Minimize Logic Synthesis FPGA – Extraction And Substitution Problems

Nguyen Huu Khanh Nhan[1] & Malinichev D.M.[2]

[1]Department of Electrical - Electronic, Ton Duc Thang University, Ho Chi Minh City, Vietnam
[2]Department of Communications - Informatics, Moscow Technical University, Moscow, Russia
E-mail : khanhnhan@tdt.edu.vn[1], mmm-63@list.ru[2]

Abstract - The objective of multi-level logic synthesis of FPGA is to find the "best" multi-level structure, where "best" in this case means an equivalent presentation that is optimal with respect to various parameters such as size, speed or power consumption... Five basic operations are used in order to reach this goal: decomposition, extraction, factoring, substitution and collapsing. In this paper we propose a novel application of Walsh spectral transformation to the evaluation of Boolean function correlation. In particular, we present an algorithm with approach to solve the problems of extraction and substitution based on the use of Walsh spectral presentation. The method, operating in the transform domain, has appeared to be more advantageous than traditional approaches, using operations in the Boolean domain, concerning both memory occupation and execution time on some classes of functions.

Keywords - *logic synthesis, Walsh spectral, boolean function, decomposition.*

I. INTRODUCTION

One of the promising directions of the element base of modern technical devices and systems for various applications are field programmable gate arrays (FPGA), which combine the advantages of standard or custom integrated circuits and gate array design with a convenience device, programmed by the user [1]. And FPGA contain a much larger number of logic gates than the complex programmable logic devices (CPLD), programmable logic arrays (PLA), etc. As a result, one FPGA can replace about 20 CPLDs in a variety of applications, thereby reducing the number of devices used in equipment. In addition, increases its reliability due to a decrease in the length of conductors on the circuit board and a smaller number of individual devices, while decreasing the required size of the circuits on the board and reduces the demands on the supply voltage. In this FPGA are the standard product, and users avoid many costs inherent in ASIC. But the problem of logic synthesis (LS) FPGA is much more complicated than traditional IP or gate array, due to their architectural features as the basic blocks and more complex interconnections between resources in the crystals.

The analysis of architecture and technology of FPGA allows us to conclude that, in addition to common for the entire microelectronics industry trends to increase the degree of integration, improving overall performance, reduce costs, etc., the new trend is the increased ease of design and debug circuits. However, with increasing complexity of both the crystals and projects to the fore front more and more are questions of design and development of algorithms for automatic logic synthesis. It follows that the main problem of logic synthesis (LS) in the basis of FPGA is minimize the number of used logic blocks and reducing the complexity of the trace.

II. KEY TASKS FOR FPGA LOGIC SYNTHESIS

When logic synthesis (LS) FPGA often use the technique of design separation into two distinct phases [3], the technology-independent and technology-dependent (technology mapping).

We describe in more detail the mathematical problems that arise when considers are the basis of FPGA as well as possible and suggest solutions. Based on an analysis of FPGA architecture, we use methods based on five basic mathematical problems: Decomposition [8], factoring, extraction, substitution and collapsing.

To illustrate the features of the problem consider in the basis of FPGA, look at an example. Let's the number of inputs BF is m = 5, and f_1 = abcdeg and f_2 = abc + bde + ae + cd. The functions f_1 and f_2 in its implementation are 6 and 10 literals, respectively. In addition, the function f_1 requires for its optimal implementation of two CLB, while f_2 - only one, since f_2 - function of 5 variables.

we introduce some definitions.

Definition 1. Expression is a free cube if there is another cube, this expression is divisible by precisely: ie, ab + c - is free of a cube, a ab + ac - no.

Definition 2. The main divider expression f - a set of expressions of the form D (f) = {f / C | C-free cube}.

Definition 3. The core of the expression f is the set of expressions of the form K (f) = {g | g $\in$ D (f) and g - free cube}.

Definition 4. The balance of the original function f, associated with the nucleus K, obtained by replacing the variable of this function for all occurrences of K in the f.

Definition 5. The basis function f, denoted as sup (f) - a set of variables which it depends on. In this case, using the operation |sup (f)| can find the inverse function.

Definition 6. The feature is called feasible if |sup (f)| < m, ie. feasible function can be implemented in one CLB.

Definition 7. Boolean network is realizable if every intermediate node implements a workable option.

The number of intermediate nodes implemented Boolean network gives an upper bound on the number CLB, necessary for its implementation. Thus, a fairly obvious looks conclusion in [2] that the synthesis algorithm can be divided into two phases, the first of which involves the construction of realizable networks, and the second - to minimize the number of nodes. And the first step usually involves the use of decomposition and extraction procedures, and the second is factoring, substitution and collapsing.

The above problems arise in the design of any architecture of FPGA. Some of their aspects were considered in [4] in terms of spectral processing BF. These studies showed promise using spectral analysis of BF for solving problems of use FPGA. In this case, the gain occurs mainly in the processing of large amounts of input data.

Note that the complexity of solving all of these math problems can be significantly reduced with the use of spectral representations of BF in the basis of Walsh functions. As a result, methods of implement FPGA can be optimized on the basis of this presentation. In this paper to solve the problems of extraction and substitution.

Extraction is process of identifying and creating some intermediate functions and variables, and re-expressing the original functions in term of the intermediate plus the original variables. The process is used to identify the common sub-expressions.

Substitution is the process of expressing a function F as a function of second function G, plus the original inputs to the function F. This is done by substituting G into F where ever possible.

III. SPECTRAL ANALYSIS OF BOOLEAN FUNCTIONS

We use the definition of BF in the monographs [4], where they are treated as multi-dimensional functions with *m*-inputs and k-outputs, and carry out mapping of the form:

$$f:\{0,1\}^m \to \{0,1\}^k \qquad (1)$$

Set of outputs is denoted as BF $f_{k-1}, \ldots f_0$, and used the decimal indices $x = (X_{m-1,\ldots,}X_0)$ € $\{0,1\}^m$ are calculated the formula:

$$x = \sum_{i=0}^{m-1} x_i 2^i \qquad (2)$$

$$x = \sum_{i=0}^{m-1} x_i$$

$$f = (f_{k-1},\ldots,f_0) \in \{0,1\}^k,\ f = \sum_{i=0}^{k-1} f_i 2^i \qquad (3)$$

Where x and f can be interpreted as the coordinates of the binary vectors to decimal numbers.

Note that the expressions (2) and (3) describe the BF as a piecewise constant function *F (x)* of real argument on the half-open interval $[0,2^m]$. With this notation system of BF (1) can be represented as a lattice of $y = f(x)$, defined at the points 0,1 ..., 2^m -1 interval $[0,2^m]$.

Extend the function $y = f(x)$ to piecewise constant function *F (x)* as follows:

$$F(x) = f(\delta) \text{ variations } x \in [\delta, \delta+1] \qquad (4)$$

We say that a piecewise constant function *F(x)* represents the original system of BF, if it satisfies the condition (4) and f (x) is constructed by formulas (1) and (3). Thus, the foundation can be described as a vector:

$$F = [f(0), f(l).....f(2^m\text{-}l)]^T,$$

where $X = (x_{m-1} \ldots X_0)$, $(0<x<2^m\text{-}l)$ - a set of input vectors, and $f(x)$ is an integer value, where $F_i = [f_i(0), f_i(l)\ldots f_i(2^m\text{-}l)]^T$, and $f_i(x)$, $0<i<k\text{-}l$, a binary value.

It is known that between BF and Walsh functions, there is a relationship, which explains the possibility of effective use of spectral analysis in the basis of Walsh functions to analyze the fleet. In order to determine this relationship, we consider details of the Walsh function.

These functions are piecewise constant and are given on the half-open interval [0, 2^m] expression:

$$W_\omega(x) = (-1)^{\sum_{i=0}^{\omega-1} \omega_{(m-1-i)} 2} \qquad (5)$$

where $0 < \omega < 2^m - 1$, $m \in N$, and ω_i and x_i are determined from the binary representations ω and x.

$$\omega = \sum_{i=0}^{m-1} \omega_i 2^{m-1-i} \qquad (6)$$

$$a.x = \sum_{i=0}^{\infty} x_i 2^{m-1-i} \qquad (7)$$

ω number for the function $W_\omega(x)$ is called its index, and the number of units in the binary expansion ω, ie. Quantity $\|\omega\| = \sum_{i=0}^{m-1} \omega_i$ called weight index or rank.

Consider a subset of Walsh functions with the weight index of 1. Denote their symbols $R_i(x)$ $(i = l, 2, ... m)$. Then (5) (6) we obtain:

$$R_j(x) = W_{2^{(i-1)}}(x) = (-1)^{x^{i-1}} = \exp(j \frac{2\pi}{2} x^{(i-1)}) \qquad (8)$$

$$j = \sqrt{-1} \qquad R_0(x) = W_0(x) = 1$$

Function $R_i(x)$, defined by (8) are called Rademacher functions. As shown in [4], as a product of Rademacher functions can be represented by any Walsh function, ie:

$$W_\omega(x) = \prod_{i=0}^{m-1} (R_{i+1}(x))^{\omega^{(m-1-i)}} \qquad (9)$$

Fast algorithms of spectral transformations in the basis Walsh similar fast discrete multiplicative transformations, using as a basis of trigonometric functions, i.e. in particular, fast Fourier transform (FFT). Issues of generalization of discrete multiplicative transformations over finite fields are considered in some detail [5], where the main theorem and in this area.

Theorem 1. Let $\Phi(x)$ - a step function representing a system of BF in m variables and $S(\omega)$ $(\omega = 0,1, ... 2^m-1)$ - its spectrum by Walsh.

Then $S(\omega) = 2^{-m} a_m(\varpi)$, where if $\omega = \sum_{i=0}^{m-1} \omega_i 2^{m-1-i}$, then $\varpi = \sum_{i=0}^{m-1} \omega_i 2^i$, and

$a_0(\omega) = \Phi(\omega)$,

$a_0(2^{m-1} + \omega) = \Phi(2^{m-1} + \omega)$,

$a_s(\omega) = a_{s-1}(2\omega) + a_{s-1}(2\omega+1)$, (10)

$a_s(2^{m-1} + \omega) = a_{s-1}(2\omega) - a_{s-1}(2\omega+1)$,

$(\omega = 0,1,... 2^{m-1}-1)$; $(s = 1,2,...m)$.

The number of operations N_w, you want to produce this algorithm for computing $S(\omega)$, as in the case of calculating the FFT is equal to $N.log_2N$, where N - number of arguments, or because the actions are performed with a binary representation of the arguments, $N_w = 2^m * m$, Where m - number of bits of the arguments.

Walsh spectrum $\hat{F}$ can be calculated as follows:

$$\hat{F} = [W]_m * F \text{, where}$$

$$[W]_m = \begin{bmatrix} [W]_{m-1} & +[W]_{m-1} \\ [W]_{m-1} & -[W]_{m-1} \end{bmatrix}, \text{ where } [W]_1 = \begin{bmatrix} 1 & +1 \\ 1 & -1 \end{bmatrix}$$

There is also the inverse Walsh, with which you can restore the BF using the relation:

$$\hat{F} : F = \hat{F} * [W]_m$$

This method of computing the Walsh spectrum is more suitable for the calculation of matrices of small dimensions. In this case, the above-introduced the matrix $[W]_m$ are the Hadamard matrices.

Note that all the above properties of Walsh functions can be formulated in terms of Hadamard matrices, so that the Walsh transform is sometimes called the Walsh-Hadamard transform.

IV. CORRELATION ANALYSIS OF BOOLEAN FUNCTIONS

Autocorrelation function of BF $f(x_0, x_1, ..., x_{m-1})$ is determined on the basis of relations:

$$B_2^{(f;f)}(\tau) = \sum_{x=0}^{2^m-1} f(x) f(x \oplus \tau), \qquad (11)$$

where $\tau \in \{0,1,...,2^m-1\}$.

As seen from (11), the original function is related to the autocorrelation function of convolution transforms.

Cross-correlation or simply the correlation function of two BF $f_1(x)$ and $f_2(x)$ is the function:

$$B_2^{(f1;f2)}(\tau) = \sum_{x=0}^{2^m-1} f_1(x) f_2(x \oplus \tau), \qquad (12)$$

where $\tau \in \{0,1,...,2^m-1\}$

Establish a connection between the correlation functions and features considered earlier Walsh. To consider the following two theorems.

Theorem 2. Dual application of Walsh transform to the piecewise constant function *F (x)* does not alter this function up to a normalizing factor of 2^{-m}. Indeed,

$$S_{\overline{S_{f(x)}}}(x) = 2^{-m} f(x) \qquad (13)$$

Proof: due to the symmetry

$$\overline{S_{f(x)}(\omega)} = 2^{-m} \sum_{x=0}^{2^m-1} f(x) W_\omega(x)$$

$$S_{\overline{S_{f(x)}}}(x) = 2^{-m} \sum_{\omega=0}^{2^m-1} S_{f(x)}(\omega) W_x(\omega) =$$

$$= 2^{-m} \sum_{\omega=0}^{2^m-1} (2^{-m} \sum_{g=0}^{2^m-1} f(g) W_\omega(g)) W_\omega(x) =$$

by shifting the argument

$$= 2^{-m} \sum_{\omega=0}^{2^m-1} \sum_{g=0}^{2^m-1} f(g) W_\omega(g \oplus x) =$$

$$= 2^{-2m} \sum_{\omega=0}^{2^m-1} f(g) \sum_{g=0}^{2^m-1} W_\omega(g \oplus x) =$$

From the orthogonal condition

$$= 2^{-m} \sum_{g=0}^{2^m-1} f(g) = 2^{-m} f(x)$$

Theorem 3. Let two BF of *m* arguments, and two lattice functions $f_1(x)$ and $f_2(x)$, corresponding to them. Then

$$B_{2,2}^{(f1,f2)} = 2^{2m} \overline{W(W(f_1)\overline{W(f_2)})} \qquad (14)$$

Proof:

$$S_1(\omega)\overline{S_2(\omega)} = 2^{-2m} \sum_{x1,x2=0}^{2^m-1} f_1(x_1) f_2(x_2) W(x_1) W(x_2)$$

Then, in view of theorems on the completeness and orthogonality, symmetry index and the argument of the argument and shift operations involution [8], we have:

$$\overline{W(W(f_1)\overline{W(\tau)})} = 2^{-m} \sum_{\omega=0}^{2^m-1} S(\omega)\overline{S_2(\omega)} W(\omega) =$$

$$= 2^{-3m} \sum_{\omega=0}^{2^m-1} \sum_{x1,x2=0}^{2^m-1} f_1(x_1) f_2(x_2) \overline{W_\omega(x_1 \oplus x_2 \oplus \tau)} =$$

$$= 2^{-2m} \sum_{x1,x2=0}^{2^m-1} f_1(x_1) f_2(x_1 \oplus \tau) +$$

$$2^{-3m} \sum_{x2 \neq x1 \oplus \tau} f_1(x_1) f_2(x_2) \sum_{\omega=0}^{2^m-1} W_\omega(x_1 \oplus x_2 \oplus \tau) =$$

From the orthogonal condition.

$$= 2^{-2m} \sum_{x1=0}^{2^m-1} f_1(x_1) f_2(x_1 \oplus \tau)$$

Note that Theorem 3 is also known as Wiener-Khinchin theorem [6].

In view of Theorems 2 and 3, we find that for BF, the following relation:

$$B_{2,2}^{(f1,f2)} = 2^{2m} W(W(f_1)W(f_2)). \qquad (15)$$

Properties of the correlation characteristics of BF determined by the properties of convolution transforms of the original features. In particular, the form of these transformations implies the invariance of the correlation characteristics to shift the argument of the original. Converse is also true that the autocorrelation function of the original function can be restored up to a shift of the argument.

For the BF shift of the argument is equivalent to α inversion of those arguments, which correspond to the nonzero components of the binary expansion of α. Therefore, the complexity of realizing the foundation in any basis, is completely determined by its autocorrelation function up to m elements of the inversion, where *m* - number of arguments of BF. This makes it possible to solve all the problems associated with minimizing the complexity of circuits that implement the system of BF, the language of the correlation characteristics, which is done in this paper.

V. RECOGNITION OF BOOLEAN FUNCTIONS

Obviously, for the recognition the linearity of BF is sufficient to calculate its Walsh spectrum, and if this range contains only one non-zero point $\omega = c$ (not counting the point $\omega = 0$), then BF is linear and is represented in the form.

$$f(x) = \bigoplus_{i=0}^{m-1} c_i x_i \qquad (16)$$

Recognition of the self-dual (anti self-duality) BF [9] reduces to the calculation of the value of its autocorrelation function at a point 2^m -1.

Recall that a BF $f(x_0, ..., x_n)$ is called self-dual (anti self-duality) then only when it takes on the opposite sets of opposite (or same anti self-duality function) values. A set of self-dual and anti self-duality functions form the class of functions closed under composition [9]. Since the properties self-dual (anti self-duality) are invariant to a shift in the arguments of BF, their detection may be accomplished by analyzing the autocorrelation function of BF.

Theorem 4. To BF $f(x) = f(x_0, ..., x_n)$ has been self-dual (anti self-duality) is necessary and sufficient that:

$$B(2^m - 1) = 0 \tag{17}$$

$$(B(2^m - 1) = \sum_{x=0}^{2^m-1} f(x)) \tag{18}$$

In addition, in the literature the notion α self-dual (anti self-duality). For BF, it would take place if and only if there is , α such that:

$$f(x_0, ..., x_{m-1}) = \overline{f(x_0^{\alpha_0}, ..., x_{m-1}^{\alpha_{m-1}})} \tag{19}$$

$$(f(x_0, ..., x_{m-1}) = \overline{f(x_0^{\alpha_0}, ..., x_{m-1}^{\alpha_{m-1}})}) \tag{20}$$

Where $x_j^{\alpha_j} = \begin{cases} \overline{x_j} - \alpha_j = 0 \\ x_j - \alpha_j = 1 \end{cases}$ with

$$\alpha = \sum_{i=0}^{m-1} \alpha_i . 2^{m-1-i} \text{ and } \overline{\alpha} = \sum_{i=0}^{m-1} \overline{\alpha}_i . 2^{m-1-i}$$

Note that for the same BF $f(x)$ there may be several $\alpha_1, ..., \alpha_g$ such that $f(x)$ α_i – self-dual $(i = 1, ..., g)$ and a few $\beta_1, \beta_2, ..., \beta_h$ such that $f(x)$ β_i - anti self-duality $(i = 1, ..., h)$. In this case, any BF 0- anti self-duality, and if it β_i - anti self-duality and β_j - anti self-duality, then $(\beta_i \oplus \beta_j)$ anti self-duality. Thus, the set of points for any anti self-duality foundation is a group (with group operation $\oplus$) and the number of points anti self-duality (order of) is equal to 2^i, $i \in \{0,1\}$. Point group, anti self-duality group called the inversion of variables. If $f(x_0, x_2 ...., x_{m-1})$ 2^s – self-duality, then $f(x_0, x_2 ...., x_{m-1})$ does not depend on x_s.

Theorem 5. The function $f(x) = (x_0, ..., x_{m-1})$ has points self-dual $\alpha_1, ..., \alpha_g$ and the points anti self-duality $\beta_1, ..., \beta_h$, if and only if when

$$B(\alpha_i) = 0 \qquad (i = 1, 2, ..., g) \tag{21}$$

$$B(\beta_i) = \sum_{x=0}^{2^m-1} f(x) \qquad (i = 1, 2, ..., h) \tag{22}$$

It is obvious that Theorem 5 generalizes Theorem 4 in $\alpha = 2m - 1$ $(\beta = 2m - 1)$.

consequence. For any BF $f(x)$ the number of points at which the autocorrelation function takes a value $\sum_{x=0}^{2^m-1} f(x)$, equal to 2^i, $i \in \{0,1\}$, and the corresponding samples of the autocorrelation function of a group. Thus, the zeros of the autocorrelation function is determined self-dual point, and the maximum count equal to $\sum_{x=0}^{2^m-1} f(x)$ - point anti self - duality. These results are used in particular for the analysis of linear codes [12].

Since monotone functions represent the other pole of complexity in their implementation of linear functions, the language of correlation functions is not very suitable for implementation. Although, by some weakening of the notion of monotony, which is the concept of a threshold of BF, you can get some interesting results. Thus, in [13] was proposed to implement the foundation for the linear-threshold network, described in terms of Walsh spectral coefficients, and in [14], a tabulation the spectra of threshold functions. In addition, known techniques that allow, in specific cases to obtain engineering solutions for a number of arguments, not exceeding six. In [15] introduced the condition of n-monotony of BF using the Walsh spectral coefficients, but it requires too many definitions and intermediate results and therefore not included here. Some practical results of work [15] can be found in [16].

In order to further address issues of recognition of BF using the Walsh spectra, we introduce additional notation:

$$S(W) = (S(W_0), ..., S(W_{2^n-1})); \tag{23}$$

where S - range of functions; W - domain transformation of BF, in addition, $d_0(W_\omega)$ - the number of constituents of BF f(x) for each value of W_ω .

Then $S(W_j) = d_0(W_j) - d_1(W_j)$ for any $W_j \in W$. We divide the region transform into subsets and call their levels, i.e. $W^n = \{W^{(0)}, ..., W^{(n)}\}$. In addition, we say that $W^j \in W^{(l)}$, $l = 1 \div n$, if $||W_j|| = l$. Consequently, the set of spectral coefficients are also divided into levels $S(W) = \{S^{(0)}, ..., S^{(n)}\}$, where $S(W_j) \in S^{(l)}$, if $||W_j|| = l$. The coefficient $S(W_0)$ indicates the number constituents of the unit BF.

The function $f(x)$ partially symmetric with respect to x_m and x_k , if the permutation x_m and x_k $f(x_i) = f(x_i * P_{m,k}) = f(x_j)$, for all $x_j \in X^n$, where $P_{m,k}$ - substitution matrix dimension n x n, and $det\ P_{m,k} = 1$; $p_{i,j} \in \{0; 1\}$; In addition, $P_{m,k}^{-1} = P_{m,k}^{-T}$. Then obviously $||x_i|| = ||x_j||$.

Theorem 6. BF *f(x)* partially symmetric if and only if $S(W_j) = S(W_q)$, where $W_q = W_j * P_{m,k}$, W_q , $W_j \in W^{(l)}$.

BF called completely symmetric if $f(x_j)= f(x_j * P)$ for any $X_i \in P$, where P - the matrix any permutation of the permutation group of order $n!$. Since any permutation matrix can be represented as a product, then using Theorem 6, we obtain proof of the corollary below.

Consequence. BF is symmetric if and only if $|S(W_j)| = |S(W_q)|$ for all $W_q, W_j \in W^{(l)}$, where $1 = 1 \div n$, i.e. symmetric function is equal in absolute Walsh coefficients at each level.

Thus, the basic properties of the BF, i.e. linearity, self-dual (anti self- duality) different with orders of symmetry and partial symmetry, can be identified using the Walsh spectrum.

VI. PROBLEMS OF EXTRACTION AND SUBSTITUTION.

To describe the algorithm for extraction recall that the autocorrelation function of a Boolean function (BF) [9] is the similarity measure of its output at a certain distance in the space of BF. At the same time unit in the binary representation of addresses maximum ratio BF indicate what arguments are involved in its formation, therefore, the minimum number in the value of the autocorrelation function corresponds to the minimum BF coupling between its arguments. Based on these general considerations, the authors propose an algorithm for solving the problem of extraction, described below.

Initially formed by the coefficients of the autocorrelation function of BF. Then take into account factors "first tier", ie depending on one variable, and among them is sought with a minimum rate. If you have coefficients with the same value, then made advanced search of the coefficients of the relevant variables at the next tier. Depending on the results of this search and select and the minimum ratio. After finding the minimum rate and to determine the original function arguments, on which it depends, there is a transition to the coefficients of the "second tier", that is dependent on two variables. In this case the coefficients in the formation of which was attended by the same arguments as in the formation of a minimum ratio of the first tier, we seek the minimum ratio. Then jumps to the next tier, etc. After completing the loop through all tiers of the above algorithm is repeated for the other variable.

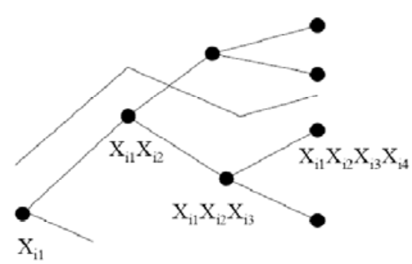
Note that the above algorithm for solving the extraction is the matrix analogue of the well-known algorithm "branch and bound", which guarantees finding is not necessarily better, but quite a good result. A very strong positive aspect of the proposed algorithm from the standpoint of reduction of brute force is the fact that the minimum ratio is sought not the entire table length 2^n, but only among C^i_n coefficients, where n - total number of arguments, and i- number of arguments on a particular tier.

Thus, the proposed algorithm for solving the division consists of the following:

1. Verified by the simultaneous fulfillment of conditions $B(\tau_0) = \max_{|\tau| \neq 0} B(\tau)$ and $\|\tau\| = 1$;
2. Determined $\|\tau\| = x_{i1}, \ldots, x_{in}$;
3. Verified that the inequality $\sum_{j=1}^{n} x_{ij} \leq M$ and $\tau \leq n$;
4. Determined $B(\tau_i) = \min_{|\tau| \leq i} B(\tau)$;
5. Go to step 1.

In the above procedures sequence $\|\tau\|$ - capacity expansion, τ - binary address, M - number of inputs of CLB.

We describe an algorithm for extraction example of the following functions: $f(x_3, x_2, x_1, X_0) = (X_0 \oplus X_1)\&(X_2 \oplus X_3)$. Define F = $[0,0,0,0,0,1,1,2,0,1,1,2,0,2,2,4]^T$, that is B = [36,24,24,16,24,12,12,8,24,12,12,8,16,8,8,4]. From the analysis in that all the functions of the first tier of the same, ie value of the autocorrelation function of BF in these points is 24. By viewing the advanced find out that the function is symmetrical. Therefore, as the first variable we take x_0, having minimum values of the autocorrelation of the variables x_3 and x_2. This means that the separation (cutting) of the original BF should be performed by separating the variable x_0 from x_3 and x_2. After a similar operation for the variable x_1, we obtain a similar result: x_1 requires separation of x_3 and x_2. Given the fact that the restriction on the number of variables is two, we find that this feature should be implemented by cutting into two parts – x_3 and x_2, as well as x_0 and x_1.



Algorithm for solving the extraction.

Another task logic synthesis FPGA - the task of the substitution (merge) - is proposed to solve by analyzing the autocorrelation function of the table corresponding

BF. In this case there are two possible levels of addressing the merge. At a higher level problem is solved not only the merge of the original BF, but also because of previously conducted its decomposition. The task becomes much more complicated because of the substantial increase in the number of searched options.

As one of the possible methods of solving the above problems we propose the following hierarchical algorithm. The first step is the analysis of the autocorrelation function table for BF couples find its values in the formation of which involved no more than three input CLB. The analysis is performed by an algorithm similar to the proposed decomposition, and, if not the subsequent stages, then this problem can be regarded as a special case of decomposition. Obtained for pairs of candidates for a merge originally carried out to verify that the terms of the presence in each pair no more than four inputs. The next step is to check the condition limiting the total number of entries to five. This order of testing candidates for a possible merge due to the fact that the total number of inputs is always restricted by the inequalities: $C_n^3 \le C_n^4 \le C_n^5$, which leads to a reduction in the total number of options searched.

VII. CONCLUTION.

It follows that the use of spectral analysis techniques in the basis of the Walsh representation for solving problems of extraction and substitution is highly effective. In the future the author envisions the building of the upper bounds of the spectral algorithms in their application for synthesis of FPGA.

This problem arise in the design of any architecture of FPGA. Some of their aspects were considered in [10,11] in terms of spectral processing BF. These studies showed promise using spectral analysis of BF for solving problems of use FPGA. In this case, the gain occurs mainly in the processing of large amounts of input data.

REFERENCES

[1] Goldenberg LM,Matyushkin BD,Polak MN "Digital Signal Processing." Moscow: Radio Svyaz, 1985.

[2] Murgai R., Nishizaki Y., Shenoy N., Brayton R. K., Sangiovanni-Vlncentelli A. "Logic synthesis algorithms for programmable gate arrays" In Proc. of the CM/IEEE Design Autom. Conf. (Orlando, FL, June), 620-625. 1990.

[3] F. Mailhot and G. De Micheli, "Algorithms for technology mapping based on binary decision diagrams and on Boolean operations", IEEE Trans. On CAD, vol. 12, no. 3, pp. 599-620, May 1993.

[4] Darrlnger J., Joyner W., Berman L, Trellyan L "Logic Synthesis through local transformation" IBM J. Res. Develop., v.25, №4, pp.272-280, July 1981.

[5] Muzio J.C. "Composite Spectra and the Analysis of Switching Circuits" IEEE Trans. On Comput Vol. C-29, № 8 Aug. 1980.

[6] Karpovsky M.G. "Finite Ortogonal Series In the Design of Digital Devices" NY.:Wiley, 1976.

[7] Zeidman B. "All about FPGAs / Logic Design Line". March 21. 2006.

[8] Karp R. M. "Functional decomposition and switching circuit design" J. SIAM 11,2, 291-335. 1963.

[9] Preparata F.P. "VLSI algorithms and architecture" In M.P. Chytil, V. Koubek (ed.;Lecture notes In Computer Science 176), Springer-Verlag, 1984.

[10] Pratt W., Kane J., Andrews H.C., "Hadamard transform image coding" Proc. IEEE Vol. 57, № 1 pp. 58-68, 1969.

[11] Trimberg S."Reprogrammable Gate Arrays and Applications" Proc. of the IEEE,V.81,N 7, July 1993.

[12] Karp R.M. "deducibility among combinatorial problem on Complexity of Computer computation" ,1972.

[13] Gary M. D. Johnson, "Computers and Intractability problem" ed. AA Friedman - Mir, 1982.

[14] M. Dertouzos, "Threshold Logic," Trans. from English. edited by VI Warsaw, Moscow, Mir.ZhT. – pp.342.

[15] Sapozhenko AA, "On the complexity of disjunctive normal forms obtained by the gradient algorithm," Discrete Analysis. Vyp.21, Novosibirsk, 1972 - pp.62-71.

[16] Proc. IEEE, "Custom Integrated Circuits conference", Boston, Mass., May 13-16, 1990 (N.Y.).1990, pp. 27.5.1-27.5.4.

[17] M. J. Ciesielski. S. Yang, and M. A. Perkowski, "Minimization of multiple-valued logic based on graph coloring," Tech. Rep. TR-CSE- 90.13, Dep. Elect. Comp. Eng., Univ. of Massachusetts. Amherst, Sept. 1990.

◈◈◈