МАТЕМАТИКА И МАТЕМАТИЧЕСКО ОБРАЗОВАНИЕ, 2015
MATHEMATICS AND EDUCATION IN MATHEMATICS, 2015
Proceedings of the Forty Fourth Spring Conference
of the Union of Bulgarian Mathematicians
SOK "Kamchia", April 2–6, 2015

APPLICATIONS OF THE BINARY REPRESENTATION OF INTEGERS IN ALGORITHMS FOR BOOLEAN FUNCTIONS*

Iliya Bouyukliev, Dusan Bikov

In this note we describe two transformations of boolean functions based on the binary representation of the nonnegative integers. We present corresponding algorithms which are very important in cryptology.

1. Introduction. Boolean functions are basic objects in discrete mathematics. In this note we consider these objects and some of their properties related to cryptography. A boolean function f of n variables is a mapping from $\mathbb{F}_2^n$ into $\mathbb{F}_2$, where $\mathbb{F}_2 = \{0, 1\}$ is a field with 2 elements. A boolean function can be represented in different ways. Two natural representations of a boolean function of n variables are its Truth Table (TT(f)) and its Algebraic Normal Form (ANF(f)). Truth Table is 2^n-dimensional vector which have as coordinates the function values of f for all vectors from $\mathbb{F}_2^n$. We can consider the vectors in $\mathbb{F}_2^n$ as binary presentations of the integers in the interval $[0, \ldots, 2^n - 1]$. This consideration is very useful when we try to describe and explain some transformations of boolean functions and related algorithms. Here we give an approach how binary representations of the nonnegative integers can help us for the calculations related to boolean functions.

Let $S = \{0, 1, 2, \ldots, 2^n - 1\}$. For any integer $u \in S$, $u = u_1 2^{n-1} + u_2 2^{n-2} + \cdots + u_{n-1} 2^1 + u_n$ we correspond the binary vector $\overline{u} = (u_1, u_2, \ldots, u_{n-1}, u_n)$. Let S_{set} be the set $S_{set} = \{\overline{0}, \overline{1}, \overline{2}, \ldots, \overline{2^n - 1}\}$, and let $S_{mat}^{(n)}$ be the matrix $S_{mat}^{(n)} = \left(\overline{0}\ \overline{1}\ \cdots\ \overline{2^n - 1}\right)^t$. The matrices $S_{mat}^{(n)}$ can be defined recursively in the following way: $S_{mat}^{(n+1)} = \begin{pmatrix} 0 & S_{mat}^{(n)} \\ 1 & S_{mat}^{(n)} \end{pmatrix}$.

Any boolean function f of n variables is uniquely determined by its Truth Table $TT(f)$, whose coordinates are the function values of f after the lexicographic ordering of the inputs from S_{set}. We mean by the weight $\mathrm{wt}(f)$ of a function f the weight of the corresponding vector $TT(f)$. Analogously, the distance between two functions is computed by considering the distance between the corresponding Truth Tables.

Another way of uniquely representing a Boolean function f is as a binary polynomial of n variables, whose monomials consist of variables of degree 0 or 1. This polynomial is called the algebraic normal form (ANF) of the function [1]. All boolean functions whose

*2010 Mathematics Subject Classification: 06E30.
Key words: Möbius transform, Walsh transform, algorithms.

ANF are polynomials of degree 1 together with the constant 0 define the family of the linear boolean function.

One of the aims of this paper is to describe and motivate a fast algorithm for the transformation from ANF to TT and vice versa. Very important cryptografic property of a boolean function f is its nonlinearity which is related to the distances from f to the linear functions. The other aim is to present an efficient algorithm for the calculation of the Walsh spectrum (see [1]). Practically, the considered algorithms can be presented as a matrix vector multiplication. In our case the considered matrices have not only recursive structure (coming from the recursive structure of the matrix $S^{(n)}_{mat}$) but this structure is quite specific and enables a very effective (*butterfly*) multiplication. Our algorithms are in the same efficiency class as the previously known but they are much more compact, legible, clear and use smaller number of variables.

2. Algebraic normal form and Truth Table. Let the boolean function f be given by its ANF. Then f is a sum of monomials in the following form $x_{i_1}x_{i_2}\cdots x_{i_k}$, $1 \le i_1 < i_2 < \cdots < i_k \le n$, $0 \le k \le n$. We can define all monomials of this type using the set S_{set}. Let $x^{(*u)}$ be the monomial $M^u(x) = x_1^{u_1}x_2^{u_2}\dots x_n^{u_n}$ where $u \in S_{set}$. The value of $M^u(x)$ for $x = v$, $v \in S_{set}$, is $M^u(v) = v_1^{u_1}v_2^{u_2}\dots v_j^{u_j}\dots v_n^{u_n}$. The Truth Table of $M^u(x)$ is

$$\begin{pmatrix} M^u(\overline{0}) \\ M^u(\overline{1}) \\ \vdots \\ M^u(\overline{2^n-1}) \end{pmatrix} = \begin{pmatrix} \overline{0}^{*u} \\ \overline{1}^{*u} \\ \vdots \\ \overline{2^n-1}^{*u} \end{pmatrix} = \left(S^{(n)}_{mat}\right)^{*u}.$$

Let $f(x) = f_0x^{*\overline{0}} \oplus f_1x^{*\overline{1}} \oplus \cdots \oplus f_{2^n-1}x^{*\overline{2^n-1}}$, where $f_0, f_1, \dots, f_{2^n-1} \in \mathbb{F}_2$. The vector $\overline{f} = (f_0, f_1, \dots, f_{2^n-1})$ determines uniquely the algebraic normal form of the function f. For the Truth Table we have the following:

$$\begin{pmatrix} f(\overline{0}) \\ f(\overline{1}) \\ \vdots \\ f(\overline{2^n-1}) \end{pmatrix} = \begin{pmatrix} \overline{0}^{*\overline{0}} & \overline{0}^{*\overline{1}} & \dots & \overline{0}^{*\overline{2^n-1}} \\ \overline{1}^{*\overline{0}} & \overline{1}^{*\overline{1}} & \dots & \overline{1}^{*\overline{2^n-1}} \\ & & \dots & \\ \overline{2^n-1}^{*\overline{0}} & \overline{2^n-1}^{*\overline{1}} & \dots & \overline{2^n-1}^{*\overline{2^n-1}} \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ \dots \\ f_{2^n-1} \end{pmatrix} = A_n \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{2^n-1} \end{pmatrix},$$

$$A_n = \left(\left(S^{(n)}_{mat}\right)^{*\overline{0}} \left(S^{(n)}_{mat}\right)^{*\overline{1}} \ \cdots \ \left(S^{(n)}_{mat}\right)^{*\overline{2^n-1}}\right) = \left(\begin{pmatrix} 0 & S^{(n-1)}_{mat} \\ 1 & S^{(n-1)}_{mat} \end{pmatrix}^{*\overline{0}} \cdots \begin{pmatrix} 0 & S^{(n-1)}_{mat} \\ 1 & S^{(n-1)}_{mat} \end{pmatrix}^{*\overline{2^n-1}}\right).$$

The first coordinate of the vectors $\overline{i}$ for $i < 2^{n-1}$ is 0. Hence
$((0\ S^{(n-1)}_{mat})^{*\overline{0}} \dots (0\ S^{(n-1)}_{mat})^{*\overline{2^{n-1}-1}}) = ((1\ S^{(n-1)}_{mat})^{*\overline{0}} \dots (1\ S^{(n-1)}_{mat})^{*\overline{2^{n-1}-1}}) = A_{n-1}$.
The first coordinate of the vectors $\overline{i}$ for $i \ge 2^{n-1}$ is 1. In the case $x_1 = 1$ the value of monomials doesn't depend on this coordinate. If $x_1 = 0$ then the value of the monomial is zero. It follows that

$$((1\ S_{mat}^{(n-1)})^{*\overline{2^n-1}} \cdots (1\ S_{mat}^{(n-1)})^{*\overline{2^n-1}}) = A_{n-1},$$
$$((0\ S_{mat}^{(n-1)})^{*\overline{2^n-1}} \cdots (0\ S_{mat}^{(n-1)})^{*\overline{2^n-1}}) = 0.$$

These equalities show that $A_n = \begin{pmatrix} A_{n-1} & 0 \\ A_{n-1} & A_{n-1} \end{pmatrix}$, $A_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$, $A_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$,

$$A_3 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}, \quad A_3\overline{f}^t = \begin{pmatrix} f_1 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \\ f_0 \oplus f_1 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \\ f_0 \oplus & 0 \oplus & f_2 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \\ f_0 \oplus f_1 \oplus f_2 \oplus f_3 \oplus & 0 \oplus & 0 \oplus & 0 \oplus & 0 \\ f_0 \oplus & 0 \oplus & 0 \oplus & 0 \oplus f_4 \oplus & 0 \oplus & 0 \oplus & 0 \\ f_0 \oplus f_1 \oplus & 0 \oplus & 0 \oplus f_4 \oplus f_5 \oplus & 0 \oplus & 0 \\ f_0 \oplus 0 \oplus & f_2 \oplus & 0 \oplus & f_4 \oplus & 0 \oplus f_6 \oplus 0 \\ f_0 \oplus f_1 \oplus f_2 \oplus f_3 \oplus f_4 \oplus f_5 \oplus f_6 \oplus f_7 \end{pmatrix}.$$

The matrix A_n is a $2^n \times 2^n$ binary matrix with determinant 1 so $A_n \in \mathrm{SL}(2^n, \mathbb{F}_2)$. It is easy to see that $A_n^2 = I_{2^n}$ therefore $A_n^{-1} = A_n$. We can conclude the following: We can calculate the Truth Table $TT(f)$ of any boolean function f by multiplication of the matrix A_n by the vector $\overline{f}$ (determined by its ANF) and opposite. In other words, $TT(f) = A_n\overline{f}^t$ and $\overline{f} = A_n \cdot TT(f)^t$. The transform μ, which maps ANF to the Truth Table and vice versa is known as Möbius transform. The binary Möbius transform can be considered also as a permutation of the vectors in $\mathbb{F}_2^{2^n}$ given by the matrix A_n.

Let us consider the sums in $A_3 \cdot \overline{f}^t$ more carefully. We notice that the sum of the first two coordinates in the second row is repeated in any even row, and the sum of coordinates 1 and 2 of row 1 is repeated in any odd row. We see similar repetitions for the nonzero summands 3 and 4 and so on. We can avoid repetitions of sums and in this way to multiply a matrix by a vector in n steps instead of 2^n using the following butterfly diagram (Diagram 1).

Diagram 1

(x_1, x_2, x_3)	ANF		Step 1		Step 2		Step 3
000	f_0	$\longrightarrow$	f_0	$\longrightarrow$	f_0	$\longrightarrow$	f_0
001	f_1	$\searrow$	$f_0 \oplus f_1$	$\longrightarrow$	$f_0 \oplus f_1$	$\longrightarrow$	$f_0 \oplus f_1$
010	f_2	$\longrightarrow$	f_2	$\searrow$	$f_0 \oplus f_2$	$\longrightarrow$	$f_0 \oplus f_2$
011	f_3	$\searrow$	$f_2 \oplus f_3$	$\searrow$	$f_0 \oplus f_1 \oplus f_2 \oplus f_3$	$\longrightarrow$	$f_0 \oplus f_1 \oplus f_2 \oplus f_3$
100	f_4	$\longrightarrow$	f_4	$\longrightarrow$	f_4	$\searrow$	$f_0 \oplus f_4$
101	f_5	$\searrow$	$f_4 \oplus f_5$	$\longrightarrow$	$f_4 \oplus f_5$	$\searrow$	$f_0 \oplus f_1 \oplus f_4 \oplus f_5$
110	f_6	$\longrightarrow$	f_6	$\searrow$	$f_4 \oplus f_6$	$\searrow$	$f_0 \oplus f_2 \oplus f_4 \oplus f_6$
111	f_7	$\searrow$	$f_6 \oplus f_7$	$\searrow$	$f_4 \oplus f_5 \oplus f_6 \oplus f_7$	$\searrow$	$f_0 \oplus f_1 \oplus \cdots \oplus f_7$

The proof why this diagram goes in the general case can be seen in [1]. The diagram 1 can be realized with the following algorithm:

Algorithm 1: Fast Möbius Transform

Input: The Truth Table TT of the Boolean function f, with 2^n entries
Output: The Algebraic Normal Form ANF of the Boolean function f, with 2^n entries

```
j ← 1; ANF ← TT;
while (j < 2^n) do
    for i = 0 to 2^n − 1 do
        if (ī_[n−j+1]=1) then                  /*   if((i&j) == j) * /
                        ANF[i] ← (ANF[i] ⊕ ANF[i − j])
    end for
    j ← 2 ∗ j;
end while.
```

3. Linear boolean functions and Walsh spectrum. Let $f(x) = u_1x_1 \oplus u_2x_2 \oplus \cdots \oplus u_nx_n$ be a linear boolean function of n variables. We use the notation $u_1x_1 \oplus u_2x_2 \oplus \cdots \oplus u_nx_n = x^{(\oplus u)}$. The binary n-dimensional vector u uniquely defines $f(x)$ and therefore we denote it by $f^{(\oplus u)}(x)$. The Truth Table of $f^{(\oplus u)}(x)$ has the form

$$\begin{pmatrix} f^{(\oplus u)}(\overline{0}) \\ f^{(\oplus u)}(\overline{1}) \\ \vdots \\ f^{(\oplus u)}(\overline{2^n-1}) \end{pmatrix} = \begin{pmatrix} \overline{0}^{(\oplus u)} \\ \overline{1}^{(\oplus u)} \\ \vdots \\ \overline{2^n-1}^{(\oplus u)} \end{pmatrix} = (S_{mat}^{(n)})^{(\oplus u)}.$$

The values of the linear functions for $\overline{0}, \overline{1}, \ldots, \overline{2^n-1}$ form the following matrix:

$$H_n^+ = \begin{pmatrix} \overline{0}^{\oplus\overline{0}} & \overline{0}^{\oplus\overline{1}} & \ldots & \overline{0}^{\oplus\overline{2^n-1}} \\ \overline{1}^{\oplus\overline{0}} & \overline{1}^{\oplus\overline{1}} & \ldots & \overline{1}^{\oplus\overline{2^n-1}} \\ & & \ldots & \\ \overline{2^n-1}^{\oplus\overline{0}} & \overline{2^n-1}^{\oplus\overline{1}} & \ldots & \overline{2^n-1}^{\oplus\overline{2^n-1}} \end{pmatrix}$$

Hence

$$H_n^+ = \left((S_{mat}^{(n)})^{\oplus\overline{0}}, (S_{mat}^{(n)})^{\oplus\overline{1}}, \ldots, (S_{mat}^{(n)})^{\oplus\overline{2^n-1}}\right)$$

$$= (\begin{pmatrix} 0S_{mat}^{(n-1)} \\ 1S_{mat}^{(n-1)} \end{pmatrix}^{\oplus\overline{0}} \ldots \begin{pmatrix} 0\ S_{mat}^{(n-1)} \\ 1\ S_{mat}^{(n-1)} \end{pmatrix}^{\oplus\overline{2^{n-1}-1}} \begin{pmatrix} 0\ S_{mat}^{(n-1)} \\ 1\ S_{mat}^{(n-1)} \end{pmatrix}^{\oplus\overline{2^{n-1}}} \ldots \begin{pmatrix} 0\ S_{mat}^{(n-1)} \\ 1\ S_{mat}^{(n-1)} \end{pmatrix}^{\oplus\overline{2^n-1}}).$$

For the matrix H_n^+ we have

$$((0\ S_{mat}^{(n-1)})^{\oplus\overline{0}} \ldots (0\ S_{mat}^{(n-1)})^{\oplus\overline{2^{n-1}-1}}) = ((1\ S_{mat}^{(n-1)})^{\oplus\overline{0}} \ldots (1\ S_{mat}^{(n-1)})^{\oplus\overline{2^{n-1}-1}}) = H_{n-1}^+,$$

$$((1\ S_{mat}^{(n-1)})^{\oplus\overline{2^{n-1}}} \ldots (1\ S_{mat}^{(n-1)})^{\oplus\overline{2^n-1}}) = \overline{H_{n-1}^+},$$

where the matrix $\overline{H_{n-1}^+}$ is obtained from H_{n-1}^+ after replacing 0 by -1 and 1 by 0. It follows that

$$H_n^+ = \begin{pmatrix} H_{(n-1)}^+ & H_{(n-1)}^+ \\ H_{(n-1)}^+ & \overline{H_{(n-1)}^+} \end{pmatrix}, H_1^+ = \begin{pmatrix} 0\ 0 \\ 0\ 1 \end{pmatrix}, H_2^+ = \begin{pmatrix} 0\ 0\ 0\ 0 \\ 0\ 1\ 0\ 1 \\ 0\ 0\ 1\ 1 \\ 0\ 1\ 1\ 0 \end{pmatrix}.$$

It is easy to see that H_n is a symmetric matrix. Its rows (and columns) form n

dimensional linear space. In coding theory this space (without zero coordinate) is known as a simplex code. This space together with its coset with representative $(11\ldots1)$, form the first order Reed-Muller code.

Let $a = (a_1, a_2, \ldots, a_m)$ be a binary vector. The polarity representation $a^{(p)}$ of a is obtained from a after replacing 0 by 1 and 1 by -1. Consider the scalar product $s = a^{(p)} \cdot b^{(p)}$ over $\mathbb{Z}$. Let s^- (respectively s^+) be the number of the coordinates, for which $a_j^{(p)} b_j^{(p)} = -1$ (respectively $a_j^{(p)} b_j^{(p)} = +1$). Then $s^- = d(a, b)$ is the number of coordinates with different value for a and b. And s^+ is the number of the coordinates with equal values for a and b or number of the coordinates with different value for a and $11\ldots1 \oplus b$. We have that $s = s^+ - s^-$ and $m = s^+ + s^-$ or $s^- = (m-s)/2$, $s^+ = (m+s)/2$.

Let us denote by $PTT(f)$ and H the polarity representations of $TT(f)$ and the matrix H^+. The vector $W_f = H \cdot (PTT(f))^t = (f^w(\overline{0}), f^w(\overline{1}), \ldots, f^w(\overline{2^n-1}))$, $W_f = (W_0, \ldots, W_{2^n-1})$, is called Walsh spectrum, and the function $f^w(\overline{a})$ defines the Walsh transform. The value W_i determines the distance between the Truth Table of f and the Truth Table of the linear function $x^{\oplus i}$, which equals to $(2^n - W_i)/2$, and also the distance between $TT(f)$ and the Truth Table of the affine function $1 + x^{\oplus i}$ which equals to $(2^n + W_i)/2$.

Similarly to the previous section, the matrix vector multiplication $H \cdot PTT(f)^t$ can be given by a butterfly diagram and a corresponding algorithm, namely *Diagram 2 and Algorithm 2*.

This algorithm (similarly to the previous one) passes all elements of the matrix $S_{mat}^{(n)}$ in n steps column by column starting from the last one. Depending on the value in the i-th row and $(n-j+1)$-th column of the matrix $S_{mat}^{(n)}$ the algorithm calculates a new values for $W_f[i]$ and $W_f[i + 2^j]$. This algorithm (as the first one) entirely depends on the binary representation of the nonnegative integers smaller than 2^n. It is in the same efficiency class as the previously known algorithms but it is much more compact, legible, clear and uses smaller number of variables.

Let us compare Algorithm 2 with Algorithm 9.3 from the classical book *Algorithmic Cryptanalysis*, p. 275 [2]. Algorithm 2 uses 3 integer variables, respectively this number is 6 in Algorithm 9.3 [2]. Our algorithm has 3 assignments respectively the algorithm in [2] has 5 for the inner loop. Instead of 3 nested loops in [2] Algorithm 2 has 2 nested loops and "if else" construction. In a similar way we can compare Algorithm 1 with previously known algorithms.

Diagram 2

(x_1, x_2, x_3)	$PTT(f)$	Step 1	Step 2	Step 3
000	t_0	$\searrow\ t_0 + t_1$	$\searrow\ t_0 + t_1 + t_2 + t_3$	$\searrow\ t_0 + t_1 + t_2 + t_3 + t_4 + t_5 + t_6 + t_7$
001	t_1	$\nearrow\ t_0 - t_1$	$\searrow\ t_0 - t_1 + t_2 - t_3$	$\searrow\ t_0 - t_1 + t_2 - t_3 + t_4 - t_5 + t_6 - t_7$
010	t_2	$\searrow\ t_2 + t_3$	$\nearrow\ t_0 + t_1 - t_2 - t_3$	$\searrow\ t_0 + t_1 - t_2 - t_3 + t_4 + t_5 - t_6 - t_7$
011	t_3	$\nearrow\ t_2 - t_3$	$\nearrow\ t_0 - t_1 - t_2 + t_3$	$\searrow\ t_0 - t_1 - t_2 + t_3 + t_4 - t_5 - t_6 + t_7$
100	t_4	$\searrow\ t_4 + t_5$	$\searrow\ t_4 + t_5 + t_6 + t_7$	$\nearrow\ t_0 + t_1 + t_2 + t_3 - t_4 - t_5 - t_6 - t_7$
101	t_5	$\nearrow\ t_4 - t_5$	$\searrow\ t_4 - t_5 + t_6 - t_7$	$\nearrow\ t_0 - t_1 + t_2 - t_3 - t_4 + t_5 - t_6 + t_7$
110	t_6	$\searrow\ t_6 + t_7$	$\nearrow\ t_4 + t_5 - t_6 - t_7$	$\nearrow\ t_0 + t_1 - t_2 - t_3 - t_4 - t_5 + t_6 + t_7$
111	t_7	$\nearrow\ t_6 - t_7$	$\nearrow\ t_4 - t_5 - t_6 + t_7$	$\nearrow\ t_0 - t_1 - t_2 + t_3 - t_4 + t_5 + t_6 - t_7$

Algorithm 2: Fast Walsh Transform

Input: The Polarity Truth Table PTT of the Boolean function f, with 2^n entries
Output: The Walsh spectrum W_f of the Boolean function f, with 2^n entries
$j \leftarrow 1$; $W_f \leftarrow PTT$;
while $(j < 2^n)$ do
  for $i = 0$ to $2^n - 1$ do
    if $\overline{i}_{[n-j+1]} = 0$ then $/* \quad if((i\&j) == 0) \quad */$
      $temp \leftarrow W_f[i]$;
      $W_f[i] \leftarrow W_f[i] + W_f[i+j]$;
      $W_f[i+j] \leftarrow temp - W_f[i+j]$;
    end then
  end for
  $j \leftarrow 2 * j$;
end while.

REFERENCES

[1] Carlet C. (2010), *Boolean Functions for Cryptography and Error Correcting Codes.* In: Crama C, Hammer PL, (Eds.), *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, Cambridge University Press, 257–397.

[2] Joux A. Algorithmic Cryptanalysis. Chapman & Hall/CRC Cryptography and Network Security Series, 2012.

Iliya Bouyukliev
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
POBox 323
5000 V. Tarnovo, Bulgaria
e-mail: iliyab@math.bas.bg

Dusan Bikov
Faculty of Mathematics and Informatics
Veliko Tarnovo University
5000 V. Tarnovo, Bulgaria
e-mail: dule.juve@gmail.com

ДВОИЧНОТО ПРЕДСТАВЯНЕ НА ЦЕЛИТЕ ЧИСЛА С ПРИЛОЖЕНИЕ В АЛГОРИТМИ ЗА БУЛЕВИ ФУНКЦИИ

Илия Буюклиев, Душан Биков

В това съобщение се обсъждат някои трансформации на булеви функции, чието описание се прави много естествено чрез двоичното представяне на целите неотрицателни числа. Представени са и някои алгоритми с приложение в криптографията.