Acta Cybernetica 14 (1999) 263-283.

Construction of Recursive Algorithms for Polarity Matrices Calculation in Polynomial Logical Function Representation

Dragan Janković *

Abstract

There is no algorithm for the calculation of optimal fixed polarity expansion. Therefore, the efficient calculation of polarity matrix consisting of all fixed polarity expansion coefficients is very important task. We show that polarity matrix can be generated as convolution of function f with rows of relates transform matrix. The recursive properties of the convolution matrix affect to properties of polarity matrix. In literature are known some recursive algorithms for the calculation of polarity matrix of some expressions for Multiple-valued (MV) functions [3,6]. We give a unique method to construct recursive procedures for the polarity matrices calculation for any Kronecker product based expression of MV functions. As a particular cases we derive two recursive algorithms for calculation of fixed polarity Reed-Muller-Fourier expressions for four-valued functions.

1 Introduction

Compact representation of switching functions is not only the mater of notation convenience, but highly relates to the analysis and synthesis of these functions. Both analysis and synthesis procedures, as well as final realizations, can be greatly simplified by choosing appropriate representations of switching functions.

In the case of Reed-Muller (RM) expressions, the problem to determine the most compact representation reduces to the determination of optimal polarity for switching variables. By choosing between the positive or negative literals for each variable, but not both at the same time, the Fixed polarity RM (FPRM) expressions are defined [5].

In a FPRM, the number of products, or equivalently, the number of non-zero coefficients may be considerably reduced by choosing different polarities for the variables. The FPRM with the minimum number of products is taken as the

*Faculty of Electronic Engineering, University of Niš, Yugoslavia

optimal FPRM for f. If there are two FPRMs with the same number of products, the one with the smaller number of literals in the products is taken.

There is no method to determine apriori the polarities of variables for a given function f. In practice, it is necessary to generate all the FPRMs and chose the optimal one. That can be efficiently done by generating the polarity matrices $\mathbf{P}_{RM}$ whose rows are RM-coefficients for the given f with different polarities of variables. The efficiency of generation of $\mathbf{P}_{RM}$ is based upon its recursive structure originating in the Kronecker product representation of the RM-transform matrix.

Polynomial representations of Multiple-valued (MV) functions are very interesting with advent of multiple-valued circuit technology, in particular recent experience with current-mode circuits that are very attractive for implementation of MV functions. Specially, the realization of the corresponding 4-valued circuit is very efficient. The problem of compact representations is even harder in the case of MV functions. Galois field (GF) expressions are a generalization of RM-expressions to MV case [7]. Optimization of GF-expressions can be studied and solved in a way similar to that used for RM-expressions. In particular, efficient methods for generation of polarity matrices $\mathbf{P}_{GF}$ for GF-expressions of ternary functions are reported in [6], while the corresponding methods for quaternary functions are reported in [3], and further elaborated in [1], [2], [4].

Reed-Muller-Fourier (RMF) expressions are an alternative extension of RM-expressions to MV case [8]. It has been shown that RMF-expressions require on the average smaller number of products than GF-expressions to represent a given function f [9]. The optimization of RMF-expressions is performed in the same way as in the GF-expressions by choosing different polarities for the variables. As in the case of RM and GF-expressions, there are no methods to determine apriori the polarity for the variables in a given f to get the RMF-expression with the minimum number of products. For that reason, the efficient calculation of polarity matrices is a very important task. An analyse of present recursive methods for calculation of polarity matrix for some particular expressions shows that recursive approaches are more efficient than others methods. Therefore, the construction of recursive relations for polarity matrix calculation for various expressions are a very interesting problem.

In this paper, we uniformly consider the coefficients in various expressions for logic functions as spectral coefficients in particular spectral transforms. We show that polarity matrix can be generated as convolution of f with columns of related transform matrix. The recursive properties of the polarity matrix result from properties of the convolution matrix. We give a unique method to construct recursive procedures for the polarity matrices calculation for any Kronecker product based expression of MV functions.

This method involves existing methods as a particular cases and permits various generalizations. For illustration, we derive two recursive algorithms for calculation of fixed polarity Reed-Muller-Fourier expressions for four-valued functions.

2 Notations and Definitions

Definition 1 *Let* $\mathbf{E}(q)$ *be the set of integers modulo* q. n*-variable* q*-valued logical function is mapping*

$$f : \mathbf{E}(q)^n \to \mathbf{E}(q).$$

Definition 2 *Each* n*-variable* q*-valued logical function* f *can be represented in polynomial form*

$$\begin{aligned} f(x_1, \ldots, x_n) &= c_0 \oplus c_1 x_n \oplus c_2 x_n^2 \oplus \ldots \oplus c_{q-1} x_n^{q-1} \oplus c_q x_{n-1} \\ &\oplus c_{q+1} x_n x_{n-1} \oplus \ldots \oplus c_{q^n-1} x_n x_{n-1} \ldots x_1. \end{aligned}$$

The coefficient vector $\mathbf{C}$, consisting from the coefficients $c_i, i = 0, \ldots, q^n - 1$ can be calculated as direct transform of function f, given by its truth vector $\mathbf{F} = [f(0), \ldots, f(q^n - 1)]^T$ i.e.

$$\begin{aligned} \mathbf{C} = (c_0, c_1, \ldots, c_{q^n-1}) &= \mathbf{T}_n \mathbf{F} = \left(\bigotimes_{i=1}^{n} \mathbf{T}_1 \right) \cdot \mathbf{F} \\ &= \left(\bigotimes_{i=1}^{n} \begin{bmatrix} 1 & x_i & x_i^2 & \ldots & x_i^{q-1} \end{bmatrix}^{-1} \right) \cdot \mathbf{F}, \end{aligned} \tag{1}$$

where with $^{-1}$ is denoted the inverse matrix and $\otimes$ denotes Kronecker product. $\mathbf{T}_n$ is transform matrix.

The number of non-zero coefficients in vector $\mathbf{C}$ is usually used criteria of optimality. Optimization can be made by using different polarities of variables.

Definition 3 i*-th polarity of variable* x *in notation* $\overset{i-}{x}$ *is defined as:* $\overset{i-}{x} = x \oplus i, i = 0, ..., q-1$, *for* q*-valued functions.*

If each literal x_i in expansion (1) have complemented or noncomplemented form but not both this expansion is named fixed polarity expansion. For n-variable q-valued function the number of different polarities is q^n.

Theorem 1 *For polarity* $k = (k_1, \ldots, k_n)$ $(< k >= \sum_{i=1}^{n} k_i q^{n-i})$, *the coefficient vector can be calculated as [6]:*

$$\mathbf{C}^{<k>} = \mathbf{T}_n^{<k>} \cdot \mathbf{F} = \left(\bigotimes_{i=1}^{n} \mathbf{T}_1^{(k_i)} \right) \cdot \mathbf{F}, \tag{2}$$

where $\mathbf{T}_1^{(k_i)}$ *is the matrix* $\mathbf{T}_1$ *whose that columns are shifted for* k_i *places in according to the definition of operation* $\oplus$.

Example 1 *Let* f *is two variable function on Galois field* $GF(3)$. *The operations* $\cdot$ *and* $\oplus$ *are multiplication modulo 3 and addition modulo 3 respectively.* $< k >$ *polarity expansion coefficient vector of* f *is given as:*

$$\mathbf{C}^{<k>} = \left(\mathbf{T}_1^{(k_1)} \otimes \mathbf{T}_1^{(k_2)} \right) \cdot \mathbf{F}.$$

If $<k>=<7>$ *then* $c^{<7>}$ *is calculated as:*

$$\mathbf{T}_1^{-1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 2 & 1 \end{bmatrix}, \mathbf{T}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 1 \\ 2 & 2 & 2 \end{bmatrix},$$

$$\mathbf{T}_1^{(1)} = \begin{bmatrix} 0 & 0 & 1 \\ 2 & 1 & 0 \\ 2 & 2 & 2 \end{bmatrix}, \mathbf{T}_1^{(2)} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 2 \\ 2 & 2 & 2 \end{bmatrix}.$$

$$\begin{aligned} c^{<7>} &= \left(\mathbf{T}_1^{(2)} \otimes \mathbf{T}_1^{(1)}\right) \cdot \mathbf{F} \\ &= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 2 \\ 2 & 2 & 2 \end{bmatrix} \otimes \begin{bmatrix} 0 & 0 & 1 \\ 2 & 1 & 0 \\ 2 & 2 & 2 \end{bmatrix} \cdot \mathbf{F} \\ &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 2 \\ 2 & 1 & 0 & 0 & 0 & 0 & 1 & 2 & 0 \\ 2 & 2 & 2 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 2 & 0 & 0 & 2 & 0 & 0 & 2 \\ 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \cdot \mathbf{F}. \end{aligned}$$

Lemma 1 *The coefficient vector of polarity* $<p>$ *can be calculated as*

$$\begin{aligned} \mathbf{C}^{<p>} &= \mathbf{T}_n^{<p>}\mathbf{F} = \left(\bigotimes_{i=1}^{n} \mathbf{T}_1^{<p>}\right) \cdot \mathbf{F} = \mathbf{T}_n \cdot \mathbf{F}^{<p>} = \mathbf{T}_n \cdot \mathbf{F}^{<p_1,p_2,\ldots,p_n>} \\ &= \mathbf{T}_n \cdot \mathbf{F}\left(x_1 \oplus p_2, x_2 \oplus p_2, \ldots, x_n \oplus p_n\right). \end{aligned} \tag{3}$$

Example 2 *Let* f *is the two variable 3-valued function defined on* $GF(3)$ *and represented by truth vector* $\mathbf{F} = (122010210)$. *The vector* $c^{<7>}$ *can be calculated as*

$$c^{<7>} = \mathbf{T}_1^{(2)} \otimes \mathbf{T}_1^{(1)} \cdot \mathbf{F}^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 2 \\ 2 & 1 & 0 & 0 & 0 & 1 & 2 & 0 \\ 2 & 2 & 2 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 2 & 0 & 0 & 2 & 0 & 0 & 2 \\ 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 2 \\ 0 \\ 1 \\ 0 \\ 2 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 2 \\ 2 \\ 1 \\ 1 \\ 2 \\ 0 \end{bmatrix},$$

$$c^{<7>} = \mathbf{T}_2 \cdot \mathbf{F}^{<7>} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2 & 0 & 2 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 2 & 0 & 0 & 2 & 0 & 0 & 2 & 0 & 0 \\ 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 2 \\ 1 \\ 2 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 2 \\ 2 \\ 1 \\ 1 \\ 2 \\ 0 \end{bmatrix}.$$

Definition 4 *The polarity matrix* $\mathbf{P}$ *of an* n*-variable* q*-valued function* $f(x_1, x_2, \ldots, x_n)$ *is a* $(q^n \times q^n)$ *matrix where every row matches a coefficient vector in a different polarity* $< k >$. i*-th row corresponds to a coefficient vector in the* $< i >$*-th polarity, i.e.,* $c^{<i>}$.

Definition 5 *The optimal polarity of function* $f(x_1, x_2, \ldots, x_n)$ *is defined as polarity* k_{opt} *whose coefficient vector has the minimal number of nonzero elements.*

Example 3 *The polarity matrix of a two variable quaternary function* f*, given by truth vector* $\mathbf{F} = (0311132322321002)$ *is given as*

$$\mathbf{P} = \begin{bmatrix} 0 & 3 & 1 & 3 & 1 & 1 & 2 & 2 & 0 & 3 & 2 & 2 & 2 & 2 & 2 & 2 \\ 3 & 2 & 2 & 1 & 0 & 3 & 0 & 2 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 2 \\ 1 & 0 & 1 & 1 & 1 & 3 & 2 & 2 & 0 & 1 & 2 & 2 & 0 & 0 & 2 & 2 \\ 1 & 3 & 0 & 3 & 2 & 1 & 0 & 2 & 3 & 3 & 0 & 2 & 0 & 2 & 0 & 2 \\ 1 & 2 & 3 & 1 & 1 & 2 & 0 & 0 & 2 & 1 & 0 & 0 & 2 & 0 & 2 & 2 \\ 3 & 3 & 2 & 3 & 3 & 2 & 0 & 0 & 1 & 1 & 0 & 0 & 2 & 2 & 0 & 2 \\ 2 & 1 & 3 & 3 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 2 & 2 \\ 3 & 2 & 0 & 1 & 3 & 2 & 0 & 0 & 3 & 1 & 0 & 0 & 2 & 0 & 0 & 2 \\ 2 & 0 & 3 & 1 & 3 & 1 & 0 & 0 & 0 & 1 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 2 & 3 & 2 & 1 & 0 & 0 & 3 & 3 & 0 & 2 & 0 & 0 & 0 & 2 \\ 3 & 3 & 3 & 3 & 1 & 1 & 0 & 0 & 0 & 3 & 2 & 2 & 0 & 0 & 2 & 2 \\ 2 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 2 & 0 & 2 \\ 1 & 3 & 3 & 1 & 3 & 0 & 2 & 2 & 2 & 3 & 0 & 0 & 2 & 0 & 2 & 2 \\ 0 & 0 & 2 & 3 & 3 & 2 & 0 & 2 & 3 & 3 & 0 & 0 & 2 & 2 & 0 & 2 \\ 0 & 2 & 3 & 3 & 1 & 2 & 2 & 2 & 0 & 3 & 0 & 0 & 0 & 2 & 2 & 2 \\ 2 & 3 & 0 & 1 & 3 & 0 & 0 & 2 & 1 & 3 & 0 & 0 & 2 & 0 & 0 & 2 \end{bmatrix} \begin{matrix} c^{<0>} \\ c^{<1>} \\ c^{<2>} \\ c^{<3>} \\ c^{<4>} \\ c^{<5>} \\ c^{<6>} \\ c^{<7>} \\ c^{<8>} \\ c^{<9>} \\ c^{<10>} \\ c^{<11>} \\ c^{<12>} \\ c^{<13>} \\ c^{<14>} \\ c^{<15>} \end{matrix}.$$

3 Convolution

Definition 6 *Convolution of* n*-variable* q*-valued logic functions* f *and* g *is defined as*

$$f \star g(s) = \sum_{x=0}^{q^n-1} f(x) \cdot g(x \oplus s), \; s = 0, 1, \cdots, q^n - 1,$$

$$
\begin{aligned}
f \star g(s_1, \cdots, s_n) &= \sum_{x=(0,0,\cdots,0)}^{q-1,q-1,\cdots,q-1)} f(x_1, \cdots, x_n) \cdot g(x_1 \oplus s_1, \cdots, x_n \oplus s_n), \\
s &= (s_1, s_2, \cdots, s_n), \quad s = \sum_{i=1}^{n} s_i \cdot q^{n-i}, \\
x &= (x_1, x_2, \cdots, x_n), \quad x = \sum_{i=1}^{n} x_i \cdot q^{n-i}.
\end{aligned}
$$

Operations $\oplus$ and $\cdot$ are defined on corresponding algebraic structure.

The convolution matrix is given as:

$$
\mathbf{G}_{conv} = \begin{bmatrix} g(0 \oplus 0) & g(1 \oplus 0) & \cdots & g((q^n-1) \oplus 0) \\ g(0 \oplus 1) & g(1 \oplus 1) & \cdots & g((q^n-1) \oplus 1) \\ \vdots & \cdots & \cdots & \vdots \\ g(0 \oplus (q^n-1)) & g(1 \oplus (q^n-1)) & \cdots & g((q^n-1) \oplus (q^n-1)) \end{bmatrix}. \tag{4}
$$

Now, the convolution of f and g, in according to (4) can be write in form

$$
f \star g = \mathbf{G}_{conv} \cdot f. \tag{5}
$$

Theorem 2 *Convolution of k-th row in transform matrix t_k, with function vector* $\mathbf{F}$ *gives the vector of k-th coefficients in polarity matrix i.e.* $\mathbf{P}^k = t_k \star \mathbf{F}$.

The proof of theorem can be done from the structure of convolution matrix. Proof:

$$
\begin{aligned}
\mathbf{P}_n &= \begin{bmatrix} \mathbf{C}^{(0)} \\ \mathbf{C}^{(1)} \\ \vdots \\ \mathbf{C}^{(q^n-1)} \end{bmatrix} = \begin{bmatrix} \mathbf{T}_n \cdot \mathbf{F}^{(0)} \\ \mathbf{T}_n \cdot \mathbf{F}^{(1)} \\ \vdots \\ \mathbf{T}_n \cdot \mathbf{F}^{(q^n-1)} \end{bmatrix} \\
&= \begin{bmatrix} \sum_{j=0}^{q^n-1} \mathbf{T}_n(0,j)\mathbf{F}(j \oplus 0) & \cdots & \sum_{j=0}^{q^n-1} \mathbf{T}_n(q^n-1,j)\mathbf{F}(j \oplus 0) \\ \sum_{j=0}^{q^n-1} \mathbf{T}_n(0,j)\mathbf{F}(j \oplus 1) & \cdots & \sum_{j=0}^{q^n-1} \mathbf{T}_n(q^n-1,j)\mathbf{F}(j \oplus 1) \\ \vdots & \cdots & \vdots \\ \sum_{j=0}^{q^n-1} \mathbf{T}_n(0,j)\mathbf{F}(j \oplus q^n-1) & \cdots & \sum_{j=0}^{q^n-1} \mathbf{T}_n(q^n-1,j)\mathbf{F}(j \oplus q^n-1) \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{P}^0 & \mathbf{P}^1 & \cdots & \mathbf{P}^{q^n-1} \end{bmatrix},
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{P}^k &= \left[\mathbf{P}^k(0), \mathbf{P}^k(1), \cdots, \mathbf{P}^k(q^n-1)\right], \; k = 0, \cdots, q^n-1, \\
\mathbf{P}^k(i) &= \sum_{j=0}^{q^n-1} \mathbf{T}_n(k,j)\mathbf{F}(j \oplus i), \; i = 0, \cdots, q^n-1.
\end{aligned}
$$

It follows from equation (4)

$$\mathbf{P}^k = \mathbf{T}_n^k \star \mathbf{F}.$$

Example 4 *Tenth column from polarity matrix for function f from example 3 can be calculated as convolution of tenth row in transform matrix and truth-vector* **F***:*

$$\begin{bmatrix} 3\\1\\0\\0\\2\\2\\0\\0\\3\\1\\0\\0\\0\\0\\0\\0 \end{bmatrix} \star \begin{bmatrix} 0\\3\\1\\1\\1\\3\\2\\3\\2\\2\\3\\2\\2\\0\\0\\2 \end{bmatrix} = \mathbf{A} + \mathbf{B} = \begin{bmatrix} 3\\1\\1\\3\\1\\1\\1\\1\\1\\3\\3\\1\\3\\3\\3\\3 \end{bmatrix},$$

where

$$\mathbf{A} = \begin{bmatrix} 3\cdot 0+1\cdot 3+0\cdot 1+0\cdot 1+2\cdot 1+2\cdot 3+0\cdot 2+0\cdot 3\\ 3\cdot 3+1\cdot 1+0\cdot 1+0\cdot 0+2\cdot 3+2\cdot 2+0\cdot 3+0\cdot 1\\ 3\cdot 1+1\cdot 1+0\cdot 0+0\cdot 3+2\cdot 2+2\cdot 3+0\cdot 1+0\cdot 3\\ 3\cdot 1+1\cdot 0+0\cdot 3+0\cdot 1+2\cdot 3+2\cdot 1+0\cdot 3+0\cdot 2\\ 3\cdot 1+1\cdot 3+0\cdot 2+0\cdot 3+2\cdot 2+2\cdot 2+0\cdot 3+0\cdot 2\\ 3\cdot 3+1\cdot 2+0\cdot 3+0\cdot 1+2\cdot 2+2\cdot 3+0\cdot 2+0\cdot 2\\ 3\cdot 2+1\cdot 3+0\cdot 1+0\cdot 3+2\cdot 3+2\cdot 2+0\cdot 2+0\cdot 2\\ 3\cdot 3+1\cdot 1+0\cdot 3+0\cdot 2+2\cdot 2+2\cdot 2+0\cdot 2+0\cdot 3\\ 3\cdot 2+1\cdot 2+0\cdot 3+0\cdot 2+2\cdot 1+2\cdot 0+0\cdot 0+0\cdot 2\\ 3\cdot 2+1\cdot 3+0\cdot 2+0\cdot 2+2\cdot 0+2\cdot 0+0\cdot 2+0\cdot 1\\ 3\cdot 3+1\cdot 2+0\cdot 2+0\cdot 2+2\cdot 0+2\cdot 2+0\cdot 1+0\cdot 0\\ 3\cdot 2+1\cdot 2+0\cdot 2+0\cdot 3+2\cdot 2+2\cdot 1+0\cdot 0+0\cdot 0\\ 3\cdot 1+1\cdot 0+0\cdot 0+0\cdot 2+2\cdot 0+2\cdot 3+0\cdot 1+0\cdot 1\\ 3\cdot 0+1\cdot 0+0\cdot 2+0\cdot 1+2\cdot 3+2\cdot 1+0\cdot 1+0\cdot 0\\ 3\cdot 0+1\cdot 2+0\cdot 1+0\cdot 0+2\cdot 1+2\cdot 1+0\cdot 0+0\cdot 3\\ 3\cdot 2+1\cdot 1+0\cdot 0+0\cdot 0+2\cdot 1+2\cdot 0+0\cdot 3+0\cdot 1 \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} 3\cdot2+1\cdot2+0\cdot3+0\cdot2+0\cdot1+0\cdot0+0\cdot0+0\cdot2 \\ 3\cdot2+1\cdot3+0\cdot2+0\cdot2+0\cdot0+0\cdot0+0\cdot2+0\cdot1 \\ 3\cdot3+1\cdot2+0\cdot2+0\cdot2+0\cdot0+0\cdot2+0\cdot1+0\cdot0 \\ 3\cdot2+1\cdot2+0\cdot2+0\cdot3+0\cdot2+0\cdot1+0\cdot0+0\cdot0 \\ 3\cdot1+1\cdot0+0\cdot0+0\cdot2+0\cdot1+0\cdot0+0\cdot3+0\cdot1 \\ 3\cdot0+1\cdot0+0\cdot2+0\cdot1+0\cdot0+0\cdot3+0\cdot1+0\cdot0 \\ 3\cdot0+1\cdot2+0\cdot1+0\cdot0+0\cdot0+0\cdot1+0\cdot1+0\cdot3 \\ 3\cdot2+1\cdot1+0\cdot0+0\cdot0+0\cdot2+0\cdot1+0\cdot0+0\cdot1 \\ 3\cdot0+1\cdot3+0\cdot1+0\cdot1+0\cdot0+0\cdot1+0\cdot3+0\cdot3 \\ 3\cdot3+1\cdot1+0\cdot1+0\cdot0+0\cdot3+0\cdot3+0\cdot2+0\cdot1 \\ 3\cdot1+1\cdot1+0\cdot0+0\cdot3+0\cdot1+0\cdot2+0\cdot3+0\cdot3 \\ 3\cdot1+1\cdot0+0\cdot3+0\cdot1+0\cdot1+0\cdot3+0\cdot1+0\cdot2 \\ 3\cdot1+1\cdot3+0\cdot2+0\cdot3+0\cdot2+0\cdot2+0\cdot3+0\cdot2 \\ 3\cdot3+1\cdot2+0\cdot3+0\cdot1+0\cdot2+0\cdot3+0\cdot2+0\cdot2 \\ 3\cdot2+1\cdot3+0\cdot1+0\cdot3+0\cdot3+0\cdot2+0\cdot2+0\cdot2 \\ 3\cdot3+1\cdot1+0\cdot0+0\cdot2+0\cdot2+0\cdot2+0\cdot2+0\cdot3 \end{bmatrix}.$$

4 Calculation of the Polarity Matrix

The polarity matrix can be calculated directly with equation (2). The complexity of this direct method is $(q^n)^3$ i.e. practically unuseful for large q. For the calculation of polarity matrix can be used the FFT-like method. The complexity of this method is $n(q^n)^2$. In [3,6] is shown that the polarity matrices can be generated efficiently by recursive relations. Proposed procedure is given only for GF(3) "recursion by column" and "recursion by row" for GF(4). In this section, we give the unique method for the generation of recursive relations for the calculation of polarity matrix for arbitrary finite fields. Both "recursion by columns" and "recursion by rows" are considered. Our method is generalization of methods proposed in [3] and [6].

4.1 Unique method for the generation of recursive relations for the polarity matrix construction

If we have in mind that transform matrix is given in Kronecker product form, the next theorem is obviously.

Theorem 3 *Let* $\mathbf{T}(n)$ *is transformation matrix given as:* $\mathbf{T}(n) = \bigotimes_{i=1}^{n} \mathbf{T}_i$ *where the dimension of matrix* $\mathbf{T}_i$ *is* $q_i \times q_i$ *. Element from* p *row and* r *column* $C_{<r>}^{<p>}$ *in polarity matrix* $\mathbf{P}$ *is given as:*

$$C_{<r>}^{<p>} = C_{<r_1,r_2,\cdots,r_i,\cdots,r_n>}^{<p_1,p_2,\cdots,p_i,\cdots,p_n>} = \sum_{l=0}^{q_i-1} \mathbf{T}_i(r_i,l) \cdot C_{<r_1,r_2,\cdots,0,\cdots,r_n>}^{<p_1,p_2,\cdots,l,\cdots,p_n>}, \ i = 1,\cdots,n$$

If $\mathbf{T}_i = \mathbf{T}_j, \ \forall i,j \in \{1,2,\cdots,n\}$ *then*

$$C_{<r>}^{<p>} = C_{<r_1,r_2,\cdots,r_i,\cdots,r_n>}^{<p_1,p_2,\cdots,p_i,\cdots,p_n>} = \sum_{l=0}^{q_i-1} \mathbf{T}(r_i,l) \cdot C_{<r_1,r_2,\cdots,0,\cdots,r_n>}^{<p_1,p_2,\cdots,l,\cdots,p_n>}, \; i = 1,\cdots,n$$

or in matrix form

$$\mathbf{P} = P_{q^n \times q^n(n)} = \left[p^{n-1}(i,j)\right], \; p^{n-1}(i,j) = \sum_{l=0}^{q-1} T(j,l) \cdot p^{n-1}(l \oplus i, 0), \; j \neq 0. \quad (6)$$

This relation is recurrence by columns. ¿From recurrence relation (6) by first columns where we start from the first column i.e. 0-column it can be derived recurrence matrix relation started from any column k. Derived recurrence matrix relation we called "recurrence by k-th column". Recurrence by k-th column can be derived if we each element in k-th column from (6) denote with one letter and calculate relations. In this manner can be calculated "recurrence by k-th row", too.

The formal method for construction recurrence matrix relation for polarity matrix calculation may be presented through following steps:

1. The generation of $q \times q$ simbolic matrix $\mathbf{B}$ as $\mathbf{B} = \left[\mathbf{B}^{i \oplus j}\right], \; 0 \leq i,j \leq q-1$.
2. The generation of $q \times q$ matrix $\mathbf{Q} = \mathbf{T}^{-1} \cdot \mathbf{B}$.
3. If it wish the recurrence by k-th column/row, the elements from k-th column/row are substituted with one letter $P^i, \; 0 \leq i \leq q-1$. These substitutions give equations system consisting of q equations.
4. Solving the generating equations system.
5. The modification of the matrix $\mathbf{Q}$ in according to the solutions of previous equation system.
6. The substitution $\mathbf{Q}$ with P_n and P^i with P_{n-1}^i .

This method can be generalized to the case when matrices $\mathbf{T}_i$ are different, e.i. $\mathbf{T}_n = \bigotimes_{i=1}^n \mathbf{T}_i, \; \mathbf{T}_i \neq \mathbf{T}_j$ if $i \neq j$. In this case, it is not possible to generate polarity matrix by only one recurrence matrix equation. The polarity matrix can be generated by means n recurrence matrix equations similar to the above matrix equations. For each of n steps, we generate recurrence matrix relations based on the matrix $\mathbf{T}_i$. Namely, we run above method n times, substituting $\mathbf{T}_1^{-1}$ with $\mathbf{T}_i^{-1}, \; 1 \leq i \leq n$. Obviously, dimensions of matrices $\mathbf{B}$ and $\mathbf{Q}$ are equal $q_i \times q_i$. This will be illustrated in following example.

Example 5 *Let* $\mathbf{T} = \mathbf{T}_1 \otimes \mathbf{T}_2, \; \mathbf{T}_1 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \; \mathbf{T}_1^{-1} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix},$

$$\mathbf{T}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 3 & 2 & 1 \end{bmatrix}, \; \mathbf{T}_2^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 3 & 2 \\ 0 & 1 & 2 & 3 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

The recurrence matrix equations are:

$$P_n^1 = \begin{bmatrix} P_{n-1}^0 & P_{n-1}^0 + P_{n-1}^1 \\ P_{n-1}^1 & P_{n-1}^0 + P_{n-1}^1 \end{bmatrix},$$

$$P_n^2 = \begin{bmatrix} P_{n-1}^0 & P_{n-1}^1 + 3P_{n-1}^2 + 2P_{n-1}^3 & P_{n-1}^1 + 2P_{n-1}^2 + 3P_{n-1}^3 & S \\ P_{n-1}^1 & P_{n-1}^2 + 3P_{n-1}^3 + 2P_{n-1}^0 & P_{n-1}^2 + 2P_{n-1}^3 + 3P_{n-1}^0 & S \\ P_{n-1}^2 & P_{n-1}^3 + 3P_{n-1}^0 + 2P_{n-1}^1 & P_{n-1}^3 + 2P_{n-1}^0 + 3P_{n-1}^1 & S \\ P_{n-1}^3 & P_{n-1}^0 + 3P_{n-1}^1 + 2P_{n-1}^2 & P_{n-1}^0 + 2P_{n-1}^1 + 3P_{n-1}^2 & S \end{bmatrix},$$

where

$$S = P_{n-1}^0 + P_{n-1}^2 + P_{n-1}^2 + P_{n-1}^3.$$

Proposed above method we explain in next section for the case of polarity matrix of Reed-Muller-Fourier expression of quaternary functions.

5 RMF-expressions for Quaternary Functions

To make the paper self-contained, we present in this section basic definitions for RMF-expressions for quaternary functions. Then, we consider their optimization by choosing different polarities for variables. It is assumed single polarity for a variable in the expression. In that way the Fixed polarity RMF (FPRMF) expressions are defined.

Let $E(4)$ be the set of integers modulo 4 with the addition and multiplication modulo 4 shown in Table 1 and Table 2.

Table 1: Addition modulo 4.

$\oplus$	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

Table 2: Multiplication modulo 4.

$\cdot$	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

Define the exponentiation 4EXP and multiplication 4AND, denoted by $*$ and $\circ$, respectively, as in Table 3 and Table 4. Denote by $\mathbf{J}$ the space of n-variable quaternary functions, i.e., $f : \mathbf{E}(4)^n \to \mathbf{E}(4)$.

Definition 7 *The operator $D(n)$ in $\mathbf{J}$ is defined, in the matrix notation, by a $(4^n \times 4^n)$ diagonal matrix given by $\mathbf{D}(n) = diag(3, 1, \cdots, 1)$.*

Definition 8 *RMF-expression of a function $f \in \mathbf{J}$ given by its truth-vector $\mathbf{F} = [f(0), \ldots, f(4^n \times 4^n)]^T$ is given by [9]*

Table 3: Exponentiation 4EXP.

*	0	1	2	3
0	3	0	0	0
1	3	1	0	0
2	3	2	3	0
3	3	3	1	1

Table 4: Multiplication 4AND.

∘	0	1	2	3
0	0	0	0	0
1	0	3	2	1
2	0	2	0	2
3	0	1	2	3

$$f(x_1,\cdots,x_n) = \left(\mathbf{D}(n)\left(\bigotimes_{i=1}^{n}\left[\begin{array}{cccc} 1 & x_i & x_i^{*2} & x_i^{*3} \end{array}\right]\right)\right)\cdot\mathbf{A}, \qquad (7)$$

where $\mathbf{A} = [a(0),\ldots,a(4^n-1)]^T$ *is the vector of RMF-coefficients determined by the matrix relation*

$$\mathbf{A} = \mathbf{R}(n)\cdot\mathbf{F},$$

where

$$\mathbf{R}(n) = 3\bigotimes_{i=1}^{n}\mathbf{R}_i,\ \ \mathbf{R}_i = \left[\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 1 & 3 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 1 & 3 & 3 \end{array}\right].$$

In this relation, $\otimes$ denotes the Kronecker product and x_i^{*j}, $j \in \{2,3\}$ denotes the j-th power of x_j with respect to 4EXP.

In (7), the addition and multiplication are performed modulo 4.

6 Fixed Polarity RMF-expressions

Similarly as for RM-expressions for switching functions, and GF-expressions for MV functions, optimisation of RMF-expressions means reduction of the number of products, i.e., the number of non-zero RMF-coefficients. As noted above, the optimisation of RMF-expressions is possible if we use different polarities for the variables. For a p-valued variable, we consider $p-1$ complements defined by $\overset{i-}{x} = x\oplus i$, $i\in\{1,\cdots,p-1\}$. Thus, in a FPRMF-expression, a variable can appear as the positive literal x_i or any of $p-1$ negative literals $\overset{i-}{x}$, but not as few of them at the same time. Therefore, there are p^n different polarity FPRM-expressions for a given n-variable function f. For $p=4$, the complements are $\overset{1-}{x}$, $\overset{2-}{x}$, $\overset{3-}{x}$, and thus, there exist 4^n different FPRMF-expressions for a quaternary function f. These different possible FPRMF-expressions are determined through the polarity vector $\mathbf{H} = (h_1, \cdots, h_n)$, where the value of $h_i \in \{0, 1, 2, 3\}$ determines polarity of the literal chosen for the variable x_i.

Definition 9 *For $f \in \mathbf{J}$ given by the truth-vector $\mathbf{F}$, the FPRMF-expression with the polarity vector $\mathbf{H} = (h_1, \cdots, h_n)$ is given by*

$$f(x_1, \cdots, x_n) = \left(\mathbf{D}(n)\left(\bigotimes_{i=1}^{n}\begin{bmatrix} 1 & {}^{h_i-}x_i & {}^{h_i-}x_i^{*2} & {}^{h_i-}x_i^{*3} \end{bmatrix}\right)\right)\left(\left(3\bigotimes_{i=1}^{n}\mathbf{R}_i^{h_i}\right)\mathbf{F}\right), \quad (8)$$

where $\mathbf{R}_i^{h_i}$ is derived from $\mathbf{R}_i$ by the cyclic shift of its columns for h_i places. Thus,

$$\mathbf{R}_1^0 = \mathbf{R}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 3 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 1 & 2 & 3 \end{bmatrix}, \quad \mathbf{R}_1^1 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 3 & 0 & 0 & 1 \\ 2 & 1 & 0 & 1 \\ 1 & 3 & 3 & 1 \end{bmatrix},$$

$$\mathbf{R}_1^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 3 \\ 1 & 0 & 1 & 2 \\ 3 & 3 & 1 & 1 \end{bmatrix}, \quad \mathbf{R}_1^3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 3 & 0 \\ 0 & 1 & 2 & 1 \\ 3 & 1 & 1 & 3 \end{bmatrix}.$$

Example 6 *The zero-polarity FPRMF-expression ($\mathbf{H} = [0, 0]$) for two-variable function f, given by the truth vector $\mathbf{F} = [0311132322321002]^T$ is*

$$\begin{aligned} f \;=\; & 3x_2 \oplus x_2^{*2} \oplus 3x_2^{*3} \oplus x_1 \oplus x_1 \circ x_2 \oplus 2x_1 \circ x_2^{*2} \oplus 2x_1 \circ x_2^{*3} \oplus 3x_1^{*2} \circ x_2 \otimes \\ & \otimes 2x_1^{*2} \circ x_2^{*2} \oplus 2x_1^{*2} \circ x_2^{*3} \oplus 2x_1^{*3} \oplus 2x_1^{*3} \circ x_2 \oplus 2x_1^{*3} \circ x_2^{*3} \oplus 2x_1^{*3} \circ x_2^{*3}. \end{aligned}$$

Definition 10 *For a given $f \in \mathbf{J}$, the FPRMF-expression with the minimum number of non-zero coefficients is the optimal FPRM-expression for f.*

Example 7 *The optimal polarity RMF-expression for function f in Example 6 corresponds to the polarity vector $\mathbf{H} = [2, 3]$, and is given by.*

$$\begin{aligned} f = \; & 2 \oplus {}^{3-}x_2^{*3} \oplus {}^{2-}x_1 \circ {}^{3-}x_2 \oplus {}^{2-}x_1^{*2} \oplus {}^{2-}x_1^{*2} \circ {}^{3-}x_2 \oplus 2\,{}^{2-}x_1^{*2} \circ {}^{3-}x_2^{*3} \\ & \oplus 2\,{}^{2-}x_1^{*3} \circ {}^{3-}x_2 \oplus 2\,{}^{2-}x_1^{*3} \circ {}^{3-}x_2^{*3}. \end{aligned}$$

7 RMF-polarity Matrix

Similarly as in RM and GF expressions, an efficient way to determine the optimal polarity FPRMF-expression for a given function f is to calculate first the corresponding polarity matrix. Therefore, in this section we define polarity matrix for FPRMF-expressions for quaternary functions.

Definition 11 *The RMF polarity matrix* $\mathbf{P}_{RMF}$ *for* $f \in \mathbf{J}$ *is a* $(4^n \times 4^n)$ *matrix whose the i-th row consists of the coefficients in the FPRMF-expression for* f, *for the polarity vector* $\mathbf{H} = [i_1, \cdots, i_n]$ *where* $(i_1, \cdots, i_n)$ *is the quaternary representation of* i.

Example 8 *The RMF polarity matrix for function* f *in Example 6 is given by*

$$\mathbf{P}_{RMF} = \begin{bmatrix} 0 & 3 & 1 & 3 & 1 & 1 & 2 & 2 & 0 & 3 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 2 & 2 & 1 & 0 & 3 & 0 & 2 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 2 \\ 3 & 0 & 1 & 1 & 1 & 3 & 2 & 2 & 0 & 1 & 2 & 2 & 0 & 0 & 2 & 2 \\ 3 & 3 & 0 & 3 & 2 & 1 & 0 & 2 & 3 & 3 & 0 & 2 & 0 & 2 & 0 & 2 \\ 3 & 2 & 3 & 1 & 1 & 2 & 0 & 0 & 2 & 1 & 0 & 0 & 2 & 0 & 2 & 2 \\ 1 & 3 & 2 & 3 & 3 & 2 & 0 & 0 & 1 & 1 & 0 & 0 & 2 & 2 & 0 & 2 \\ 2 & 1 & 3 & 3 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 2 & 2 \\ 1 & 2 & 0 & 1 & 3 & 2 & 0 & 0 & 3 & 1 & 0 & 0 & 2 & 0 & 0 & 2 \\ 2 & 0 & 3 & 1 & 3 & 1 & 0 & 0 & 0 & 1 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 2 & 3 & 2 & 1 & 0 & 0 & 3 & 3 & 0 & 2 & 3 & 3 & 3 & 2 \\ 1 & 3 & 3 & 3 & 1 & 1 & 0 & 0 & 0 & 3 & 2 & 2 & 0 & 0 & 2 & 2 \\ 2 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 2 & 0 & 2 \\ 3 & 3 & 3 & 1 & 3 & 0 & 2 & 2 & 2 & 3 & 0 & 0 & 2 & 0 & 2 & 2 \\ 0 & 0 & 2 & 3 & 3 & 2 & 0 & 2 & 3 & 3 & 0 & 0 & 2 & 2 & 0 & 2 \\ 0 & 2 & 3 & 3 & 1 & 2 & 2 & 2 & 0 & 3 & 0 & 0 & 0 & 2 & 2 & 2 \\ 2 & 3 & 0 & 1 & 3 & 0 & 0 & 2 & 1 & 3 & 0 & 0 & 2 & 0 & 0 & 2 \end{bmatrix}.$$

8 Calculation of RMF Polarity Matrix

Harking and Moraga in [6] gave a method for the calculation of polarity matrices $\mathbf{P}_{GF}$ for GF-expressions of ternary functions. Their method starts from the truth-vector $\mathbf{F}$ of f. Unlike to that, Falkowski and Rahardja proposed method for calculation of polarity matrices $\mathbf{P}_{GF}$ for GF-expressions of quaternary functions starting from zero-polarity GF-expression coefficients vector [3]. In this section, we give two recursive methods for FPRMF polarity matrix calculation. The first method, named "recursion by columns", starts from the truth-vector $\mathbf{F}$ while the other named "recursion by rows", starts from the zero-polarity RMF-coefficient vector $\mathbf{A}$.

Recursion by columns

Now, we will construct the recurrence matrix relation for RMF polarity matrix calculation using proposed formal method. First, we define matrix $\mathbf{B}$.

Definition 12 *For an n-variable quaternary function* $f(x_1, x_2, \cdots, x_n)$ *the* $(4^n \times 4^n)$ *matrix* $\mathbf{B}$ *is defined as* $\mathbf{B} = \left[B^{i \oplus j}\right]$, *where* $\oplus$ *is the operation addition modulo 4.*

$$\mathbf{B} = \begin{bmatrix} B^0 & B^1 & B^2 & B^3 \\ B^1 & B^2 & B^3 & B^0 \\ B^2 & B^3 & B^0 & B^1 \\ B^3 & B^0 & B^1 & B^2 \end{bmatrix}.$$

Based on matrix $\mathbf{B}$ we generate the recursive square matrix $\mathbf{Q}_n$.

$$\begin{aligned} \mathbf{Q}_n &= \left(\mathbf{R}_1^{-1} \cdot \mathbf{B}\right)^T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 3 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 1 & 3 & 3 \end{bmatrix} \cdot \begin{bmatrix} B^0 & B^1 & B^2 & B^3 \\ B^1 & B^2 & B^3 & B^0 \\ B^2 & B^3 & B^0 & B^1 \\ B^3 & B^0 & B^1 & B^2 \end{bmatrix} \\ &= \begin{bmatrix} B^0 & B^0+3B^1 & B^0+2B^1+B^2 & B^0+B^1+3B^2+3B^3 \\ B^1 & B^1+3B^2 & B^1+2B^2+B^3 & B^1+B^2+3B^3+3B^0 \\ B^2 & B^2+3B^3 & B^2+2B^3+B^0 & B^2+B^3+3B^0+3B^1 \\ B^3 & B^3+3B^0 & B^3+2B^0+B^1 & B^3+B^0+3B^1+3B^2 \end{bmatrix} \end{aligned}$$

$$= \begin{bmatrix} Q_{n-1}^0 & Q_{n-1}^0+3Q_{n-1}^1 & W_{13} & W_{14} \\ Q_{n-1}^1 & Q_{n-1}^1+3Q_{n-1}^2 & W_{23} & W_{24} \\ Q_{n-1}^2 & Q_{n-1}^2+3Q_{n-1}^3 & W_{33} & W_{34} \\ Q_{n-1}^3 & Q_{n-1}^3+3Q_{n-1}^0 & W_{43} & W_{44} \end{bmatrix}, \qquad (9)$$

where

$$\begin{aligned} W_{13} &= Q_{n-1}^0+2Q_{n-1}^1+Q_{n-1}^2, & W_{14} &= Q_{n-1}^0+Q_{n-1}^1+3Q_{n-1}^2+3Q_{n-1}^3, \\ W_{23} &= Q_{n-1}^1+2Q_{n-1}^2+Q_{n-1}^3, & W_{24} &= Q_{n-1}^1+Q_{n-1}^2+3Q_{n-1}^3+3Q_{n-1}^0, \\ W_{33} &= Q_{n-1}^2+2Q_{n-1}^3+Q_{n-1}^0, & W_{34} &= Q_{n-1}^2+Q_{n-1}^3+3Q_{n-1}^0+3Q_{n-1}^1, \\ W_{43} &= Q_{n-1}^3+2Q_{n-1}^0+Q_{n-1}^1, & W_{44} &= Q_{n-1}^3+Q_{n-1}^0+3Q_{n-1}^1+3Q_{n-1}^2. \end{aligned}$$

In this equation, Q_{n-1}^i $(i = 0,1,2,3)$ is a square matrix, which is one order lower than the matrix Q_n.

Now we rewrite equation (9) in the usually used form [1,2,3,6].

Assume that the truth-vector $\mathbf{F}$ of $f \in \mathbf{J}$ is split into 4 subvectors of 4^{n-1} successive elements

$$\mathbf{F} = \left\{\mathbf{F}_{[n-1,0]}, \mathbf{F}_{[n-1,1]}, \mathbf{F}_{[n-1,2]}, \mathbf{F}_{[n-1,3]}\right\}.$$

Then, based on (9) RMF polarity matrix $\mathbf{P}_{RMF}$ for quaternary functions can be calculated by recursive method named "recursion by columns", given in Theorem 4.

Theorem 4 *The polarity matrix $\mathbf{P}_{RMF}$ for $f \in \mathbf{J}$ can be calculated as*

$$\mathbf{P}_{RMF} = \mathbf{Q}_n(\mathbf{F}).$$

$\mathbf{Q}_k$, $k = 1,\ldots,n$ *is determined by the following recurrence matrix relations*

$$\mathbf{Q}_k(\mathbf{F}_{[k,j]}) = \begin{bmatrix} \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,0]} & \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,0]} + 3\mathbf{F}_{[k-1,1]}) & W_{13} & W_{14} \\ \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,1]} & \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,1]} + 3\mathbf{F}_{[k-1,2]}) & W_{23} & W_{24} \\ \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,2]} & \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,2]} + 3\mathbf{F}_{[k-1,3]}) & W_{33} & W_{34} \\ \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,3]} & \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,3]} + 3\mathbf{F}_{[k-1,0]}) & W_{43} & W_{44} \end{bmatrix}, \quad (10)$$

where

$$\begin{aligned} W_{13} &= \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,0]} + 2\mathbf{F}_{[k-1,1]} + \mathbf{F}_{[k-1,2]}), \\ W_{14} &= \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,0]} + \mathbf{F}_{[k-1,1]} + 3\mathbf{F}_{[k-1,2]} + 3\mathbf{F}_{[k-1,3]}, \\ W_{23} &= \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,1]} + 2\mathbf{F}_{[k-1,2]} + \mathbf{F}_{[k-1,3]}), \\ W_{24} &= \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,1]} + \mathbf{F}_{[k-1,2]} + 3\mathbf{F}_{[k-1,3]} + 3\mathbf{F}_{[k-1,0]}, \\ W_{33} &= \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,2]} + 2\mathbf{F}_{[k-1,3]} + \mathbf{F}_{[k-1,0]}), \\ W_{34} &= \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,2]} + \mathbf{F}_{[k-1,3]} + 3\mathbf{F}_{[k-1,0]} + 3\mathbf{F}_{[k-1,1]}, \\ W_{43} &= \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,3]} + 2\mathbf{F}_{[k-1,0]} + \mathbf{F}_{[k-1,1]}), \\ W_{44} &= \mathbf{Q}_{k-1}(\mathbf{F}_{[k-1,3]} + \mathbf{F}_{[k-1,0]} + 3\mathbf{F}_{[k-1,1]} + 3\mathbf{F}_{[k-1,2]}. \end{aligned}$$

Proof: *The proof follows from the Kronecker product structure of RMF transform matrix $\mathbf{R}(n)$. Thanks to this structure, the columns of $\mathbf{P}_{RMF}$ can be expressed as the convolution of f with the corresponding columns of $\mathbf{R}(n)$ (Theorem 2). Then, the proof follows from the convolution properties of RMF-expressions [8].*

We define three auxiliary vectors

$$\begin{aligned} \mathbf{T}_{[k-1,1]} &= 3\mathbf{F}_{[k-1,0]}, \\ \mathbf{T}_{[k-1,2]} &= 2\mathbf{F}_{[k-1,2]}, \\ \mathbf{T}_{[k-1,3]} &= 2\mathbf{F}_{[k-1,3]}. \end{aligned}$$

Then, (10) can be written as

$$\mathbf{Q}_k\left(\mathbf{F}_{[k,j]}\right) = \begin{bmatrix} \mathbf{Q}_{k-1}(q_{11}) & \mathbf{Q}_{k-1}(q_{12}) & \mathbf{Q}_{k-1}(q_{13}) & \mathbf{Q}_{k-1}(q_{14}) \\ \mathbf{Q}_{k-1}(q_{21}) & \mathbf{Q}_{k-1}(q_{22}) & \mathbf{Q}_{k-1}(q_{23}) & \mathbf{Q}_{k-1}(q_{24}) \\ \mathbf{Q}_{k-1}(q_{31}) & \mathbf{Q}_{k-1}(q_{32}) & \mathbf{Q}_{k-1}(q_{33}) & \mathbf{Q}_{k-1}(q_{34}) \\ \mathbf{Q}_{k-1}(q_{41}) & \mathbf{Q}_{k-1}(q_{42}) & \mathbf{Q}_{k-1}(q_{43}) & \mathbf{Q}_{k-1}(q_{44}) \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{Q}_{k-1}\left(\mathbf{F}_{[k-1,0]}\right) & \mathbf{Q}_{k-1}\left(q_{22} + q_{13}\right) & W_{13} & W_{14} \\ \mathbf{Q}_{k-1}\left(\mathbf{F}_{[k-1,1]}\right) & \mathbf{Q}_{k-1}\left(q_{32} + q_{23}\right) & W_{23} & W_{24} \\ \mathbf{Q}_{k-1}\left(\mathbf{F}_{[k-1,2]}\right) & \mathbf{Q}_{k-1}\left(q_{42} + q_{33}\right) & W_{33} & W_{34} \\ \mathbf{Q}_{k-1}\left(\mathbf{F}_{[k-1,3]}\right) & \mathbf{Q}_{k-1}\left(q_{41} + \mathbf{T}_{[k-1,1]}\right) & W_{43} & W_{44} \end{bmatrix}, \quad (11)$$

where

$$\begin{aligned} W_{13} &= \mathbf{Q}_{k-1}\left(q_{23} + q_{14}\right), & W_{14} &= \mathbf{Q}_{k-1}\left(q_{23} + q_{33}\right), \\ W_{23} &= \mathbf{Q}_{k-1}\left(q_{21} + q_{41} + \mathbf{T}_{[k-1,2]}\right), & W_{24} &= \mathbf{Q}_{k-1}\left(q_{33} + q_{43}\right), \\ W_{33} &= \mathbf{Q}_{k-1}\left(q_{11} + q_{31} + \mathbf{T}_{[k-1,3]}\right), & W_{34} &= \mathbf{Q}_{k-1}\left(q_{13} + q_{43}\right), \\ W_{43} &= \mathbf{Q}_{k-1}\left(q_{33} + q_{14}\right), & W_{44} &= \mathbf{Q}_{k-1}\left(q_{13} + q_{23}\right). \end{aligned}$$

The number of operations needed to calculate $\mathbf{P}_{RMF}$, is reduced significantly if we first calculate these auxiliary vectors $\mathbf{T}$, and then the vectors that are arguments of the matrices $\mathbf{Q}_{k-1}$, as given in (11).

Recursion by rows

If we know the zero-polarity RMF-coefficient vector $\mathbf{A}$ of f, then the RMF polarity matrix $\mathbf{P}_{RMF}$, can be calculated through the following "recursion by rows" method, given in Theorem 5. This "recursion by rows" can be induced from (9) if we apply the 3-th, 4-th and 5-th step in proposed method for generation recurrence relations for polarity matrices calculation.

Let the vector $\mathbf{A}$ of zero-polarity RMF-coefficients is split into 4 subvectors of 4^{n-1} successive elements, i.e.,

$$\mathbf{A} = \left\{\mathbf{A}_{[n-1,0]}, \mathbf{A}_{[n-1,1]}, \mathbf{A}_{[n-1,2]}, \mathbf{A}_{[n-1,3]}\right\}.$$

Theorem 5 *The RMF polarity matrix* $\mathbf{P}_{RMF}$ *for* $f \in \mathbf{J}$ *can be calculated as*

$$\mathbf{P}_{RMF} = \mathbf{Q}_n(\mathbf{A}),$$

where the following recursive matrix relations are used for the calculation of $\mathbf{Q}_k,\ k = 1, \ldots, n:$

$$\mathbf{Q}_k\left(\mathbf{A}_{[k,j]}\right) = \begin{bmatrix} \mathbf{Q}_{k-1}(t_{11}) & \mathbf{Q}_{k-1}(t_{12}) & \mathbf{Q}_{k-1}(t_{13}) & \mathbf{Q}_{k-1}(t_{14}) \\ \mathbf{Q}_{k-1}(t_{21}) & \mathbf{Q}_{k-1}(t_{22}) & \mathbf{Q}_{k-1}(t_{23}) & \mathbf{Q}_{k-1}(t_{24}) \\ \mathbf{Q}_{k-1}(t_{31}) & \mathbf{Q}_{k-1}(t_{32}) & \mathbf{Q}_{k-1}(t_{33}) & \mathbf{Q}_{k-1}(t_{34}) \\ \mathbf{Q}_{k-1}(t_{41}) & \mathbf{Q}_{k-1}(t_{42}) & \mathbf{Q}_{k-1}(t_{43}) & \mathbf{Q}_{k-1}(t_{44}) \end{bmatrix} \tag{12}$$

where

$t_{11} = \mathbf{A}_{[k-1,0]}$; $\quad t_{12} = \mathbf{A}_{[k-1,1]}$,
$t_{13} = \mathbf{A}_{[k-1,2]}$, $\quad t_{14} = \mathbf{A}_{[k-1,3]}$,
$t_{21} = \mathbf{A}_{[k-1,0]} + 3\mathbf{A}_{[k-1,1]}$, $\quad t_{22} = \mathbf{A}_{[k-1,1]} + 3\mathbf{A}_{[k-1,2]}$,
$t_{23} = \mathbf{A}_{[k-1,2]} + 3\mathbf{A}_{[k-1,3]}$, $\quad t_{24} = 2\mathbf{A}_{[k-1,2]} + \mathbf{A}_{[k-1,3]}$,
$t_{31} = \mathbf{A}_{[k-1,0]} + 2\mathbf{A}_{[k-1,1]} + \mathbf{A}_{[k-1,2]}$, $\quad t_{32} = \mathbf{A}_{[k-1,1]} + 2\mathbf{A}_{[k-1,2]} + \mathbf{A}_{[k-1,3]}$,
$t_{33} = 3\mathbf{A}_{[k-1,2]} + 2\mathbf{A}_{[k-1,3]}$, $\quad t_{34} = 3\mathbf{A}_{[k-1,3]}$,
$t_{41} = \mathbf{A}_{[k-1,0]} + \mathbf{A}_{[k-1,1]} + 3\mathbf{A}_{[k-1,2]} + 3\mathbf{A}_{[k-1,3]}$, $\quad t_{42} = \mathbf{A}_{[k-1,1]} + 3\mathbf{A}_{[k-1,2]} + 3\mathbf{A}_{[k-1,3]}$,
$t_{43} = 3\mathbf{A}_{[k-1,2]} + 3\mathbf{A}_{[k-1,3]}$, $\quad t_{44} = 2\mathbf{A}_{[k-1,2]} + 3\mathbf{A}_{[k-1,3]}$.

$$\mathbf{Q}_k(\mathbf{A}_{[0,j]}) = \mathbf{A}_{[0,j]},\ j = 0, 1, 2, 3.$$

Similarly to the "recursion by columns" method, the number of additions and multiplications can be reduced significantly if, instead (12), we use the following formula

$$\mathbf{Q}_k\left(\mathbf{A}_{[k,j]}\right) = \begin{bmatrix} \mathbf{Q}_{k-1}(q_{11}) & \mathbf{Q}_{k-1}(q_{12}) & \mathbf{Q}_{k-1}(q_{13}) & \mathbf{Q}_{k-1}(q_{14}) \\ \mathbf{Q}_{k-1}(q_{21}) & \mathbf{Q}_{k-1}(q_{22}) & \mathbf{Q}_{k-1}(q_{23}) & \mathbf{Q}_{k-1}(q_{24}) \\ \mathbf{Q}_{k-1}(q_{31}) & \mathbf{Q}_{k-1}(q_{32}) & \mathbf{Q}_{k-1}(q_{33}) & \mathbf{Q}_{k-1}(q_{34}) \\ \mathbf{Q}_{k-1}(q_{41}) & \mathbf{Q}_{k-1}(q_{42}) & \mathbf{Q}_{k-1}(q_{43}) & \mathbf{Q}_{k-1}(q_{44}) \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{Q}_{k-1}\left(\mathbf{A}_{[k-1,0]}\right) & \mathbf{Q}_{k-1}\left(\mathbf{A}_{[k-1,1]}\right) & W_{13} & W_{14} \\ \mathbf{Q}_{k-1}\left(q_{11} + \mathbf{S}_{[k-1,1]}\right) & \mathbf{Q}_{k-1}\left(q_{42} + q_{14}\right) & W_{23} & W_{24} \\ \mathbf{Q}_{k-1}\left(q_{21} + q_{13} + \mathbf{S}_{[k-1,1]}\right) & \mathbf{Q}_{k-1}\left(q_{24} + q_{12}\right) & W_{33} & W_{34} \\ \mathbf{Q}_{k-1}\left(q_{11} + q_{42}\right) & \mathbf{Q}_{k-1}\left(q_{24} + \mathbf{S}_{[k-1,3]}\right) & W_{43} & W_{44} \end{bmatrix}.$$

where

$$\begin{aligned} W_{13} &= \mathbf{Q}_{k-1}\left(\mathbf{A}_{[k-1,2]}\right), & W_{14} &= \mathbf{Q}_{k-1}\left(\mathbf{A}_{[k-1,3]}\right), \\ W_{23} &= \mathbf{Q}_{k-1}\left(q_{13} + \mathbf{S}_{[k-1,3]}\right), & W_{24} &= \mathbf{Q}_{k-1}\left(q_{14} + \mathbf{S}_{[k-1,2]}\right), \\ W_{33} &= \mathbf{Q}_{k-1}\left(q_{23} + q_{44}\right), & W_{34} &= \mathbf{Q}_{k-1}\left(\mathbf{S}_{[k-1,3]}\right), \\ W_{43} &= \mathbf{Q}_{k-1}\left(q_{13} + q_{44}\right), & W_{44} &= \mathbf{Q}_{k-1}\left(\mathbf{S}_{[k-1,2]} + \mathbf{S}_{[k-1,3]}\right), \end{aligned}$$

$$\begin{aligned} \mathbf{S}_{[k-1,1]} &= 3\mathbf{A}_{[k-1,1]}, \\ \mathbf{S}_{[k-1,2]} &= 2\mathbf{A}_{[k-1,2]}, \\ \mathbf{S}_{[k-1,3]} &= 3\mathbf{A}_{[k-1,3]}. \end{aligned}$$

Calculation of the auxiliary vectors $\mathbf{S}$ precedes calculation of arguments in $\mathbf{Q}_{k-1}$ like in the previous method.

9 Calculation Complexity

In this section, the efficiency of the presented methods for calculation of RMF polarity matrix is estimated through the number of operations required to calculate $\mathbf{P}_{RMF}$ for a quaternary function. For comparison, we give the number of operations in the corresponding methods for GF-expressions.

There are few methods to calculate the polarity matrix for GF-expressions of quaternary functions. A direct calculation by definition of $\mathbf{P}_{GF}$ for GF-expressions of n-variable quaternary functions requires $11^n - 4^n$ additions and $\frac{2}{3}(11^n - 5^n)$ multiplications [5]. In FFT-like algorithms proposed in [5], the number of additions and multiplications is $7n4^{n-1}$ and $n4^n$, respectively. The recursive algorithm proposed by Falkowski and Rahardja in [3] requires $A_n = \frac{4}{3}(13^n - 4^n)$ additions and $M_n = \frac{3}{8}(3^n - 1)4^n$ multiplications.

By the analogy to GF-expressions, we considered few ways to calculate the RMF polarity matrix. In a direct implementation of (6), the number of required additions and multiplications is $A_d^n = 4^{2n}(4^n - 1)$ and $M_d^n = 4^{3n}$, respectively. The number of additions and multiplications required for the polarity matrix calculation with FFT-like algorithm is $A_{FFT}^n = \frac{3n}{2}16^n$ and $M_{FFT}^n = \frac{7n}{4}16^n$, respectively.

The computational cost of methods proposed in Section 8 is stated by the following theorem.

Table 5: The number of additions and multiplications in calculation of $\mathbf{P}_{RMF}$.

	direct		FFT-like	
n	A_d^n	M_d^n	A_{FFT}^n	M_{FFT}^n
1	48	64	24	28
2	3840	4096	768	896
3	258048	262144	18432	21504
4	16711680	16777216	393216	458752
5	1072693248	1073741824	7864320	9175040
6	6.87027e10	6.8719477E10	150994944	176160768
	recursion by columns		recursion by rows	
n	A_c^n	M_c^n	A_r^n	M_r^n
1	14	3	12	3
2	280	57	240	57
3	4704	903	4032	903
4	76160	13737	65280	13737
5	1222144	206823	1047552	206823
6	19568640	3105417	16773120	3105417

Theorem 6 *The number of additions required to calculate RMF polarity matrix for an n-variable quaternary function, by using the recursive matrix relation (11) (recursion by columns) is* $A_c^n = \frac{14}{12}(16^n - 4^n)$. *If the relation (13) (recursion by rows) is used, the number of additions is* $A_r^n = (16^n - 4^n)$. *In both cases, the same number of* $M_c^n = \frac{3}{11}(15^n - 4^n)$ *multiplications is required.*

For illustration of this theorem, the Table 5 shows the number of additions and multiplications in calculation of the RMF-polarity matrix for different values of the number n of variables for different methods. Figures 1 and 2 show the number of additions and the number of multiplications needed for the calculation of the RMF polarity matrix with different methods.

It is obvious that methods proposed in Section 8 are more efficient than direct computation or FFT-like methods for the calculation of the RMF polarity matrices. It is important to note that the efficiency of our method increases with the number of variables.

10 Conclusion

We have proposed a method for construction of recursive procedures for the polarity matrices calculation in polynomial logical function representation. As particular cases the recursive methods proposed in [3] and [6] can be derived by our method. Based on our method we have constructed two algorithms, denoted as "recursion by rows" and "recursion by columns", for generation of polarity matrices for RMF-expressions of quaternary functions. To estimate their efficiency, we determined the number of operations required in each of them, and provided a comparison to other algorithms for generation of $\mathbf{P}_{RMF}$, as well as to the corresponding algorithms for polarity matrix for GF-expressions.

We showed that the proposed algorithms are more efficient than both direct calculation of $\mathbf{P}_{RMF}$ and related FFT-like algorithms. An important feature is that the efficiency of the proposed algorithms grows with the number of variables n in the represented functions. For example, the ratio between the number of additions in direct calculation of $\mathbf{P}_{RMF}$ and "recursion by rows" method is 4^n. The corresponding ratio for multiplications is greater than $\frac{11}{3}4^n$.

Our method can be used for construction of recursive relations for polarity matrices calculation for any Kronecker product based expression of MV functions.

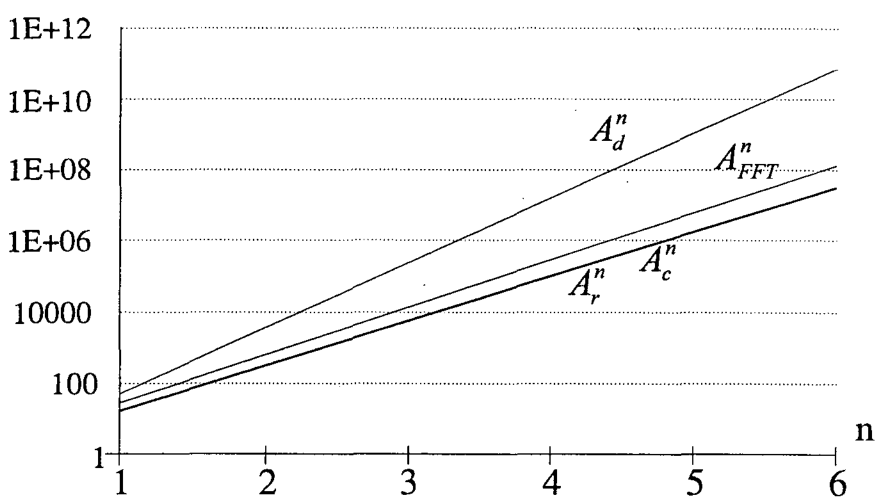

Figure 1: The number of additions needed for calculation $\mathbf{P}_{RMF}$.

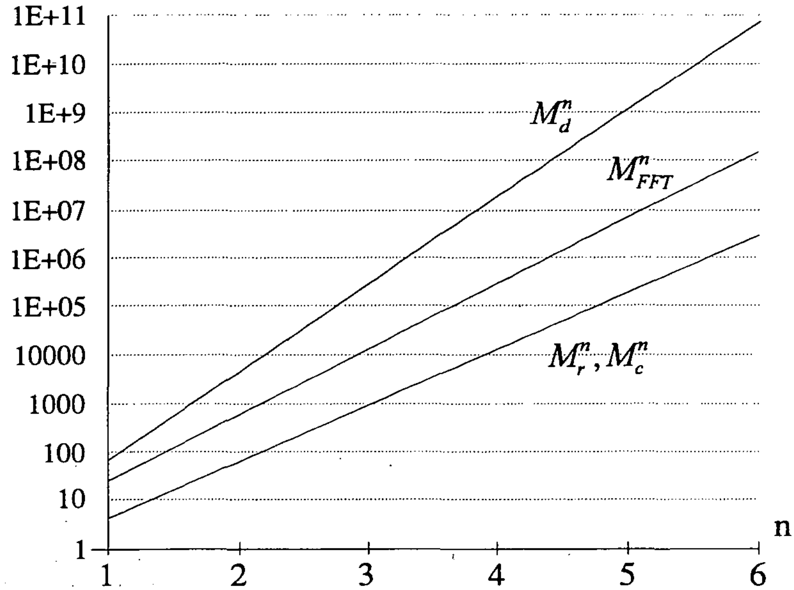

Figure 2: The number of multiplications needed for calculation $\mathbf{P}_{RMF}$.

References

[1] Falkowski, B.J., Rahardja, S., "Efficient algorithm for the generation of fixed polarity quaternary Reed-Muller expansions", *Proc. 25-th Int. Symp. on Multiple-Valued Logic*, Bloomington, 1995, 158-163.

[2] Falkowski, B.J., Rahardja, S., "Efficient computation of quaternary fixed polarity Reed-Muller expansions", *IEE Proc.-Comp.Digit.Tech.*, Vol.142, No. 5, 1995, 345-352.

[3] Falkowski, B.J., Rahardja, S., "Fast construction of polarity coefficient matrices for fixed polarity quaternary Reed-Muller expansions",*Proc. 5th International Workshop on Spectral Techniques*, Beijing, China, March 1994, 220-225.

[4] Falkowski, B.J., Rahardja, S., "Quasi-arithmetic expansions for quaternary functions",*Proc. IFIP WG 10.5 Workshop on Applications of the Reed-Muller Expansion in Circuit Design*, Chiba, Japan, 1995, 265-272.

[5] Green, D.H., "Reed-Muller expansions with fixed and mixed polarities over GF(4)", *IEE Proc.- Comp.Digit.Tech.*, Vol.137, No. 5, Sept. 1990, 380-388.

[6] Harking, B., Moraga, C., "Efficient derivation of Reed-Muller expansions in multiple-valued logic system", *Proc. 22nd IEEE Int. Symp. on Multiple-Valued Logic*, Sendai, Japan, 1992, 436-441.

[7] Muzio, J.C., Wesselkamper, T.C., *Multiple-valued Switching Theory*, Adam Hilger, Bristol, 1986.

[8] Stanković, R.S., Moraga, C., "Reed-Muller-Fourier representations of multiple-valued functions over Galois fields of prime cardinality",*Proc. IFIP WG 10.5 Workshop on Applications of the Reed-Muller Expansion in Circuit Design*, Hamburg, Germany, Sept. 1993, 115-124.

[9] Stanković, R..S., Janković, D., Moraga, C.,"Reed-Muller-Fourier versus Galois Field representations of Four- Valued Logic Functions",*Proc. 3rd Workshop on Applications of the Reed-Muller Expansion in Circuit Design*, September 19-20, 1997. Oxford, UK, 269-278.