Efficient algorithm for canonical Reed-Muller expansions of Boolean functions

B. Harking

Indexing terms: Boolean algebra, Matrix algebra

Abstract: The paper presents a new method for computing all 2^n canonical Reed-Muller forms (RMC forms) of a Boolean function. The method constructs the coefficients directly and no matrix-multiplication is needed. It is also usable for incompletely specified functions and for calculating a single RMC form. The method exhibits a high degree of parallelism.

1 Introduction

It is well known that every Boolean function can be expressed using XOR and AND gates. This representation has various advantages over the conventional description. On the one hand the representations may be shorter and therefore lead to smaller circuits on a chip. On the other hand they are much easier to test [2]. The main problem in this field is to minimise the number of product-terms in a mod-2-sum of products.

Generally two forms are distinguished in literature. First there are general mod-2-sum of products, i.e. there are no limitations on the variables or the form of the product-terms. Algorithms dealing with these forms can be found in References 9–11. The second form is the representation as a Reed-Muller-canonical (RMC) form [3–8]. Here every variable appears either complemented or uncomplemented, but not in both forms. (For a more detailed description see Section 2). The advantage of this representation is the fact that the resulting circuit needs at most n inputs in contrast to up to $2n$ inputs in other cases. The second essential advantage is the fact that for each function represented in RMC form there exists a circuit, which can be tested with maximum $3n + 4$ tests, most of them independent of the realised function [2].

It is easy to see that for a Boolean function with n variables there exist 2^n different RMC forms. Each of these forms can be characterised by 2^n Boolean values a_i indicating the presence or the absence of a given product-term. The aim is now to find the RMC form with the least number of $a_i = 1$. Algorithms existing up to now build up a $2^n \times 2^n$ matrix, called polarity-matrix [7], where every coefficient a_i of each of the 2^n polarities is given. This polarity-matrix is constructed using matrix-multiplication [3], which means that these algorithms belong to the class with complexity $AT^2 = O(16^n)$ [13].

The procedure presented here constructs the polarity-matrix directly out of the 2^n coefficients of the disjunctive normal form. It will be shown that this algorithm has a complexity $AT^2 = n^2 \cdot 3^{n-1}$. The procedure can also be used in case of incompletely specified functions and for constructing the coefficients of a single RMC form.

Paper 7316E (C2), received 8th August 1989

The author is at Universität Dortmund, Abteilung Informatik, Lehrstuhl, August-Schmidt Strasse 12, Postfach 500 500, 4600 Dortmund 50, West Germany

2 The problem

In this paper the following notations are used:

A Boolean function f is a mapping f: $\{0, 1\}^n \rightarrow \{0, 1\}$.

$\dot{x}_i$ stands for x_i or $\bar{x}_i$.

$p = \prod_{i=1}^{k} \dot{x}_i$ is a product term; if $k = n$, p will be called a minterm.

$$\dot{x}_i^{e_i} = \begin{cases} 1 & \text{if } e_i = 0 \\ \dot{x}_i & \text{if } e_i = 1 \end{cases}$$

$\cdot_2$ stands for a matrix multiplication, where the addition is carried out mod-2.

A number $i \in \mathbb{N}$, $0 \leqslant i \leqslant 2^n - 1$, is used both in decimal and in binary form $(i_1, \ldots, i_n)$, $i_j \in \{0, 1\}$.

The following theorem shows that every Boolean function can be described using only XOR and AND gates.

Theorem 1: Every Boolean function can be written as:

$$f = \oplus \sum_{j=0}^{2^n-1} a_j \cdot \dot{x}_1^{j_1} \cdot \cdots \cdot \dot{x}_n^{j_n} \quad a_j \in \{0, 1\} \tag{1}$$

The vector $(\bar{\dot{x}}_1, \ldots, \bar{\dot{x}}_n)$ will be called the polarity k, $0 \leqslant k \leqslant 2^n - 1$, of the RMC form of f.

Proof: see Reference 1

Example 1: Let the Boolean function $f(x_1, x_2, x_3)$ be given as:

$$f = \bar{x}_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 x_3$$

or as:

$$\begin{aligned} f = {} & f_0 \bar{x}_1 \bar{x}_2 \bar{x}_3 + f_1 \bar{x}_1 \bar{x}_2 x_3 + f_2 \bar{x}_1 x_2 \bar{x}_3 \\ & + f_3 \bar{x}_1 x_2 x_3 + f_4 x_1 \bar{x}_2 \bar{x}_3 + f_5 x_1 \bar{x}_2 x_3 \\ & + f_6 x_1 x_2 \bar{x}_3 + f_7 x_1 x_2 x_3 \end{aligned}$$

with $f_0 = f_1 = f_3 = f_6 = 0$ and $f_2 = f_4 = f_5 = f_7 = 1$. In the following a Boolean function f will be described by its minterm-vector $[f] = [f_0, \ldots, f_{2^n-1}]$; in this case

$$[f] = [0\ 0\ 1\ 0\ 1\ 1\ 0\ 1]$$

Theorem 1 now states that for every given polarity k, $0 \leqslant k \leqslant 2^n - 1$, f can be written using only XOR and AND. Let k be 0; i.e. $\dot{x}_1 = x_1, \dot{x}_2 = x_2, \dot{x}_3 = x_3$.

$$\oplus \sum_{j=0}^{7} a_j \cdot x_1^{j_1} \cdot x_2^{j_2} \cdot x_3^{j_n}$$
$$= a_0 x_1^0 x_2^0 x_3^0 \oplus a_1 x_1^0 x_2^0 x_3^1 \oplus a_2 x_1^0 x_2^1 x_3^0 \oplus a_3 x_1^0 x_2^1 x_3^1 \oplus a_4 x_1^1 x_2^0 x_3^0 \oplus a_5 x_1^1 x_2^0 x_3^1 \oplus a_6 x_1^1 x_2^1 x_3^0 \oplus a_7 x_1^1 x_2^1 x_3^1$$
$$= a_0 \oplus a_1 x_3 \oplus a_2 x_2 \oplus a_3 x_2 x_3 \oplus a_4 x_1 \oplus a_5 x_1 x_3 \oplus a_6 x_1 x_2 \oplus a_7 x_1 x_2 x_3$$

If now a_0, a_1, a_5, a_6, a_7 are set to 0 and a_2, a_3, a_4 are set to 1 the example function f is described in terms of XOR and AND as:

$$f = x_2 \oplus x_2 x_3 \oplus x_1$$

In the following, RMC forms will also be characterised by their polarity and the corresponding coefficient-vector $[a] = [a_0, \ldots, a_{2^n-1}]$. For the solution given above we have $[a] = [0\ 0\ 1\ 1\ 1\ 0\ 0\ 0]$.

If the polarity is changed to, for example, $k = 5$ we obtain:

$$\oplus \sum_{j=0}^{7} a_j \cdot \bar{x}_1^{j_1} \cdot x_2^{j_2} \cdot \bar{x}_3^{j_n}$$
$$= a_0 \oplus a_1 \bar{x}_3 \oplus a_2 x_2 \oplus a_3 x_2 \bar{x}_3 \oplus a_4 \bar{x}_1 \oplus a_5 \bar{x}_1 \bar{x}_3 \oplus a_6 \bar{x}_1 x_2 \oplus a_7 \bar{x}_1 x_2 \bar{x}_3$$

Now f is given by: $[a] = [1\ 0\ 0\ 1\ 1\ 0\ 0\ 0]$. $f = 1 \oplus x_2 \bar{x}_3 \oplus \bar{x}_1$.

It is obvious, that for every function f in n variables there exist 2^n RMC forms. With the polarity k and the related a_i, $0 \leqslant i \leqslant 2^n - 1$, the function is described completely. The problem is now to determine the polarity for a given function f, where most of the a_i are 0, i.e. where f has the shortest representation. For instance for the function f of the example, the minimal solution is obtained with polarity 1, because then $[a] = [0\ 0\ 0\ 1\ 1\ 0\ 0\ 0]$ and $f = x_1 \oplus x_2 \bar{x}_3$.

For the case of functions with up to four variables it is certainly possible to obtain the RMC forms by inspection of the Karnaugh-map of the function, recalling that now 1-minterms are covered an odd number of times meanwhile 0-minterms either remain uncovered or are covered an even number of times.

Consider the three cases discussed in the example above. The maps of the corresponding solutions are shown in Fig. 1. It becomes apparent that for a larger number of arguments this kind of analysis is no longer applicable. A general method is given in the next Section.

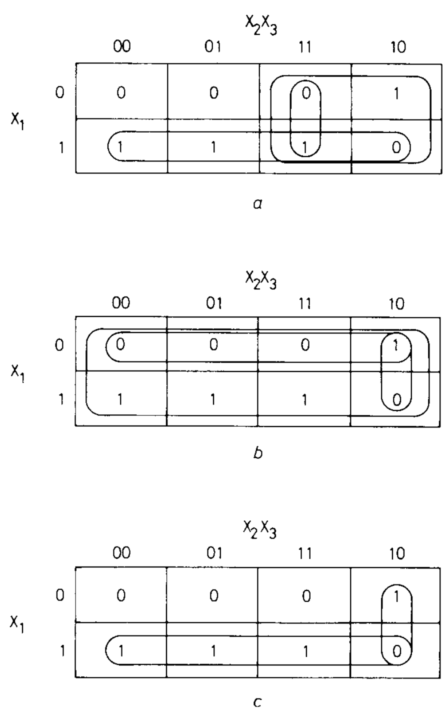

Fig. 1 *Map-analysis of three RMC forms of the function in example 1*
a $K = 0, f = x_1 \oplus x_2 \oplus x_2 x_3$
b $K = 5, f = 1 \oplus \bar{x}_1 \oplus x_2 \bar{x}_3$
c $K = 1, f = \bar{x}_1 \oplus x_2 \bar{x}_3$

3 Method

Definition: Let T^n be a $2^n \times 2^n$ binary matrix. T^n will recursively be defined as:

$$T^n = \begin{bmatrix} T^{n-1} & 0 \\ T^{n-1} & T^{n-1} \end{bmatrix} \quad \text{and} \quad T^0 = [1] \tag{2}$$

It becomes apparent that the same matrix can be obtained by the nth Kronecker-power [12] of T^1.

Theorem 2: Let a Boolean function f in n variables be given as $[f] = [f_0, \ldots, f_{2^n-1}]$. The coefficients $[a] = [a_0, \ldots, a_{2^n-1}]$ of the RMC form with polarity 0 can be computed by:

$$[a]^T = T^n \cdot_2 [f]^T \tag{3}$$

Proof: [3, 6, 8]

Example 2: Let f be given as: $[f] = [0\ 0\ 1\ 0\ 1\ 1\ 0\ 1]$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix} = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 1&1&0&0&0&0&0&0 \\ 1&0&1&0&0&0&0&0 \\ 1&1&1&1&0&0&0&0 \\ 1&0&0&0&1&0&0&0 \\ 1&1&0&0&1&1&0&0 \\ 1&0&1&0&1&0&1&0 \\ 1&1&1&1&1&1&1&1 \end{bmatrix} \cdot_2 \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The RMC form of f with polarity 0 is then: $f = x_2 \oplus x_2 x_3 \oplus x_1$.

Definition: The polarity-matrix of f is a $2^n \times 2^n$ binary matrix $M[f]$, where every element m_{ij} (row i, column j) corresponds to the coefficient a_j of the RMC form of f with polarity i.

The optimum polarity of a given function f is the row i of the polarity-matrix of f with the minimum number of nonzero coefficients a_{ij}.

Definition: A Boolean function f is given as $[f] = [f', f'']$, with $[f'] = [f_0, \ldots, f_{2^{n-1}-1}]$ and $[f''] = [f_{2^{n-1}}, \ldots, f_{2^n-1}]$. Then the $2^n \times 2^n$ matrix $B[f]$ is defined as:

$$B[f] = \begin{bmatrix} B[f'] & B[f''] \\ B[f''] & B[f'] \end{bmatrix} \quad \text{and} \quad B[f_i] = f_i \tag{4}$$

Theorem 3:
Let Z^n be the nth Kronecker-power of

$$Z = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

Then:

$$M[f] = B[f] \cdot_2 Z^n. \tag{5}$$

Proof: By induction
Beginning: $n = 1$

$$f = f_0 \bar{x}_1 \oplus f_1 x_1$$

RMC form with polarity 0:

$$f = f_0 \oplus (f_0 \oplus f_1) x_1$$

RMC form with polarity 1:

$$f = f_1 \oplus (f_0 \oplus f_1) \bar{x}_1$$

It follows:

$$M[f] = \begin{bmatrix} f_0 & f_0 \oplus f_1 \\ f_1 & f_0 \oplus f_1 \end{bmatrix}$$

Since $[f] = [f_0 \quad f_1]$

$$B[f] \cdot_2 Z = \begin{bmatrix} f_0 & f_1 \\ f_1 & f_0 \end{bmatrix} \cdot_2 \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} f_0 & f_0 \oplus f_1 \\ f_1 & f_0 \oplus f_1 \end{bmatrix} = M[f]$$

Conclusion: Let the theorem be proved for functions up to n variables. Let:

$$[f'] = [f_0, \ldots, f_{2^n - 1}]^T$$
$$[f''] = [f_{2^n - 1}, \ldots, f_{2^{n+1} - 1}]^T$$
$$C^n = B[f'] \cdot_2 Z^n$$
$$D^n = B[f''] \cdot_2 Z^n$$

It holds:

$$\begin{aligned} f &= \bar{x}_0 (f_0 \bar{x}_1 \cdots \bar{x}_n + \cdots + f_{2^n - 1} x_1 \cdots x_n) \\ &\quad + x_0 (f_{2^n} \bar{x}_1 \cdots \bar{x}_n + \cdots + f_{2^{n+1} - 1} x_1 \cdots x_n) \\ &= \bar{x}_0 (b_0 \oplus b_1 \dot{x}_n \oplus \cdots \oplus b_{2^n - 1} \dot{x}_1 \cdots \dot{x}_n) \\ &\quad \oplus x_0 (c_0 \oplus c_1 \dot{x}_n \oplus \cdots \oplus c_{2^n - 1} \dot{x}_1 \cdots \dot{x}_n) \end{aligned}$$

The coefficients b_i and c_i are obtained from the corresponding rows of C^n and D^n. Hence, for polarities $0 \leqslant k < 2^n$ we have:

$$\begin{aligned} f &= (b_0 \oplus b_1 \dot{x}_n \oplus \cdots \oplus b_{2^n - 1} \dot{x}_1 \cdots \dot{x}_n) \\ &\quad \oplus ((b_0 \oplus c_0) x_0 \oplus (b_1 \oplus c_1) x_0 \dot{x}_n \\ &\quad \oplus \cdots \oplus (b_{2^n - 1} \oplus c_{2^n - 1}) x_0 \dot{x}_1 \cdots \dot{x}_n) \end{aligned}$$

and for polarities $2^n \leqslant k < 2^{n+1}$:

$$\begin{aligned} f &= (c_0 \oplus c_1 \dot{x}_n \oplus \cdots \oplus c_{2^n - 1} \dot{x}_1 \cdots \dot{x}_n) \\ &\quad \oplus ((b_0 \oplus c_0) \bar{x}_0 \oplus (b_1 \oplus c_1) \bar{x}_0 \dot{x}_n \\ &\quad \oplus \cdots \oplus (b_{2^n - 1} \oplus c_{2^n - 1}) \bar{x}_0 \dot{x}_1 \cdots \dot{x}_n) \end{aligned}$$

Summarised:

$$M[f] = \begin{bmatrix} C^n & C^n \oplus D^n \\ D^n & C^n \oplus D^n \end{bmatrix}$$

Here '$\oplus$' denotes componentwise mod-2 addition of the two $2^n \times 2^n$ matrices.

$$\begin{aligned} M[f] &= \begin{bmatrix} C^n & C^n \oplus D^n \\ D^n & C^n \oplus D^n \end{bmatrix} \\ &= \begin{bmatrix} B[f'] & B[f''] \\ B[f''] & B[f'] \end{bmatrix} \cdot_2 \begin{bmatrix} Z^n & Z^n \\ 0 & Z^n \end{bmatrix} \\ &= B[f] Z^{n+1} \end{aligned}$$

q.e.d.

A similar version of this theorem also appears in Reference 3, but without proof.

Example 3: Let f be as in example 1. $B[f] \cdot_2 Z^3 = M[f]$

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\cdot_2 \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

It is simple to see that the second row of $M[f]$ has the least nonzero coefficients. Hence, the optimum polarity is $k = 1$ with $f = x_2 \bar{x}_3 \oplus x_1$.

Definition: A Boolean function is given as $[f] = [f_0, \ldots, f_{2^n - 1}]$. Let

$$\begin{aligned} [f'] &= [f_0, \ldots, f_{2^{n-1} - 1}] \\ [f''] &= [f_{2^{n-1}}, \ldots, f_{2^n - 1}] \\ [f'''] &= [f_0 \oplus f_{2^{n-1}}, \ldots, f_{2^{n-1} - 1} \oplus f_{2^n - 1}] \end{aligned} \tag{6}$$

The $2^n \times 2^n$ matrix $P[f]$ is defined as follows:

$$P[f] = \begin{bmatrix} P[f'] & P[f'''] \\ P[f''] & P[f'''] \end{bmatrix} \quad \text{and} \quad P[f_i] = f_i \tag{7}$$

It can be seen that P is an automorphism, because:

$$P[f'] \oplus P[f''] = P[f' \oplus f''] = P[f'''] \tag{8}$$

Theorem 4:

$$P[f] = M[f] \tag{9}$$

Proof: By induction
Beginning: $n = 1$

$$P[f_0, f_1] = \begin{bmatrix} f_0 & f_0 \oplus f_1 \\ f_1 & f_0 \oplus f_1 \end{bmatrix} = M[f]$$

see also proof of theorem 3.

Conclusion: Let the theorem be proved for functions with up to n variables. It follows from theorem 3:

$$P[f'] = B[f_0, \ldots, f_{2^n - 1}] \cdot_2 Z^n = B[f'] \cdot_2 Z^n$$

$$P[f''] = B[f_{2^n}, \ldots, f_{2^{n+1} - 1}] \cdot_2 Z^n = B[f''] \cdot_2 Z^n$$

Furthermore:

$$M[f] = \begin{bmatrix} B[f'] \cdot_2 Z^n & B[f'] \cdot_2 Z^n \oplus B[f''] \cdot_2 Z^n \\ B[f''] \cdot_2 Z^n & B[f'] \cdot_2 Z^n \oplus B[f''] \cdot_2 Z^n \end{bmatrix}$$

$$= \begin{bmatrix} P[f'] & P[f'] \oplus P[f''] \\ P[f''] & P[f'] \oplus P[f''] \end{bmatrix}$$

Here '$\oplus$' denotes again componentwise mode-2 addition of the two $2^n \times 2^n$ matrices. It follows:

$$M[f] = \begin{bmatrix} P[f'] & P[f'''] \\ P[f''] & P[f'''] \end{bmatrix} = P[f] \qquad \text{q.e.d.}$$

Example 4: Let f be as in example 1.

$$P[f] = \begin{bmatrix} P[0\ 0\ 1\ 0] & P[1\ 1\ 1\ 1] \\ P[1\ 1\ 0\ 1] & P[1\ 1\ 1\ 1] \end{bmatrix}$$

$$= \begin{bmatrix} P[0\ \ 0] & P[1\ \ 0] & P[1\ \ 1] & P[0\ \ 0] \\ P[1\ \ 0] & P[1\ \ 0] & P[1\ \ 1] & P[0\ \ 0] \\ P[1\ \ 1] & P[1\ \ 0] & P[1\ \ 1] & P[0\ \ 0] \\ P[0\ \ 1] & P[1\ \ 0] & P[1\ \ 1] & P[0\ \ 0] \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Notice that computation of the coefficients of the RMC form of f for a given polarity is possible without constructing the whole matrix P. This can be advantageous in cases of lack of storage.

Example 5: Let f be as in example 1 and let the selected polarity be 3. The fourth row of $P[f]$ is the fourth row of the submatrix

$$[P[0\ \ 0\ \ 1\ \ 0]\ \ P[1\ \ 1\ \ 1\ \ 1]]$$

which is the second row of

$$[P[1\ \ 0]\ \ P[1\ \ 0]\ \ P[1\ \ 1]\ \ P[0\ \ 0]]$$

which is $[0\ 1\ 0\ 1\ 1\ 0\ 0\ 0]$.

The following theorem refers to the so called area-time tradeoff AT^2 of constructing $P[f]$. This is a parameter for estimating the time T and chip area A used by a given procedure. With this parameter it is especially possible to estimate the complexity of algorithms with a high degree of parallelism.

Theorem 5: The area-time tradeoff for constructing $P[f]$ is:

$$AT^2 = n^2 \cdot 3^{n-1} \tag{10}$$

Proof: It will be shown, that 3^{n-1} processors are needed for constructing $P[f]$ within n steps. $P[f]$ can be built up (see definition) recursively by constructing successively $P[f']$, $P[f'']$ and $P[f''']$, which themselves are computed by the same scheme. This procedure needs n steps.

Step 1: The 2^{n-1} values $f_0 \oplus f_{2^{n-1}}, \ldots, f_{2^{n-1}-1} \oplus f_{2^n - 1}$ of $[f''']$ have to be computed.

Step 2: For each of the three submatrices $P[f']$, $P[f'']$ and $P[f''']$ 2^{n-2} values have to be computed, altogether $3 \cdot 2^{n-2}$ values.

Step 3: At the next iteration step $3^2 \cdot 2^{n-3}$ have to be computed.

Step i: $3^{i-1} \cdot 2^{n-i}$ values have to be computed.

It follows that for constructing $P[f]$ in n steps, $\max_{1 \leq i \leq n} 3^{i-1} \cdot 2^{n-i} = 3^{n-1}$ processors are needed.

For constructing the polarity-matrix of a function f in n variables following theorem 3 the area-time tradeoff is [13]

$$AT^2 = O(16^n). \tag{11}$$

4 Incompletely specified functions

In this Section an approach for computing the polarity-matrix of an incompletely specified function using the method of Section 3 is presented. Let an incompletely specified function be given as: $[f] = [f_0, \ldots, f_{2^n - 1}]$ with $f_i \in \{0, 1, d_i\}$.

Step 1: Replace in $[f]$ all d_i by 0 and build up $P[f]$. The polarity-matrix of f with all don't care terms set to 0 is obtained.

Step 2: Replace in $[f]$ all $f_i = 1$ by 0 and build up $B[f]$. An entry $b_{ij} = d_k$ characterises the coefficient of the RMC form of f with polarity i, which has the lowest column index among all coefficients influenced by the choice of d_k.

Step 3: Let the entry b_{ij} of the matrix $B[f]$ built up in step 2 be d_k. If now d_k is set to 1 for the RMC form of polarity i, all those values p_{is} of $P[f]$ have to be complemented, where for the column index $(s_1 \ldots s_n)$ the following equation holds:

$$(s_1 \cdot j_1, \ldots, s_n \cdot j_n) = (j_1, \ldots, j_n) \tag{12}$$

At this point several strategies for choice of the don't-care terms $d_1, \ldots, d_k$ can be formulated:

(*a*) Compute $P[f]$ for all 2^k possible coverings of the don't-care terms [3]. Obviously the best solution is received, but the time needed for great k increases exponentially.

(*b*) Let be $b_{ij} = d_k$ and all b_{ir} with $r < j$ equal to 0. If $p_{ij} = 1$, set d_k to 1 and complement all coefficients p_{is} affected by choice of d_k [7].

(*c*) Let be $b_{ij} = d_k$ and all b_{ir} with $r < j$ equal to 0. Inspect all by choice of d_k affected coefficients p_{is}. If $|\{p_{is}: p_{is} = 1\}| > |\{p_{is}: p_{is} = 0\}|$, set d_k to 1 and complement all p_{is}.

The heuristic procedures (*b*) and (*c*) have the advantage of being computed easily, but they do not yield the optimum result in every case.

Example 6: Let the function f be given as:

$$[f] = [1\ \ d_1\ \ 0\ \ 1\ \ d_4\ \ 1\ \ 0\ \ 1]$$

First the matrices $P[f]$ and $B[f]$ are built up according to step 1 and 2.

$$P = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & d_1 & 0 & 0 & d_4 & 0 & 0 & 0 \\ d_1 & 0 & 0 & 0 & 0 & d_4 & 0 & 0 \\ 0 & 0 & 0 & d_1 & 0 & 0 & d_4 & 0 \\ 0 & 0 & d_1 & 0 & 0 & 0 & 0 & d_4 \\ d_4 & 0 & 0 & 0 & 0 & d_1 & 0 & 0 \\ 0 & d_4 & 0 & 0 & d_1 & 0 & 0 & 0 \\ 0 & 0 & d_4 & 0 & 0 & 0 & 0 & d_1 \\ 0 & 0 & 0 & d_4 & 0 & 0 & d_1 & 0 \end{bmatrix}$$

In $B[f]$, $b_{01} = 1$. If now d_1 is set to 1 for the RMC form of f with polarity 0, those values p_{0s} of $P[f]$ have to be changed where the equation $(s_1 \cdot 0,\ s_2 \cdot 0,\ s_3 \cdot 1) = (0, 0, 1)$ holds. This relation is fulfilled for $s = 1, 3, 5, 7$. If the heuristic procedure (*b*) is applied ($p_{01} = 1$, $b_{00} = 0$), row 0 of $P[f]$ is changed to: $[1\ 0\ 1\ 1\ 1\ 1\ 1\ 1]$.

After setting b_{01} to 0, the only remaining don't-care term is $b_{04} = d_4$. Because of $p_{04} = 1$, d_4 is set to 1 and p_{04}, p_{05}, p_{06} and p_{07} are complemented. It follows that the coefficients of the RMC form of f with polarity 0 are $[1\ 0\ 1\ 1\ 0\ 0\ 0\ 0]$ with $d_1 = d_4 = 1$.

In the following table the values of the don't-care terms d_1 and d_4, following of the procedures (*b*) and (*c*),

Table 1: Values of don't-care terms and number of terms for the function in example 6

Polarity	Procedure (*b*)			Procedure (*c*)			Number of terms needed, if d_1 and d_4 are chosen optimally
	values of		number of terms	values of		number of terms	
	d_1	d_4		d_1	d_4		
0	1	1	3	0	0	5	3
1	0	0	4	0	0	4	2
2	0	1	3	0	0	3	3
3	1	1	3	0	0	4	3
4	0	0	3	0	0	3	3
5	1	1	2	0	0	4	2
6	0	0	2	0	0	2	2
7	1	0	3	0	0	3	3

are shown. Also the number of used product-terms for realisation of the corresponding RMC form is presented. The last column contains the number of product-terms if the don't-care terms are optimally chosen. It can be seen that (at least in this example) the heuristic procedures often yield the optimum result. But it cannot be followed out of this example that procedure (*b*) always shows better results than procedure (*c*). It is also a simple coincidence that in this example procedure (*c*) always yields $d_1 = d_4 = 0$. As a counterexample see the function:

$$[g] = [1\ \ d_1\ \ 0\ \ d_3\ \ 1\ \ 0\ \ 0\ \ 1]$$

5 Concluding remarks

A new method for constructing the polarity-matrix, which is used for minimisation of RMC forms of Boolean functions is presented. It can be used both for fully specified and incompletely specified functions. Owing to its high degree of parallelism the algorithm has got an area-time-tradeoff $AT^2 = n^2 \cdot 3^{n-1}$, if n is the number of variables.

6 References

1 AKERS, S.B.: 'On a theory of Boolean functions', *J. of the SIAM*, 1959, **7**, pp. 487–498

2 REDDY, S.M.: 'Easily testable realizations for logic functions', *IEEE Trans. on Computers*, 1972, **21**, pp. 1183–1188

3 GREEN, D.H.: 'Reed-Muller expansions of incompletely specified functions', *IEE Proc. E, Computers & Digital Techniques*, 1987, **134**, pp. 228–236

4 WU, X., CHEN, X., and HURST, S.L.: 'Mapping of Reed-Muller coefficients and the minimisation of XOR switching functions', *IEE Proc. E, Computers & Digital Techniques*, 1982, **129**, pp. 15–20

5 BIOUL, G., DAVIO, M., and DESCHAMPS, J.P.: 'Minimisation of ring-sum expansions of Boolean functions'. Philips Research Reports, 1973, **28**, pp. 17–36

6. BESSLICH, P.W.: 'Efficient computer method for XOR logic design', *IEE Proc. E, Computers & Digital Techniques*, 1983, **130**, pp. 203–206

7 FISHER, L.T.: 'Unateness properties of AND-XOR logic circuits', *IEEE Trans. on Computers*, 1974, **23**, pp. 166–172

8 ZHANG, Y.Z., and RAYNER, P.J.W.: 'Minimisation of Reed-Muller polynomials with fixed polarity', *IEE Proc. E, Computers & Digital Techniques*, 1984, **131**, pp. 177–186

9 EVEN, S., KOHAVI, I., and PAZ, A.: 'On minimal modulo-2 sums of products for switching functions', *IEEE Trans. on Computers*, 1967, **16**, pp. 671–674

10 BESSLICH, P.W., and BÄSSMANN, H.: 'Synthesis of Exclusive-Or logic functions using spectral techniques'. Second International Symposium on Spectral Techniques and Fault Detection, Ecole polytechnique de Montreal, Montreal, Canada 1986

11 FLEISHER, H., TAVEL, M., and YAEGER, J.: 'A computer algorithm for minimising Reed-Muller canonical forms', *IEEE Trans. on Computers*, 1987, **36**, 247–250

12 DAVIO, M., DESCHAMPS, J.P., and THAYSE, A.: 'Discrete and switching functions' (McGraw-Hill, 1978)

13 SAVAGE, J.E.: 'Area-time tradeoffs for matrix multiplication and related problems in VLSI models', *J. of Computer and System Sciences*, 1981, **22**, 230–242