) & \sin(27\alpha) & \sin(45\alpha) & \dots & \sin(171\alpha) & \sin(189\alpha) \\ \sin(11\alpha) & \sin(33\alpha) & \sin(55\alpha) & \dots & \sin(209\alpha) & \sin(231\alpha) \\ \sin(13\alpha) & \sin(39\alpha) & \sin(65\alpha) & \dots & \sin(247\alpha) & \sin(273\alpha) \\ \sin(15\alpha) & \sin(45\alpha) & \sin(75\alpha) & \dots & \sin(285\alpha) & \sin(315\alpha) \\ \sin(17\alpha) & \sin(51\alpha) & \sin(85\alpha) & \dots & \sin(323\alpha) & \sin(357\alpha) \\ \sin(19\alpha) & \sin(57\alpha) & \sin(95\alpha) & \dots & \sin(361\alpha) & \sin(399\alpha) \\ \sin(21\alpha) & \sin(63\alpha) & \sin(105\alpha) & \dots & \sin(399\alpha) & \sin(441\alpha) \end{bmatrix} \times \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \\ x(8) \\ x(9) \\ x(10) \end{bmatrix}, \quad (1)$$

where  $x(i)$ ;  $(i=0,1,\dots,N-1)$  is a real input sequence.

We will reformulate relation (1) as a parallel decomposition based on a pseudo-circular correlation forms using a single new input restructuring sequence as opposed to other solutions where we have used two such auxiliary input sequences. Further, we'll use the properties of DST-IV kernel and the properties of the Galois Field of indexes to appropriately permute the auxiliary input and output sequences.

To illustrate our approach, we will consider an example with the length  $N=11$  and the primitive root  $g=2$ .

Thus, we will introduce the following auxiliary input sequence  $\{x'(i):i=0,1,\dots,N-1\}$ . It can be recursively computed as follows:

$$x'(N-1) = x(N-1) \quad (2)$$

$$x'(i) = (-1)^i x(i) + x'(i+1) \quad (3)$$

for  $i=N-2,\dots,0$

Using this restructuring input sequence we can reformulate (1) as follows:

$$Y(0) = \sum_{i=0}^{N-1} x(i), \quad (4)$$

$$Y(k) = 2T_C(k) + Y(k-1), \text{ for } k=1,\dots,N-1, \quad (5)$$

with

$$\begin{bmatrix} T_c(1) \\ T_c(2) \\ T_c(3) \\ T_c(4) \\ T_c(5) \\ T_c(6) \\ T_c(7) \\ T_c(8) \\ T_c(9) \\ T_c(10) \end{bmatrix} = x'(0) \cdot \begin{bmatrix} \cos(2\alpha) \\ \cos(4\alpha) \\ \cos(6\alpha) \\ \cos(8\alpha) \\ \cos(10\alpha) \\ \cos(12\alpha) \\ \cos(14\alpha) \\ \cos(16\alpha) \\ \cos(18\alpha) \\ \cos(20\alpha) \end{bmatrix} - \begin{bmatrix} T(1) \cdot \sin(2\alpha) \\ T(2) \cdot \sin(4\alpha) \\ T(3) \cdot \sin(6\alpha) \\ T(4) \cdot \sin(8\alpha) \\ T(5) \cdot \sin(10\alpha) \\ T(6) \cdot \sin(12\alpha) \\ T(7) \cdot \sin(14\alpha) \\ T(8) \cdot \sin(16\alpha) \\ T(9) \cdot \sin(18\alpha) \\ T(10) \cdot \sin(20\alpha) \end{bmatrix}. \quad (6)$$

The new auxiliary output sequence,  $\{T(k) : k = 1, 2, \dots, N-1\}$  can be computed in parallel as two pseudo-cycle convolutions, if the transform length  $N$  is a prime number, as following:

$$\begin{bmatrix} T(2) \\ T(4) \\ T(8) \\ T(6) \\ T(10) \end{bmatrix} = \begin{bmatrix} \sin(16\alpha) & \sin(32\alpha) & -\sin(20\alpha) & \sin(40\alpha) & -\sin(36\alpha) \\ \sin(32\alpha) & -\sin(20\alpha) & \sin(40\alpha) & -\sin(36\alpha) & -\sin(28\alpha) \\ -\sin(20\alpha) & \sin(40\alpha) & -\sin(36\alpha) & -\sin(28\alpha) & -\sin(12\alpha) \\ -\sin(40\alpha) & \sin(36\alpha) & \sin(28\alpha) & \sin(12\alpha) & -\sin(24\alpha) \\ -\sin(36\alpha) & -\sin(28\alpha) & -\sin(12\alpha) & \sin(24\alpha) & -\sin(4\alpha) \end{bmatrix} \times \begin{bmatrix} x'(2) - x'(9) \\ x'(4) - x'(7) \\ x'(8) - x'(3) \\ x'(5) - x'(6) \\ x'(10) - x'(1) \end{bmatrix}, \quad (7)$$

$$\begin{bmatrix} T(9) \\ T(7) \\ T(3) \\ T(5) \\ T(1) \end{bmatrix} = \begin{bmatrix} -\sin(28\alpha) & -\sin(12\alpha) & \sin(24\alpha) & \sin(4\alpha) & \sin(8\alpha) \\ -\sin(12\alpha) & \sin(24\alpha) & -\sin(4\alpha) & -\sin(8\alpha) & \sin(16\alpha) \\ \sin(24\alpha) & -\sin(4\alpha) & \sin(8\alpha) & -\sin(16\alpha) & \sin(32\alpha) \\ \sin(4\alpha) & -\sin(8\alpha) & -\sin(16\alpha) & \sin(32\alpha) & \sin(20\alpha) \\ \sin(8\alpha) & \sin(16\alpha) & \sin(32\alpha) & \sin(20\alpha) & \sin(40\alpha) \end{bmatrix} \times \begin{bmatrix} x'(2) + x'(9) \\ x'(4) + x'(7) \\ x'(8) + x'(3) \\ x'(5) + x'(6) \\ x'(10) + x'(1) \end{bmatrix}. \quad (8)$$

We have used two index mappings,  $\nu(i)$  and  $\mu(i)$ , to 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:

$$\begin{aligned} &\{\nu(i): 1 \rightarrow 2, 2 \rightarrow 4, 3 \rightarrow 8, 4 \rightarrow 6, 5 \rightarrow 10\}, \\ &\{\mu(i): 1 \rightarrow 9, 2 \rightarrow 7, 3 \rightarrow 3, 4 \rightarrow 5, 5 \rightarrow 1\}. \end{aligned}$$

The signs of terms in eqs. (7) and (8) are given by the functions  $\varepsilon(k,i)$  and  $\gamma(k,i)$  defined respectively as follows:

$$\text{a) } \varepsilon(k,i) \text{ is defined by the matrix } \begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \text{ and}$$

$$\text{b) } \gamma(k,i) \text{ is defined by the matrix } \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

### 3. A VLSI Implementation Discussion

Using the algorithm presented in Section 2 we can obtain two linear systolic arrays using a dependence-graph based synthesis procedure. These arrays represent the main part of the architecture used for the VLSI implementation of the derived algorithm. The obtained processing elements consists from a multiplier, an adder and some multiplexers used to manage the differences in sign in eqs. (7) and (8), respectively. We can obtain a further reduction of the hardware complexity using an appropriate hardware sharing method.

One important feature of the proposed solution is its low I/O cost, an aspect that could be very useful in designing systolic arrays where the so called I/O bottle-neck can seriously limit the usefulness of this concept. The tag-control mechanism can be used to place all I/O channels at the two extreme ends of each systolic array and to control the internal registers using only I/O channels placed at the two ends of the linear systolic array.

The pre-processing stage realizes the computation of the auxiliary input sequence using eqs. (2) and (3) and also a permutation of the resulting input sequence. There is also necessary to reorder the input sequence in order to

compute eqs. (2) and (3). Each permutation can be obtained using a RAM with  $N$  words.

The post-processing stage has the role to reorder the auxiliary output sequence  $\{T(k) : k = 1, 2, \dots, N-1\}$  in such a way that to put it in a natural order, to compute the auxiliary output sequence,  $\{T_C(k) : k = 1, 2, \dots, N-1\}$ , and will yield the final output sequence using eq. (5).

#### 4. Conclusions

In this paper it is presented a new VLSI algorithm for type IV DST that leads to an efficient VLSI architecture with appealing features for a VLSI implementation and high performances. The proposed algorithm uses some modular and regular computational structures that can be computed in parallel thus resulting a high throughput VLSI implementation. It can be mapped into two linear systolic arrays with a high throughput and low I/O cost and hardware complexity. By combining two such linear systolic arrays we can obtain an efficient VLSI architecture for type IV DST that is highly regular and modular and having local connections being well adapted to the VLSI technology.

#### REFERENCES

- Ahmed N., Natarajan T., Rao K.R., *Discrete Cosine Transform*. IEEE Trans. Comput., **C-23**, 1, 90-94 (1974).
- 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 (2005).
- Chiper D.F., Swamy M.N.S., Ahmad M.O., Stouraitis T., *A Systolic Array Architecture for the Discrete Sine Transform*. IEEE Trans. On Signal Proc., **50**, 9 (2002).
- Chivukula R.K., Reznik Y.A., *Efficient Implementation of a Class of MDCT/IMDCT Filterbanks for Speech and Audio Coding Applications*. IEEE Internat. Conf. Acoust., Speech, Signal Proc. (ICASSP), July 2008, 213-216.
- Jain A.K., *A Fast Karhunen-Loeve Transform for a Class of Random Processes*. IEEE Trans. Comm., **COM-24**, 10, 1023-1029 (1976).
- Kung H.T., *Why Systolic Architectures*. IEEE Comp., **15**, 37-46 (1982).
- Malvar H.S., *Signal Processing with Lapped Transforms*. Artech House, Boston, 1992.
- Rao K.R., Hwang J.J., *Techniques and Standards for Image, Video and Audio Coding*. Prentice-Hall, Upper Saddle River, NJ, 1996.
- Wang Z., *Comments on Generalized Discrete Hartley Transforms*. IEEE Trans. on Signal Proc., **SP-43**, 1711 (1995).
- White S.A., *Applications of Distributed Arithmetic to Digital Signal Processing: A Tutorial Review*. IEEE ASSP Mag., **6**, 3, 5-19 (1989).

## UN NOU ALGORITM SISTOLIC PENTRU TRANSFORMATA 2-D DST

(Rezumat)

Se prezintă un nou algoritm VLSI pentru transformata DST-IV care conduce la o arhitectură VLSI eficientă având anumite caracteristici atrăgătoare pentru o implementare VLSI și cu performanțe ridicate. Algoritmul propus utilizează niște structuri computaționale modulare și regulate care pot fi calculate în paralel conducând la un „throughput” ridicat. El poate fi mapat pe două arii sistolice având un „throughput” ridicat, o complexitate a operațiilor de intrare/ieșire scăzută și o complexitate hardware redusă. Prin combinarea celor două arii sistolice se poate obține o arhitectură VLSI eficientă pentru transformata DST-IV care are un caracter modular și regulat, cu conexiuni locale, caracteristică specifică ariilor sistolice.

