Some Observations Related to Searching for Logic Functions Decomposition in Reed-Muller Spectral Domain

Dariusz Polok, Edward Hrynkiewicz
Institute of Electronics, Silesian University of Technology, Gliwice, Poland

Dariusz.Polok@polsl.pl, Edward.Hrynkiewicz@polsl.pl

Abstract - The paper deals with the problems of searching for decomposition of Boolean functions in the domain of Reed-Muller spectrum. The Ashenhurst decomposition is considered in the first turn but the decomposition based on Curtis decomposition model was mentioned, too. The special attention was paid for analyzing some problems associated with searching for decomposition with use of spectra with various polarities and to the method of finding beneficial input variables with possible permutations between them. It turned out that searching for decomposition in different polarizations of the Reed-Muller spectral domain was fruitless. But in some cases successful results were achieved owing to permutations between input variables before the Reed-Muller spectrum is calculated. Some promising observations related to that field are also included into this paper.

Keywords: Logic function decomposition, Reed-Muller spectrum, spectrum polarization, logic function implementation, FPGA

1. INTRODUCTION

Decomposition of Boolean functions is considered as one of key techniques applicable to the engineering process of electronic equipment. It may happen that designer intuitionally applies the decomposition methods tending to simplify the engineering process by splitting a large and hard manageable job into smaller and much simpler ones. But more often it is a conscious process and associated with the structural approach, when a general structure of a future device is developed first in the form of e.g. a block diagram and then individual parts of the project are assigned to different designers.

Since the time when FPGA modules appeared in the common market the decomposition approach has become even more important because large logic projects had to be subdivided into smaller sub-structures to fit into logic blocks of available hardware. Thus, in case of all engineering developments, including CPLD and PSoC structures, decomposition is enforced by discrete arrangement of such modules.

In the field of digital circuits the underpinning ideas for decomposition of Boolean functions were formulated by Ashenhurst (1959) in the following form:

$$f(x_{n-1}, \ldots , x_{r+1}, x_r, x_{r-1}, \ldots, x_0) = h[A,g(B)] \quad (1)$$

where:

$A = \{ x_{n-1}, \ldots ,x_{r+1}, x_r\}$ - is referred to as a free set.

$B = \{ x_{r-1}, \ldots ,x_1, x_0\}$ - is referred to as a bound set.
Let us assume that $\alpha\|\beta$ is the division between the arguments of the logic function, where (α_{min}=1; β_{min}=2). The symbol of α stands for the cardinality of the A set whilst β is the cardinality of the B set, i.e. $\alpha = \|A\|$ and $\beta = \|B\|$, The relationship (1) entails the fact that the logic block for the bound function has only one output. Curtis (1962) generalized the approach of Ashenhurst and set forth the baseline for so called disjoint Curtis decomposition that is defined by means of the formula (2):

$$f(x) = f((A), g_{k-1}(B), g_{k-2}(B), \ldots, g_0(B)) \quad (2)$$

where: k – number of g_i functions and $k_{max}=2^\alpha$

It means that the logic block for the bound function can be also a multi-output module, which offers simpler solutions in specific cases.

Since that times decomposition techniques underwent tempestuous development and great many decomposition methods have been presented and implemented into various software tools, even substantially different from each other For example Łuba and Salvaraj (1995) presented the decomposition process based on so called partition calculus and the achieved results are really beneficial with regard to management of FPGA resources. The Binary Decision Diagrams (BDD) are also used for the decomposition process which consists in splitting the original diagram into few smaller ones that can be easily implement in FPGA logic cell (Minato, 1996). The benefit from that approach consists in a cohesive representation of Boolean functions,

which is essential for decomposition. Another approach can be found in Kania, 2012 where author suggested using diagrams of outputs to seek for a really efficient decomposition. The approach based on functional decomposition was also presented in Sasao and Kurimoto (2000 One more research method is based on the idea disclosed by Varna and Trachtenberg (1993) who elaborated a solution that enables decomposition of logic functions in the Reed-Muller spectral domain. According to their idea Boolean functions can be implemented within a set of logic gates. According to their idea logic functions can be implemented within a set of logic gates. In the works of Hrynkiewicz and Kołodziński (2007) the authors demonstrated that that method can be even enhanced to enable decomposition of logic function implemented in CLBs of FPGA.

2. DISJOINT ASHENHURST DECOMPOSITION IN THE REED-MÜLLER SPECTRAL DOMAIN

The straight Reed-Müller transform is expressed by the formula (3) (Green 1986; Varma, Trachtenberg 1993; Thorton at al. 2001).

$$f_S = R(n) \cdot f \qquad \text{with respect to GF(2)} \qquad (3)$$

where: $R(n)$ is the Reed-Müller matrix; f – vector of the Boolean function values; f_S – Reed-Müller spectrum (the vector of spectral coefficients); n number of variables.

$$\text{for } n=2,3,4, \; .... \qquad (4)$$
$$R(n) = R(1) \otimes R(n\text{-}1) \qquad (5)$$

where: $\otimes$ - Kronecker product

Because $\qquad R(1) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \qquad$ (Green, 1986) $\qquad$ (6)

Then, for example

$$R(2) = \begin{bmatrix} R(1) & 0 \\ R(1) & R(1) \end{bmatrix} \qquad (7)$$

The reverse Reed-Müller transform is expressed by the formula (8)

$$f(x_{n\text{-}1}, \, ..., \, x_0) = X_R f_S \text{ with respect to GF(2)} \qquad (8)$$

where: $\qquad X_R = \overset{n\text{-}1}{\underset{i=0}{\otimes}} [1 \; x_{n-i-1}] \qquad (9)$

and $[1 \; x_{n\text{-}i\text{-}1}]$ – the basic vector of the reverse Reed-Müller transform.

To carry out decomposition in the spectral domain the vector of spectral coefficient must be transposed to the form of a single row and then split into 2^α groups of coefficients.

Next, coefficients with the lowest indexes are removed from each group. The disjoint Ashenhurst decomposition is possible when the non-zero partial spectra obtained in the foregoing way are mutually equal or one partial spectrum is equal to zero (Warma and Trachtenberg, 1993; Hrynkiewicz and Kołodziński, 2007). The logic function implemented in the bound block is calculated as a reverse Reed-Muller transform from a non-zero partial spectrum supplemented with the digit of "0" at the left-hand side, while the logic function represented by the free block is calculated from spectrum of a decomposed function, where the non-zero partial spectrum is substituted with the digit of "1" and the null partial spectrum is substituted with the digit of "0".

Example 1.
Let us consider the function f(x)=f (x_4, x_3, x_2, x_1, x_0)
$=\Sigma m$(2,3,6,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,26, 27,30), where m means the minterms.

First, the input variables are split into two sets:
A={x_4, x_3} and B={x_2, x_1, x_0} After appropriate calculations the transposed Reed-Muller spectrum for the foregoing function adopts the following form:

$$f_S = [0 \; \underset{S_0}{\underline{0100001}} \quad 0 \; \underset{S_1}{\underline{0100001}} \quad 0 \; \underset{S_2}{\underline{0100001}} \quad 0 \; \underset{S_3}{\underline{0000000}}]^T$$

It means that the requirements for Ashenhurst decomposition are met then function $f(x)$ can be decomposed with the assumed division. Therefore

$$g(x_2, x_1, x_0) = x_1 \oplus x_2 x_1 x_0$$
$$h(x_4, x_3, g) \;\; = \; g \oplus x_3 \oplus x_3 g \oplus x_4 \oplus x_4 g$$

The implementation of the exemplary Boolean function is shown in Fig. 1.
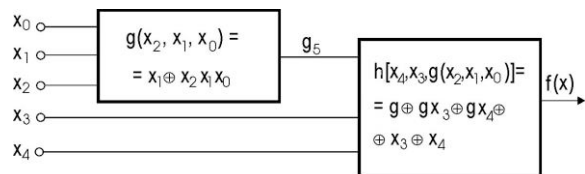

Fig. 1. Block diagram of the logic function from Example 1

3. SAME OBSERVATIONS MADE WHILE EXPERIMENTS IN THE REED-MULLER SPECTRAL DOMAIN WERE CARRIED OUT

Since the existing decomposition methods are incapable of finding out an optimum solution for all possible cases of Boolean functions the research studies in that area are still