***Proceedings of the 1995 International Workshop on Boolean Problems, Freiberg, Germany, 1996, pp. 115-120***

# Analogues of Boolean Differences and Differentials in Arithmetical Logic[1]

Svetlana N.Yanushkevich

Institute of Computer Science & Information Systems, Technical University, Al.Piastow 41, 71-065, Szczecin, Poland, Tel: 4891-344329, Fax: 4891-340946
e-mail: januszki@beta.ii.tuniv.szczecin.pl

**Abstract**

The paper extend the new conception of an arithmetical analogue of Boolean Difference onto a wide class of operators such as Arithmetical Derivative of a Boolean function with respect to vectors of variables, Differential with respect to variable, Total Differential and others. Some properties of the introduced operators are shown to apply for analysis Boolean functions represented by its arithmetical polynomials forms, including finding tests for switching circuits. All algorithms are represent both in symbolic and matrix form.

## 1 Introduction

Arithmetical representation of Boolean functions have been applied in computer science for a long time, in particular, in synthesis of threshold elements in [12], pseudo-Boolean and integer programming [6,8], fault detection in switching circuits [7,13], Boolean functions systems representation and Petri networks modelling [10,11,14], binary dynamic systems, finite automata analysis [4,5], logic design [3].

A definition of an arithmetical analogue of Boolean Difference was considered in [16], and a partial Gibbs derivative on finite dyadic group was introduced in [2].

We accept, as a base, a conception of the so-called logic Taylor expansion whose coefficients are the values of some Boolean Differences [1,15]. Drawing parallels between the family of logic Taylor expansions (the family of Reed-Muller forms with fixed polarity), a definition of Arithmetical Derivative for Boolean functions was formalised in matrix form and generalised for multivalued logic functions [17,18]. A development of a class of such operators presented in the paper, enriches the Arithmetical Logic in both theoretical and practical areas.

## 2. Arithmetical Derivatives for Boolean functions

The analogue of logic Taylor series in arithmetical logic is the following canonical form describing a family of $2^n$ arithmetical forms with fixed polarity [13,14,17,18]

$$P_c(X) = \sum_{i=0}^{2^n-1} p_c^{(j)} (x_1 \oplus c_1)^{i_1} (x_2 \oplus c_2)^{i_2} .. (x_n \oplus c_n)^{i_n}$$

where $c \in \overline{0, 2^n - 1}$ is the value defining the polynomial's type so that

$$(x_j \oplus c_j)^{i_j} = \begin{cases} x_j \text{ when } c_j = 0, i_j = 1; \\ \overline{x_j} \text{ when } c_j = 1, i_j = 1; \\ 1 \text{ when } c_j \in (0,1), i_j = 0; \end{cases}$$

$c_1, c_2, \dots, c_n$, is the binary representation of $c$; $j = \overline{1,n}$, $p_c^{(j)}$ are the coefficients of the polynomial $P_c(X)$.

Proceeding from the fact that coefficients of the Logic Taylor expansion (Reed-Muller fixed polarity $C$ form) are the values of multi-order Boolean Differences of the function $f(X)$ in the point $C \in \overline{0, 2^n - 1}$ [15], the author has accepted in [17,18]

---

[1] This work was supported by the Research Grant 93-3052 (Reg.no.199-41-086), Ministry of Education and Science, Republic of Belarus, and the personal Research Grant 21-0119/17-99-00, Technical University of Szczecin, Poland

$$p_c^{(J)} = \left. \frac{\tilde{\partial}^{(J)} f(X)}{\tilde{\partial} x_1^{j_1}\, \tilde{\partial} x_2^{j_2} \ldots \tilde{\partial} x_n^{j_n}} \right|_{X=C} .$$

where $J = \sum_{i=1}^{n} j_i$ ; $j_1 j_2 \ldots j_n$ is binary representation of $j$; $\tilde{\partial} x_i^{j_i} = \begin{cases} \tilde{\partial} x_i, j_i = 1, \\ 1, j_i = 0. \end{cases}$

The Derivative introduced by such way was defined as the Arithmetical Derivative of a Boolean function with respect to variable $x_i$ [17,18]:

$$\tilde{\partial} f(X)/\tilde{\partial} x_i = -f(x_1,..,x_i,..,x_n) + f(x_1,..,\bar{x}_i,..,x_n). \quad (1)$$

It follows from (1) that this Derivative can take the values 0, 1,-1 and indicates the direction of changing a function (Table 1).

Table 1

The possible values of the Arithmetical Derivative

| $f(x_1,.., x_i,.. x_n)$ | $f(x_1,..., \bar{x}_i,.. x_n)$ | $\tilde{\partial} f(X)/\tilde{\partial} x_i$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | -1 |
| 1 | 1 | 0 |

The matrix analogue of expression (1) is defined by the formula [17,18]

$$\tilde{\partial} X/\tilde{\partial} x_i = A_{2^n}^{(i)} X , \quad (2)$$

where $2^n \times 2^n$ matrix $A_{2^n}^{(i)}$ is formed by the rule

$$A_{2^n}^{(i)} = I_{2^{i-1}} \otimes \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \otimes I_{2^{n-i}} .$$

A m-ordered arithmetical derivative with respect to variables $x_{i_1}, x_{i_2}, \ldots, x_{i_m} \subset (x_1, x_2, \ldots, x_n)$ of a Boolean function $f(X) = f(x_1, x_2, \ldots, x_n)$ is defined in matrix form as

$$\frac{\tilde{\partial}^{(m)} X}{\tilde{\partial} x_{i_1} \tilde{\partial} x_{i_2} \ldots \tilde{\partial} x_{i_m}} = A_{2^n}^{(i_m)} \ldots A_{2^n}^{(i_2)} A_{2^n}^{(i_1)} X . \quad (3)$$

The result of computing the arithmetical derivative $\tilde{\partial} f(X) / \tilde{\partial} x_i$ in symbolic form is defined by an algebraic sum of some minterms $m_i$ : $f^*(x_1, x_2, \ldots, x_n) = \sum_i m_i$, which is an analogue of notation "sum of products" for a Boolean function.

Let us consider some properties of the Derivative which are quite different from the according properties for Boolean Differences.

*Property 1.* Arithmetical Derivative of a Boolean function with respect to variable $x_i$ is calculated by the following expression:

$$\tilde{\partial} f(X)/\tilde{\partial} x_i = (-1)^{x_i} (-f(x_1,..,0,..,x_n) + f(x_1,..,1,..,x_n)).$$

*Proof*: Using the canonical expansion $f(X) = \bar{x}_i f(x_i = 0) + x_i f(x_i = 1)$, we write

$$\tilde{\partial} f(X)/\tilde{\partial} x_i = -(\bar{x}_i f(x_i = 0) + x_i f(x_i = 1) + (x_i f(x_i = 0) + \bar{x}_i f(x_i = 1) =$$
$$= (-\bar{x}_i + x_i) f(x_i = 0) + (-\bar{x}_i + x_i) f(x_i = 1) = (-\bar{x}_i + x_i)(-f(x_i = 0) + f(x_i = 1))$$

The expression $(-\bar{x}_i + x_i)$ when $\bar{x}_i$=0 is equal to 1=$(-1)^0$, and -1=$(-1)^1$ when $\bar{x}_i$=1, i.e. we can write that $(-\bar{x}_i + x_i) = (-1)^{x_i}$ .

*Example* . Let us compute the Arithmetical Derivative with respect to variable $x_1$ of the Boolean function $f(X) = \bar{x}_1 \bar{x}_2 x_3 \vee \bar{x}_1 x_2 \bar{x}_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 x_2 x_3$ given by its truth column vector $X$=[01101001]. We use expression (2):

$$\frac{\tilde{\partial} X}{\tilde{\partial} x_1} = A_{2^3}^{(1)} X = \begin{bmatrix} -1 & & & & 1 & & & \\ & -1 & & & & 1 & & \\ & & -1 & & & & 1 & \\ & & & -1 & & & & 1 \\ 1 & & & & -1 & & & \\ & 1 & & & & -1 & & \\ & & 1 & & & & -1 & \\ & & & 1 & & & & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \\ -1 \\ 1 \\ 1 \\ -1 \end{bmatrix} .$$

Then

$$\tilde{\partial} f(X)/\tilde{\partial} x_1 = \bar{x}_1(\bar{x}_2\bar{x}_3 - \bar{x}_2 x_3 - x_2\bar{x}_3 + x_2 x_3) - x_1(\bar{x}_2\bar{x}_3 - \bar{x}_2 x_3 - x_2\bar{x}_3 + x_2 x_3) =$$
$$= (-1)^{x_1}(\bar{x}_2\bar{x}_3 - \bar{x}_2 x_3 - x_2\bar{x}_3 + x_2 x_3)$$

The truth column vectors of all Derivatives and all corresponding fixed polarity arithmetical polynomials for the function considered in Example 1 are shown in Table 2.

Table 2.

Values of arithmetical derivatives and their connection with values of arithmetical polynomial coefficients for the Boolean function $f(X) = \bar{x}_1\bar{x}_2 x_3 \vee \bar{x}_1 x_2\bar{x}_3 \vee x_1\bar{x}_2\bar{x}_3 \vee x_1 x_2 x_3$

| | | i | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|---|---|
| $c$ | $c_1c_2c_3$ | $i_1 i_2 i_3$ | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| | | $P_c$ | X | $\frac{\tilde{\partial} X}{\tilde{\partial} x_3}$ | $\frac{\tilde{\partial} X}{\tilde{\partial} x_2}$ | $\frac{\tilde{\partial}^{(2)} X}{\tilde{\partial} x_2 \tilde{\partial} x_3}$ | $\frac{\tilde{\partial} X}{\tilde{\partial} x_1}$ | $\frac{\tilde{\partial}^{(2)} X}{\tilde{\partial} x_1 \tilde{\partial} x_3}$ | $\frac{\tilde{\partial}^{(2)} X}{\tilde{\partial} x_1 \tilde{\partial} x_2}$ | $\frac{\tilde{\partial}^{(3)} X}{\tilde{\partial} x_1 \tilde{\partial} x_2 \tilde{\partial} x_3}$ |
| 0 | 0 0 0 | $P_0$ | 0 | 1 | 1 | -2 | 1 | -2 | -2 | 4 |
| 1 | 0 0 1 | $P_1$ | 1 | -1 | -1 | 2 | -1 | 2 | 2 | -4 |
| 2 | 0 1 0 | $P_2$ | 1 | -1 | -1 | 2 | -1 | 2 | 2 | -4 |
| 3 | 0 1 1 | $P_3$ | 0 | 1 | 1 | -2 | 1 | -2 | -2 | 4 |
| 4 | 1 0 0 | $P_4$ | 1 | -1 | -1 | 2 | -1 | 2 | 2 | -4 |
| 5 | 1 0 1 | $P_5$ | 0 | 1 | 1 | -2 | 1 | -2 | -2 | 4 |
| 6 | 1 1 0 | $P_6$ | 0 | -1 | -1 | -2 | -1 | -2 | -2 | 4 |
| 7 | 1 1 1 | $P_7$ | 1 | 1 | 1 | 2 | 1 | 2 | 2 | -4 |

*Property 2.* Arithmetical Derivative of a Boolean function with respect to variable $x_i$ depends on this variable.

It follows from Property 1.

*Theorem 1.* A Boolean function is arithmetically linear with respect to a variable $x_i$ if its Arithmetical Derivative with respect to this variable takes values 1 or -1 (its truth vector consists from 1 and -1) and (m+1)-order Derivative with respect to $x_i$ and other variables $x_{k_1}, \ldots, x_{k_m} \in (x_1, \ldots, x_n)$

$$\frac{\tilde{\partial}^{(m+1)} f(X)}{\tilde{\partial} x_i \tilde{\partial} x_{k_1} \ldots \tilde{\partial} x_{k_m}} = 0.$$

*Proof:* We will proceed from the following definition of arithmetically linear Boolean function: a Boolean function *f(X)* is linear with respect to variable $x_i$ if it is represented as

$$f(X) = k\, x_i + g(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n),$$

where $k$=1 or -1. Taking into account Property 1, we write

$$\tilde{\partial} f(X)/\tilde{\partial} x_i = -(kx_i + g(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)) + (k\bar{x}_i + g(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)) =$$
$$= -kx_i + k\bar{x}_i = k(-x_i + \bar{x}_i) = k(-1)^{x_i}.$$

Then $\tilde{\partial}^{(2)} f(X)/\tilde{\partial} x_i \tilde{\partial} x_j = -k(-1)^{x_i} + k(-1)^{x_i} = 0.$

For example, function $f(X) = \bar{x}_1\bar{x}_2 x_3 \vee \bar{x}_1 x_2\bar{x}_3 \vee x_1\bar{x}_2\bar{x}_3 \vee x_1 x_2 x_3$ is not arithmetically linear with respect to variable $x_1$ (see Table 2), because Arithmetical Derivatives $\tilde{\partial} f(X)/\tilde{\partial} x_1 \tilde{\partial} x_2, \tilde{\partial} f(X) / \tilde{\partial} x_1 \tilde{\partial} x_3$ are not equal to 0. So, it arithmetical polynomial form is $P(X)=x_3+x_2-2x_2x_3+x_1-2x_1x_3-2x_1x_2+4x_1x_2x_3$. At the same time, this function is linear in usual sense since it corresponds to the linear Reed-Muller expansion $x_1 \oplus x_2 \oplus x_3$.

It should be indicated that the second property allows to identify arithmetically linear Boolean functions. It is very important to analyse arithmetically linear systems and automata [4,5].

# 3. Arithmetical Derivatives for Boolean functions with respect to Vector of variables

Let us introduce the definition of Arithmetical Derivative with respect to vector of variables using analogies with the same type of operators in Boolean algebra.

*Definition.* Arithmetical Derivative of a Boolean function *f(X)* with respect to a vector of variables $X_i=(x_{i_1}, x_{i_2}, \ldots, x_{i_m})$, $X_i \subset X$, is defined as follows

$$\tilde{\partial} f(X)/\tilde{\partial} X_i = -f(X_i, X_k) + f(X_i, \bar{X}_k),$$

where $X_k = (x_{k_1}, x_{k_2}, \ldots, x_{k_{n-m}})$, $X_i \cup X_k$, $X_i \neq X_k$, $\bar{X}_i = (\bar{x}_{i_1}, \bar{x}_{i_2}, \ldots, \bar{x}_{i_m})$.

*Definition*. Arithmetical Derivative of a Boolean function $f(X)$ with respect to a vector of variables $X_i$ in matrix form is expressed by

$$\frac{\partial X}{\partial x_i} = A_{2^n}^{(X_i)} X, \qquad (4)$$

where

$$A_{2^n}^{(X_i)} = -I_{2^n} + \prod_{j=1}^{m} \tilde{D}_{2^n}^{(i_p)}, \quad \tilde{D}_{2^n}^{(i_p)} = I_{2^{i_p - 1}} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes I_{2^{n-i_p}}, p = \overline{1,m}.$$

A connection between the Arithmetical Derivative with respect to vector of variables and $m$-order Derivative is described by the equations

$$\frac{\tilde{\partial} f(X)}{\tilde{\partial}(x_{i_1}, x_{i_2}, \ldots, x_{i_m})} = \sum_{j=1}^{2^m - 1} \frac{\tilde{\partial}^{(J)} f(X)}{\tilde{\partial} x_{i_1}^{j_1} \tilde{\partial} x_{i_2}^{j_2} \ldots \tilde{\partial} x_{i_m}^{j_m}}, \qquad \frac{\tilde{\partial}^{(m)} f(X)}{\tilde{\partial} x_{i_1} \tilde{\partial} x_{i_2} .. \tilde{\partial} x_{i_m}} = \sum_{j=1}^{2^m - 1} (-1)^{m-j} \frac{\tilde{\partial} f(X)}{\tilde{\partial}(x_{i_1}^{j_1}, x_{i_2}^{j_2}, .., x_{i_m}^{j_m})},$$

where $J = \sum_{k=1}^{m} j_k$ ; $j_1, j_2, .., j_m$ is binary representation of $j$; $\tilde{\partial} x_{i_p}^{j_p} = \begin{cases} \tilde{\partial} x_{i_p}, j_p = 1, \\ 1, j_p = 0. \end{cases}$

## 4. Arithmetical Differentials for Boolean functions

Here we extend the class of Differential operators by introducing analogues of Boolean Differentials for Arithmetical Logic. We use a conception of the Differential $dx_i$ of variable $x_i$ as for Boolean case [1,15] : $dx_i = x_i \oplus x_i^*$ , where $x_i$ and $x_i^*$ are an initial and final values of variable. Taking into account this, we write the partial Arithmetical Differential with respect to variable $x_i$

$$\tilde{d}_{x_i} f(X) = -f(x_1, .., x_i, .., x_n) + f(x_1, .., x_i \oplus dx_i, .., x_n).$$

The following equation is true for this operator

$$\tilde{d}_{x_i} f(X) = dx_i \frac{\tilde{\partial} f(X)}{\tilde{\partial} x_i},$$

being the generalisation of the same notation for Boolean partial Differential [1].

A matrix interpretation of the expression is of the form

$$\tilde{d}_{x_i} X = A^{(i)} X,$$

where matrix $A^{(i)}$ is formed from matrix $A_{2^n}^{(i)}$ and $2^n$x$2^n$ matrix $O_{2^n}$ by the rule

$$A^{(i)} = \begin{bmatrix} O_{2^n} \\ --- \\ A_{2^n}^{(i)} \end{bmatrix}$$

And we define a Total Arithmetical Differential of a Boolean function $f(X)$

$$\tilde{d} f(X) = -f(X) + f(X \oplus dX).$$

A connection between the Total Differential and Arithmetical Differences with respect to variables is described as follows

$$\tilde{d} f(X) = \sum_{j=1}^{2^n - 1} \frac{\tilde{\partial}^{(J)} f(X)}{\tilde{\partial} x_1^{j_1} \tilde{\partial} x_2^{j_2} .. \tilde{\partial} x_n^{j_n}} \tilde{d} x_1^{j_1} \tilde{d} x_2^{j_2} .. \tilde{d} x_n^{j_n}$$

We give the definition of the total Arithmetical Differential in matrix form

$$\tilde{d} X = A \ X,$$

where $A = \begin{bmatrix} O_{2^n} \\ ---- \\ A_{2^n}^{(X_1)} \\ \dots\dots\dots \\ A_{2^n}^{(X_{2^n-1})} \end{bmatrix}$, $A_{2^n}^{(X_p)} = -I_{2^n} + \prod_{i=1}^{n} (\widetilde{D}_{2^n}^{(i)})^{p_i}$, $p = \overline{0,2^n - 1}$, $p_1p_2...p_n$ is binary representation of $p$

## 5. Application for finding tests in switching circuits

The Arithmetical Derivatives with respect to variables can be used to describe the conditions to detect stuck-at faults in switching circuits. Moreover, the features of the operators defined in section 2 allow to define how the value of the output function have been changed - from 0 to 1 or from 1 to 0.

The conditions to observe a fault in input $x_i$ and transportation it to output is formally written by the following system:

$$\frac{\widetilde{\partial} f(X)}{\widetilde{\partial} x_i} = 1, \quad \frac{\widetilde{\partial} f(X)}{\widetilde{\partial} x_i} = -1.$$

Using these equations, the expressions to find tests can be formed. So, to find tests to detect stuck-at-o fault, the following system is applied

$$x_i \frac{\widetilde{\partial} f(X)}{\widetilde{\partial} x_i} = 1, \quad x_i \frac{\widetilde{\partial} f(X)}{\widetilde{\partial} x_i} = -1.$$

The equations below are useful to form tests for stuck-at-1 fault detection

$$\bar{x}_i \frac{\widetilde{\partial} f(X)}{\widetilde{\partial} x_i} = 1, \quad \bar{x}_i \frac{\widetilde{\partial} f(X)}{\widetilde{\partial} x_i} = -1.$$

The finding test in the case is not quite different from methods based on Boolean Differences [15], but the process of applying the obtained tests is more similar. So, to conclude that the fault is available, it is enough to fix the output value when testing. It is not required to compare this value with the true value $f(X)$. Thus, if a test is obtained from the first equation of system (when $\widetilde{\partial} f(X)/\widetilde{\partial} x_i = 1$), it is means that true output value is equal to 0, and the false value is 1. If a test have been found from the second equation of the system (when $\widetilde{\partial} f(X)/\widetilde{\partial} x_i = -1$), then true and false values are correspondingly equal to 1 and 0. Therefore the conclusion about available of the fault in input $x_i$ can be made from the output value fixed when testing (Table 3).

Table 3
The values of Arithmetical Derivatives to indicate the fault

| $\widetilde{\partial} f(X)/\widetilde{\partial} x_i$ | Fixed output value | Conclusion about available the fault |
|---|---|---|
| 1 | 0 | No |
| 1 | 1 | Yes |
| -1 | 0 | No |
| -1 | 1 | Yes |

*Example*. Let us find the tests to detect stuck-at-0 and stuck-at-1 for input $x_3$ of the switching circuit realising the function considered in the example above. Results of computations in matrix form are represented in Table 4 ($X_3$ denotes truth vector of function $x_3$).

To detect „stuck-at-0" fault in input $x_3$, we can use one of tests 001,011,101,111, and „stuck-at-1" fault - one of tests 000,010,100,110. For instance, when using test 000 we have to fix the output value: if it is equal to 1, the „stuck-at-1" fault was occurred in input $x_3$ (we have not to know true output value and compare it with observed one).

Table 4
The values of Arithmetical Derivatives and tests to detect „stuck-at" faults

| $x_1x_2x_3$ | $X$ | $X_3$ | $\frac{\widetilde{\partial} X}{\widetilde{\partial} x_3}$ | $X_3 \frac{\widetilde{\partial} X}{\widetilde{\partial} x_3}$ | $\bar{X}_3 \frac{\widetilde{\partial} X}{\widetilde{\partial} x_3}$ |
|---|---|---|---|---|---|
| 000 | 0 | 0 | 1 | 0 | 1 |
| 001 | 1 | 1 | -1 | -1 | 0 |
| 010 | 1 | 0 | -1 | 0 | -1 |
| 011 | 0 | 1 | 1 | 1 | 0 |
| 100 | 1 | 0 | -1 | 0 | -1 |
| 101 | 0 | 1 | 1 | 1 | 0 |
| 110 | 0 | 0 | 1 | 0 | 1 |
| 111 | 1 | 1 | -1 | -1 | 0 |

## Conclusion

The new operators considered above represent, in total, the extension of the Boolean Differential Calculus to Arithmetical Logic. It practically means

- an efficient application of the whole potential of the Boolean Differential Calculus when arithmetical extending of the last one as tools to analyse the binary discrete [1] and discrete-time systems [4,5];
- a using the considered operators together with arithmetical canonical forms for Arithmetic Modelling in binary process control [4].

The worked out matrix algorithms to synthesise these operators allow to realise the parallel computations. It means

- an overcoming the problems of symbolic computations such as complexity and „small size" limitations,
- a practical possibility to realise the computations on homogeneous parallel processors, in particular, systolic arrays [9].

## References


[1] Bochmann, D.; Posthoff, Ch.: Binare dinamische systeme. Springer-Verlag, Berlin, 1981

[2] Edwards, C.R.: The Gibbs dyadic differentiator and its relationship to the Boolean difference. Computers and Electronic Engineering, 5 (1978) 335-344

[3] Falkowski, B.J.; Perkowski, M.A.: A family of all essential radix-2 additional/substruction multi-polarity transforms: algorithms and interpretations in Boolean domain". Proc. 23rd IEEE Int. Symp. on Circuits and Systems. New Orleans, Louisiana (1990) 2913-2916

[4] Franke, D.: Arithmetic Modelling - a Link between Discrete-time Systems and Binary Process Control, Appl.Math. and Comp. Sci. vol.5, No.3 (1995) 511-521

[5] Franke, D.: Sequentielle Systeme-Binare und Fuzzy Automatisierung mit endlich vielen Zustanden. Berlin, VEB Verlag, 1994

[6] Hammer, P.L.; Rudeanu, S.: Boolean methods in operational research and related areas. Berlin: Springer, 1968

[7] Heidtmann, K.D.: Arithmetic spectra applied to stuck-at-fault detection for combinational networks. Proc.2nd Technical Workshop "New Directions in IC Testing", Winnipeg, USA (1987) 4.1-4.13

[8] Ivanescu, P.L.; Rudeanu, S.: Pseudo-Boolean methods for Bivalent Programming. Lecture Notes. Math., 23 (1966) 1114-1122

[9] Kukharev, G.A.; Shmerko, V.P.; Yanushkevich, S.N.: Technique of binary data parallel processing for VLSI. Minsk, Vysheyshaya Shkola, 1991 (In Russian)

[10] Malyugin, V.D.: Arithmetical representation of Petri Nets. Automation and Remote Control (USA) vol.48, no.5 (1987)

[11] Malyugin, V.D.: Realisation of Boolean functions by arithmetical polynomials. Automation and Remote Control (USA) vol.43, no.4, part.1 (April 1982) 496-504

[12] Minsky, M.; Papert, S.: Perceptrons: an introduction to computational geometry. Cambridge, MA: MIT Press, 1969

[13] Muzio, J.C.: Stuck fault sensitivity of Reed-Muller and arithmetical coefficients. Proc.3rd Int.Workshop "Theory and Applications of Spectral Techniques", Ed.by C.Moraga, Dortmund, Germany (1988) 36-45

[14] Shmerko, V.: Synthesis of Arithmetical Forms of Boolean Functions by means of Fourier Transform. Automation and Remote Control (USA) vol.50, no.5, part 2 (Oct.1989) 648-691

[15] Thayse, A.: Boolean Differential calculus. Philips Res.Reports 26 (1971) 229-246

[16] Tosic, J.: Polynomial representations of Boolean functions and their minimisation, Trans.of Academy of USSR Science, Engineering Cybernetics, 3 (1967) (In Russian)

[17] Yanushkevich, S.: Arithmetical Analogues of Reed-Muller Canonical Expansions with fixed Polarity for MVL-functions. Proc.of the Workshop on Applications of Reed-Muller Expansions in Circuit Design, Tokyo (1995) 300-307

[18] Yanushkevich, S.N.: Development of Boolean Differential Calculus methods for arithmetical logic. Automation and Remote Control (USA) vol.55, no.4 (1994)