Journal of Fundamental and Applied Sciences

ISSN 1112-9867

Available online at http://www.jfas.info

Research Article

Special Issue

GENERAL METHOD OF SYNTHESIS BY PLIC/FPGA DIGITAL DEVICES TO PERFORM DISCRETE ORTHOGONAL TRANSFORMATIONS

I. I. Ismagilov[*,1], S. F. Khasanova[1], V. M. Zakharov[2], S. V. Shalagin[2]

[1]Kazan Federal University, Institute of management, economic and finance Kazan, Russian Federation

[2]Computer systems dept. Kazan National Research Technical Universitynamed after A. N. Tupolev — KAI Kazan, Russian Federation

Published online: 08 August 2017

ABSTRACT

A general method is proposed to synthesize digital devices in order to perform discrete orthogonal transformations (DOT) on programmable logic integrated circuits (PLIC) of FPGA class. The basic and the most "slow" operation during DOT performance is the operation of multiplying by a constant factor (constant) - OMC. To perform DOT digital devices are implemented at the use of the same type of IP-cores, which allow to realize OMC. According to the proposed method, OMC is determined on the basis of picturing set over the elements of the Galois field. Due to the distributed computing of nonlinear polynomial function systems defined over the Galois field in PLIC/FPGA architecture, the reduction in the estimates of time complexity concerning OMC performance is achieved. Each non-linear polynomial function, like OMC, is realized on the basis of the same type of IP-cores according to one of the structural schemes in accordance with the requirements for the device to perform DOT. The use of IP cores significantly reduces the cost of designing a device that implements DOT in the PLIC/FPGA architecture.

Keywords - digital signal processing, discrete orthogonal transformations, distributed computing, nonlinear polynomial functions, Galois fields, FPGAs, digital devices

Author Correspondence, e-mail: iiismag@mail.ru

doi: http://dx.doi.org/10.4314/jfas.v9i2s.75

INTRODUCTION

Discrete orthogonal transformations (DOT) [1-3] are a recognized tool to create effective methods for digital signal processing (DSP) problems. At the same time, the requirements for the volume and the speed of numerical data array processing increase constantly and steadily.

Many digital signal processing applications require the creation of high-speed and cost-effective signal processing facilities. This is related to the complexity of the tasks to be solved, the increase of the requirements to the results of processing and to the expanding application of real-time DSP systems. The main ways to improve the performance of DSP tools are technological, architectural and algorithmic one.

At present, intensive studies are carried out in the following areas of DSP algorithm efficiency increase. The first one is the development of the applied theory of DOT in the direction of the development of algorithms that are computationally effective (fast algorithms, the algorithms with reduced computational complexity, balanced computational complexity algorithms), and hardware algorithms oriented to the implementation in the form of specialized computing devices [13, 14]. It should be noted that a wide application during the solution of a number of DSP problems are found by DOT in various ordered systems of discrete Walsh functions [4-7]. A number of generalizations is proposed concerning the systems of discrete Walsh functions for the use in DSP, including the oblique-angled variants, among which we note [8-13]. The efficiency of discrete transformation application with respect to Walsh discrete function systems and their generalizations in DSP is related to the high speed of corresponding fast transformation algorithms.

The second direction is related to the implementation of computationally time-consuming operations, such as multiplication by a constant, based on operations for Galois field [14]. It was shown in [15, 16] that the set of operations of multiplication (OY) on a constant is realizable at the use of distributed computations, namely, the system of nonlinear polynomial functions (NPF) from many variables in respect of Galois field of the form $GF(2^k)$. By distributed computing we mean the ways of computational problem solution with the use of two or more computing devices and the application of a computational process parallelization and the data-flow computing with the preservation of intermediate results.

In order to synthesize (or to create prototypes) such classes of computing devices (CCD) as a "system on a chip", embedded and portable systems, IP-cores (English - Intellectual Property) are used widely - ready-made units applied for microchip design and presented at the level of the abstract description, at the functional and physical levels. With the limitations on

performance and the size of a chip occupied area, IP-cores allow to increase the process of CCD synthesis on microcircuits significantly, including PLIC / FPGA [17 ± 19]. At present time distributed computing systems with programmable architecture (DCS PA) are created for various purposes, using unified basic modules - multiprocessor reconfigurable processors based on PLIC/FPGA [20]. DCS PA, whose elements are configured PLIC / FPGA, allow to implement various CCD created on the basis of the same type of IP cores and reconfigurable in real time [15].

The solution of CCD synthesis problem for the processing of data arrays based on the implementation of DOT is based on the task of the same type of devices implementation performing the OM on a constant within PLIC/FPGA architecture. In order to implement OM on a constant, the same IP-cores in the architecture are used, which perform both the calculation of NPF over Galois field and the operations for the elements of the indicated field. A flexible FPGA interconnection system creates the prerequisites for the distributed execution of OM by a constant based on the device of Galois field theory using these IP cores. The foregoing creates the prerequisites for the use of DCS PA, including the configured PLIC/FPGA in its composition, in order to solve the actual task of DOT distributed execution.

I. DISCRETE ORTHOGONAL TRANSFORMATIONS FOR ELEMENTS OF GALOIS FIELD (THEORETICAL PART)

In order to analyze the data sets of large volumes, various subclasses of DOT [1 - 3] are being used currently. The devices that implement the calculation of DOT are critical ones in terms of high performance provision.

The expressions that constitute DOT are representable in a matrix form [2, 3]:

$$F = D \cdot S, \; S = D^{-1} \cdot F, \qquad (1)$$

where F, S are the sampling matrices of an original signal and the spectral coefficients of dimension $\underbrace{N \times \ldots \times N}_{a}$, D and is the direct and an inverse DOT matrix of the dimension $N \times N$, $D \cdot D^{-1} = I$, I is a single matrix.

Let the elements of the matrices F and D in (1) are presented by n-bit binary vectors, whose values are varied, and the elements of the matrix S are h-bit binary vectors, $h \le 2n$, whose values are constants. MO occur for two n-bit factors, one of which is a constant, and are denoted as MO (n, h).

Note 1. The total number of MO (n, h) required to compute a matrix expression of the form (1) can be reduced with respect to the order N^a by the performance of fast DOT [2].

The approach related to the implementation of MO (n, h) is proposed with the use of NPF in respect of Galois field. The advantage of this approach is that polynomial transformations allow to parallelize a computationally complex multiplication operation over real numbers and the staging of data flow processing. Thus, the implementation of computationally time-consuming operations is performed using distributed computations, based on the same operations for Galois field.

This circumstance makes it possible to reduce the estimates of time complexity to calculate the operations of multiplication by a constant, and, consequently, the estimation of the delay time for CCD functioning that realize different subclasses of DOT: discrete Fourier transformation, discrete Hartley transformation, discrete Walsh transformation, wavelet transformation, etc. [1 - 5]. The task of DOT execution is reduced to the calculation of the same type of MO (n, h).

Let us consider the possibility of MO presentation (n, h) on the basis of NPF and/or their systems for Galois field. Let φ is the mapping of Galois field elements $G_{(v)}$ into itself of the following form:

$$\varphi : \underbrace{G_{(v)} \times ... \times G_{(v)}}_{m} \to G_{(v)} . \qquad (2)$$

According to [21], any mapping of elements $G_{(v)}$ of the form (2) is set by NPF for a given field from *m* variables of the following type:

$$f(q_1, ..., q_m) = \sum_{i_1=0}^{w} ... \sum_{i_m=0}^{w} a_{i_1 ... i_m} q_1^{i_1} ... q_m^{i_m} , \qquad (3)$$

where $w = 2^v - 1$, $a_{i_1 ... i_m}, q_1, ..., q_m \in G_{(v)}$, the symbol Σ denotes the operation of the bitwise sum according to module two. The obtaining of NPF coefficients of the form (3) represented by the matrix $A = \left(a_{i_1 ... i_m}\right)_{2^v \times ... \times 2^v}$, is performed by the system of equations soltion of the following form:

$$V^{(1)}(*, i_2, ..., i_m) = C^{-1} V(*, i_2, ..., i_m),$$
$$V^{(2)}(i_1, *, ..., i_m) = C^{-1} V^{(1)}(i_1, *, ..., i_m), ...,$$
$$V^{(m)}(i_1, ..., i_{m-1}, *) = C^{-1} V^{(m-1)}(i_1, ..., i_{m-1}, *),$$
$$A = V^{(m)}, \quad i_1 = \overline{0, w}, ..., i_m = \overline{0, w},$$

where C^{-1} is the matrix determined for the field $G_{(v)}$, at the size of $2^v \times 2^v$:

$$\tilde{N}^{-1}=\begin{pmatrix} 1 & 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & \xi^{r-1} & \dots & \xi^{r-j} & \dots & \xi & 1 \\ \dots & \dots & & \dots & & \dots & \dots \\ 0 & \xi^{r-k} & \dots & \xi^{k(r-j)\bmod r} & \dots & \xi^{k} & 1 \\ \dots & \dots & & \dots & & \dots & \dots \\ 0 & \xi & \dots & \xi^{j} & \dots & \xi^{r-1} & 1 \\ 1 & 1 & \dots & 1 & \dots & 1 & 1 \end{pmatrix},$$

$j=\overline{1,(r-1)}$, $r=2^{v}-1$, ξ - primitive element $G_{(v)}$.

MO (n, h) can be represented either as two mappings of the form (2), if: $m=2$, $v=n$, $n+1\le h\le 2n$, or in the form of a single mapping (2), if $m=2$, $v=n$, $h\le n$. In general case, the problem of device synthesizing for MO performance (n, h) in PLIC/FPGA architecture is solved by NPF presentation of the form (3) from m variables over the field $G_{(v)}$ at the use of l NPF (3) from z variables over the field $G_{(k)}$ each, $l=]v/k[$, $z=m\cdot l$. The solution of this task proves:

Statement 1. The picturing φ of the form (2) is presented as the family of presentations:

$$\varphi_i:\underbrace{G_{(k)}\times\dots\times G_{(k)}}_{z}\to G_{(k)}, \qquad (4)$$

where $i=\overline{1,l}$, at that each image φ_i can be presented as NPF (3) from $z=m\cdot l$ variables over the field $G_{(k)}$: $l=]v/k[$.

The distributed calculation of MO for two n-bit factors and the h-bit product, $h\le 2n$, – the basic operation for performance (1) is determined on the basis of a family of mappings of the form (4) for which in the general case, $l\le v/k$, $z\le m\cdot l$. MO (n, h) – if: $l=]\log_k h[$, $z=]\log_k n[$.

The statement 1 justifies the method of image representation of the form (2) by the system of l NPF of the form (3) from $z=m\cdot l$ variables over the field $G_{(k)}$, where $l=]v/k[$, which includes three stages. Stage 1 is the representation of each of the m sets of values of the field elements $G_{(v)}$ in the left-hand side and the set of values $G_{(v)}$ in the right-hand side of the image (2) on the basis of l images of the form (2) over the field $G_{(k)}$: $G_{(v)}=\underbrace{\{G_{(k)}\ \dots\ G_{(k)}\}}_{l}$. Stage 2 - on the basis of z sets of values $G_{(k)}$, $z=m\cdot l$, on the left and l sets of values $G_{(k)}$ the obtaining of the image system φ_i, $i=\overline{1,l}$, of the type (4) over the field $G_{(k)}$. Stage 3 – on the basis of each and from φ_i, $i=\overline{1,l}$, we obtain NPF (3) from z variables over the field $G_{(k)}$.

The representation of the mapping system φ_i, $i=\overline{1,l}$, of the type (4) by NPF system from z variables over $G_{(k)}$ allows to solve an actual problem connected with the processing of data

arrays presented in digital form and having a large dimension in a limited period of time by organizing distributed computations. The processing of binary vectors of large dimension is efficiently implemented in Galois field. This opens the possibility to solve the problems of device synthesizing that implement DOT in homogeneous computing network structures that allow parallel implementation. One of the implementations of these structures is DCS PA, both existing ones [20] and prospective.

IP-cores, which allow to calculate multiplication and addition operations over field elements $G_{(k)}$ are realized on the basis of arbitrary Boolean functions calculation of 2k variables (BF (2k)). The following is fair:

Statement 2. Operations for the elements $G_{(k)}$ are realized on the basis of BF(2k): k BF(2k) allow to realize either the operation of multiplication from two arguments or the operation of a bit to bit sum according to module two from k arguments. The estimation of calculation time of each of these operations over $G_{(k)} - T_{(BF(2k))}$

On the basis of statement 1 and 2, a general method is proposed to synthesize digital devices over the Galois field in order to calculate discrete orthogonal transformations, at DCS PA using single-type IP cores. The method includes three stages.

1. Representation of a digital device or its element on the basis of a image family of the form (4) in accordance with the abovementioned procedure.
2. The synthesis of IP-cores to calculate NPF that implements each of the elements (4), based on one of the given structural schemes [22]: parallel, systolic, sequential or parallel-sequential one.
3. The synthesis of IP cores for the computing of multiplication operations and the sum by module 2 over the elements $G_{(k)}$ at the use of k BF(2k) in PLIC/FPGA architecture.

Note 2. In order to ensure distributed calculation of MO by a constant, IP cores used to calculate both NPF of the form (3) for $G_{(k)}$, and the operations on the field elements $G_{(k)}$, can be distributed over different PLIC / FPGA crystals that make the part of DCS PA. In accordance with the requirements imposed on devices synthesized on DCS PA concerning the speed of operation and the number of processor elements, the actual task of IP cores adaptation to compute NPF of the form (3) for the PLIC/FPGA architecture was solved in [16]. In particular, the width and height of the tier-parallel graphs [23], which represent the calculation of NPF type (3), makes it possible to determine the characteristics of the devices that implement these NPFs on DCS PA.

II. STRUCTURAL SCHEMES CALCULATING NON-LINEAR POLYNOMIAL FUNCTIONS OVER $G_{(k)}$ (PRACTICAL PART)

IP-cores, which allow to calculate NPF of the form (3) from z variables over the field $G_{(k)}$ (further referred to as NPF (z, k)), are realized on the basis of structural schemes alternative in terms of time and hardware complexity: parallel, systolic, sequential and parallel-sequential [22]. Let's consider the estimates of hardware and time complexity for these schemes, expressed through the operations for the elements of the field $G_{(k)}$.

In order to calculate NPF terms (z, k) at the use of a parallel scheme, the following multiplication operations for the field $G_{(k)}$: $\left(2^{mk}-1\right)$ on the constant ($\otimes_{const}$) are required, no more than $\left(2^{mk}-2\right)$ of additions ($\oplus$) and $(m-1)\left(2^k-2\right)^m+m\left(2^k-2\right)+\sum_{i=2}^{m-1}(i-1)C_m^i$ of multiplication operations ($\otimes$). The upper estimate of the time complexity for NPF calculation (z, k) – $\left(k+m\cdot k+1+\right]\log_2 m[\,)\cdot T_{(BF(2k))}$

The systolic scheme assumes the use of Horner's scheme for NPF calculation (z, k):

$$f(q_1,\ldots,q_m)=\left(\ldots\left(f_w^{(1)}\left(q_2\ldots q_m\right)q_1+f_{w-1}^{(1)}\left(q_2\ldots q_m\right)\right)q_1+\ldots\right.$$
$$\left.+f_0^{(1)}\left(q_2\ldots q_m\right)=f\left(q_1,f_{i1}^{(1)}\left(q_2\ldots q_m\right)\right),\quad i_1=\overline{0,w},\quad w=2^k-1.\right.$$

In a general case

$$f_{i_1\ldots i_{t-1}}^{(t-1)}\left(q_t\ldots q_m\right)=f_{i_1\ldots i_{t-1}}^{(t-1)}\left(q_t,f_{i_1\ldots i_t}^{(t)}\left(q_{t+1}\ldots q_m\right)\right)=$$
$$=\left(\ldots\left(f_{i_1\ldots i_{t-1}w}^{(t)}\left(q_{t+1}\ldots q_m\right)q_t+f_{i_1\ldots i_{t-1}(w-1)}^{(t)}\left(q_{t+1}\ldots q_m\right)\right)q_t+\ldots\right.$$
$$\left.+f_{i_1\ldots i_{t-1}0}^{(t)}\left(q_{t+1}\ldots q_m\right),\ t=\overline{2,(m-1)},\ i_j=\overline{0,w},\ j=\overline{1,m},\right.$$
$$f_{i_1\ldots i_{m-1}}^{(m)}\left(q_m\right)=f_{i_1\ldots i_{m-1}}^{(m)}\left(q_m,a_{i_1\ldots i_m}\right).$$

In order to calculate NPF (z, k) according to the Horner's scheme, you need to perform the following operations for $G_{(k)}$: $w\left((w+1)^m-1\right)\oplus$, $(w+1)^m(w-1)-w\otimes$, $2^{mk}\otimes_{const}$ and $(w-1)(w+1)^m\oplus_{const}$. The upper estimation of time complexity for a given polynomial calculation – $\left(2(m-1)w+(w-1)+1\right)\cdot T_{(BF(2k))}$, $w=2^k-1$.

The sequential scheme for NPF calculation (z, k), represented by the Horner's scheme, is realized by a single IP-core, which allows us to calculate the following expression over the elements $G_{(k)}$:

$$g^{(t+1)}=g^{(t)}u+b_{w-t}=\varphi\left(g^{(t)},u,b_{w-t}\right),\ t=\overline{1,w},\ w=2^k-1.$$

In order to calculate NPF (z, k), you need to perform one operation $\otimes$ and $\oplus$ for the field $G_{(k)}$. The lower estimation for time complexity calculation $g^{(t+1)}$ – $2\left(2^{mk}-1\right)\cdot T_{(BF(2k))}$.

A parallel-serial circuit (PSC) allows the calculation of NPF values (z, k) by sequential execution groups from d operations of the following type: $a_{i_1 \ldots i_m} q_1^{i_1} \ldots q_m^{i_m}$ for $G_{(k)}$, performed in parallel, $d \in \left[2, 2^{kz-1}\right]$. The estimates of temporal and hardware complexity for PSC occupy an intermediate position between the corresponding estimates for the parallel and systolic structures on the one hand and the sequential scheme on the other hand [22].

The proposed structural diagrams allow the implementation of CCD in accordance with the requirements for the device to perform DOT in PLIC/FPGA architecture.

CONCLUSION

The proposed general method of synthesis by DCS PA to calculate DOT for Galois field allows to provide a high speed of these devices due to distributed computing in the PLIC/FPGA architecture based on arithmetic of Galois fields. The method is relevant for the calculation of computational resources required for the synthesis of digital devices based on DCS PA, both existing and prospective ones. The use of the same type of IP cores to calculate both NPF from many variables for $G_{(k)}$, and the operations for $G_{(k)}$ makes it possible to simplify the process of digital device design significantly concerning this class on DCS PA.

The foregoing allows us to determine the prospective trend to increase the efficiency of DSP algorithms based on DOT due to their hardware implementation during the solution of a wide class of applied problems.

ACKNOWLEDGEMENTS

The work is performed according to the Russian Government Program of Competitive Growth of Kazan Federal University.

REFERENCES

[1] Lawrence R. Rabiner, Bernard Gold, Theory and application of digital signal processing. – Prentice-Hall, 1975. – 762 p.

[2] Richard E. Blahut, Fast algorithms for digital signal processing. Addison-Wesley Pub. Co., 1985. – 441 p.

[3] Alan V. Oppenheim, Ronald W. Schafer, Discrete-Time Signal Processing (3rd Edition) (Prentice Hall Signal Processing). Prentice Hall; 3 edition (August 28, 2009). – 1120 p.

[4] Harmuth, H. F. Sequency Theory: Foundations and Applications (Academic, New York, 1977; Mir, Moscow, 1980).

[5] Ahmed N. and Rao, K. R. Orthogonal Transforms for Digital Signal Processing. Berlin/Heidelberg/New York: Springer-Verlag, 1975.

[6] Golubov, B. I. Yefimov, A. V. and Skvortsov, V. A. Walsh Series and Transforms: Theory and Application (Izdat. LKI, Moscow, 2008) [in Russian]

[7] Ismagilov, I.I. An approach to ordering of systems of the Walsh discrete functions Radioelectronics and Communications Systems, 49 (1), pp. 46-50.

[8] Ismagilov, I.I. A class of discrete orthogonal bases for representing and processing digital signals Automatic Control and Computer Sciences, 30 (3), pp. 62-66.

[9] Ismagilov, I. I. Discrete Transforms in Basis of Walsh-Like Functions: Theory and Application in Digital Signal Processing (Otechestvo, Kazan, 2003) [in Russian].

[10] Ismagilov, I. I. Inclined Rademacher functions: properties and application in digital signal processing problems, Izv. Vyssh. Uchebn. Zaved., Radioelektron. 39(12), 11 (1996); Radioelectron. Commun. Syst. 39(12), 9 (1996).

[11] Ismagilov, I. I. Discrete transforms in basis of piecewise-power functions and their properties," Izv. Vyssh. Uchebn. Zaved., Radioelektron. 44(3), 54 (2001); Radioelectron. Commun. Syst. 44(3), 46 (2001).

[12] Ismagilov, I.I. Oblique generalizations of the Walsh basis Radioelectronics and Communications Systems, 53 (12), pp. 625-635. doi: 10.3103/S0735272710120010

[13] Ismagilov I.I. Algorithms of parametric estimation of polynomial trend models of time series on discrete transforms/ I.I. Ismagilov, S. F. Khasanova// Academy of Strategic Management Journal, Volume 15, Special Issue, 2016. – P. 21-28.

[14] Zakharov, V.M. Executing discrete orthogonal transformations based on computations on the Galois field in the FPGA architecture/ V.M.Zakharov, S.V.Shalagin// 2016 International Siberian Conference on Control and Communications (SIBCON). IEEE. 12-14 May 2016. DOI 10.1109/SIBCON.2016.7491652. P. 1-4. IEEE. http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7491652

[15] V.M. Zakharov. Calculation of nonlinear polynomial functions by a multiprocessor computer system with a programmable architecture. / V.M. Zakharov, S.V. Shalagin // Information technologies. - 2012. - № 5, - pp. 6-11

[16] Shalagin, S.V. Estimation of distributed computation complexity of a nonlinear polynomial function for the field GF (2K) using a multiprocessor computer system. Shalagin S.V. New information technologies and systems: Collection of scientific articles of the XIth international scientific and technical conf. - Penza, November 25-27, 2014. - Penza: PGU publishing house, 2014 - pp. 9 - 12.

[17] Virtex-6 FPGA Family/ Xilinx Inc. Cop. 2015. [Electronic resource]. – Access mode: http://www.xilinx.com/products/silicon-devices/fpga/virtex-6.html

[18] FLEX10K. Embedded Programmable Logic Family/ Altera Inc. Cop. 1998. [Electronic resource]. – Access mode: http://www.allcomponents.ru/pdf/altera/flex10k.pdf

[19] PLIC 5576XC4T/ CJSC «Radiant-Elcom». 1997-2013. [Electronic resource]. – Access mode: http://www.radiant.su/rus/news/?action=show&id=565

[20] Kalyaev I.A. Reconfigurable multicopy computing structures. Kalyayev I.A., I.I. Levin, E.A. Semernikov and others - 2nd ed. - Rostov on/Don: Publishing house UNTS RAS, 2009. - 344 p.

[21] Rudolf Lidl, Harald Niederreiter, Finite Fields. Cambridge University Press, 1984. – 755 p.

[22] S.Shalagin Methods of synthesis for computer technology devices on the basis of nonlinear polynomial functions for a finite field: the author's abstract from the thesis of technical science Doct. / Shalagin Sergey Viktorovich. - Kazan, 2013. - 32 p.

[23] Pospelov D.A. Introduction to the Theory of Computing Systems. Soviet Radio Publ., Moscow, 1972. 280 p. (In Russian).

How to cite this article:
Ismagilov I I, Khasanova S F, Zakharov V M, Shalagin S V. General method of synthesis by plic/fpga digital devices to perform discrete orthogonal transformations. J. Fundam. Appl. Sci., 2017, *9(2S), 998-1007.*