

Article

Application of Generalized Reed–Muller Expression for Development of Non-Binary Circuits

Elena Zaitseva ^{1,*}, Vitaly Levashenko ¹, Igor Lukyanchuk ², Jan Rabcan ¹, Miroslav Kvassay ¹ and Patrik Rusnak ¹

- ¹ Faculty of Management Science and Informatics, University of Žilina, 010 26 Žilina, Slovakia; vitaly.levashenko@fri.uniza.sk (V.L.); jan.rabcan@fri.uniza.sk (J.R.); miroslav.kvassay@fri.uniza.sk (M.K.); patrik.rusnak@fri.uniza.sk (P.R.)
- ² Laboratory of Condensed Matter Physics, University of Picardie, 33 rue St. Leu, 80000 Amiens, France; lukyanc@ferroix.net or igor.loukiantchouk@u-picardie.fr
- * Correspondence: elena.zaitseva@fri.uniza.sk

Received: 20 November 2019; Accepted: 16 December 2019; Published: 21 December 2019



Abstract: Semiconductor devices and binary information technology reach their limits set by the atomic size of miniaturization, calculation speed, and the fundamental principle of energy dissipation per bit processing. Therefore, new technologies in logic design and mathematical approaches must be investigated. Application of multiple-valued logic (MVL) in logic design allows developing gates and circuits with more than two stable states. This enables packing an unprecedented high-density of information. Based on this idea, a new technique of the programmable logic arrays (PLA) construction based on MVL units is considered. The unique aspect of this technique is the application of recurrent generalized Reed–Muller expression (GRME) for MVL function representation. The recurrent procedure for this expression's construction is considered and applied in the PLA development. The proposed structure of PLA consists of two blocks that are memory and logic block. In this paper, we also consider the possibility to use the ferroelectrics for the implementation of cells of the memory block of PLA. The development of gates with multi-stable states is possible by the ferroelectrics ability to pin the polarization as a sequence of stable states.

Keywords: logic circuit; programmable logic arrays; multiple-valued logic; generalized Reed–Muller expression

1. Introduction

One of the main challenges in the field of data processing is providing technology for efficient information processing that can work with a huge amount of data. This especially holds at the level of basic computations. Nowadays, semiconductor-based computation is the most used technology for such a task. A basic principle of this technology is that the data bit can have one of only two values: zero and one. These values correspond to two stable states of transistors: off and on. This principle has its limitations as researchers point out back in the 1970s–1980s. One such problem is interconnection on and between chips. Up to 70% of the active logic elements are taken by interconnections [1], which is caused mostly by the complex routing, digital logic components placement and by mechanical, thermal and electrical restrictions that occur due to the increased number of connections [2,3]. Another problem in semiconductor technology development lies in clock speed increasing [4]. The clock speed represents how quick the transistor states can be changed and it directly influences the computer performance. Until recently, the clock speed has doubled almost every year. Nowadays, the clock speed cannot be raised that rapidly which resulted in dual-core or quad-core processors. These new approaches have their benefits, but they do not improve the performance in such a significant way as



is required. This is caused by the limitation of the binary data amount that can be transferred. Based on these problems, there are tendencies to consider other approaches that do not focus only on binary data. One such approach is known as multiple-valued logic (MVL), which works with more than two values and thus allows encoding more information in "multi-valued bit" than in case of binary-valued bit in two-valued logic [1,5].

In the development of the new innovative approach in computer design, the two main pieces of research should be considered. First of them is technological aiming at the development of new multi-valued logic gates. Another one is mathematical, which deals with the development of new methods based on MVL in logic design. The acceptance and usage of semiconductor technology in computer design was due to the help of the appropriate mathematical tools that were introduced in binary logic, which started with the early work of G. Boole [6] and C.E. Shannon [7]. Its results have been used for the development of methods of logic synthesis in nanotechnology [8,9]. In the case of MVL design, the main priority has been devoted to elaborate logic gates based on 3-valued logic because logic with this radix, i.e., with radix 3, results in almost minimal complexity of multi-valued based hardware [1]. Because of that, this type of logic is also considered in this paper.

There were two first independent investigations in multiple-valued logic about 1920 conducted by E. Post [10] and J. Lukasiewicz [11]. However, the fist algebra that was functionally complete for any radix was the Post's algebra. This algebra is based on a totally ordered set M of elements $0 < 1 < \ldots < m - 1$ and it has defined operations maximum (MAX) that is OR in binary algebra, minimum (MIN) that is AND in binary algebra, and literal that coincides with NOT in binary algebra. By using Post's algebra, it is possible to define any function over M that will use a composition of MIN, MAX, and literals. This fact is essential because it allows constructing any logic circuit from the gates implementing the primitive functions from this algebra. Another algebra that is based on modulo-sum and modulo-product operators was considered by B. A. Bernstein in 1924 [12]. As can be found in [9,13–15], several authors considered its use in logic design. As for the binary case, this algebra is known as Reed-Muller algebra. Another MVL algebra is known as Webb's algebra. This algebra has one 2-variable operator. This operator can be transformed for the binary case into Sheffer-Stroke operator [16]. Many authors combine those operators introduced in these algebras with various variants in order to achieve specific properties that make them accepted in the multi-valued logic-based design of computation devices [9,15,17,18]. The algebra constructed over modulo-sum and modulo-product operations can be transformed into the regular recurrent canonic structure (expression), which is a very important advantage in the technological process [9,14,15].

Considering all the previously mentioned advantages, the MVL based design can replace binary technologies, but this is contingent on the availability of circuit realizations that must be better as in case of circuits of binary technologies. In the years 1950–1960 some of the first multi-valued based devices were designed. For example, multiple-phase devices such as the parametron [19], the multiple-frequency oscillator concept of Edson [20], and multi-aperture square-loop ferrite devices [21]. The latter multi-aperture square-loop ferrite devices were applied to the construction of a ternary computer, the SETUN computer [22]. Unfortunately, they were overrun by the semiconductor technology (bipolar junction transistor and unipolar MOS and CMOS transistors).

There are some tendencies to build an MVL based circuit using CMOS (some devices can be found in [23,24]) or to combine binary and multi-valued blocks in logic circuits for CMOS based design as has been shown in [25,26]. The use of another semiconductor technology known as resonant tunneling transistors and diodes has also been considered for multi-valued logic circuits [27,28], but there are some technological disadvantages. For example, the tunnel diode is low power due to low voltages (tenths of a volt) and it has small junction areas. As an alternative to MOS transistors in MVL based design, carbon-nanotube field effect transistor (CNFET)s have been studied [29]. One of the first CNFET-based approaches for ternary logic circuit design has been presented in [30]. The main downfall of this approach was in the use of resistive loads and hence large off-chip resistance needs. Some aspects of CNFET technology application for multi-valued have been considered in [3,31,32].

Investigations on quantum computation and biochips based MVL circuits have also been conducted. Quantum computations try to use quantum phenomena, such as quantum superposition and quantum entanglement, to achieve computational capacities that increase exponentially with the increase in the amount of data. The basic building block of these computers is a quantum bit (qubit). It can be represented by nanoscale physical properties, such as nuclear spin [33,34]. On the other hand, biochips represent biochemical logic circuits, which are implemented using biochemical reactions. Such circuits are useful for the creation of novel living organisms with well-defined purposes and behaviors. A key issue in the design and implementation of this kind of circuits is matching logic gates in such a way that the couplings can produce the correct behavior [18,35].

Ferroelectricity is another technology which can has nature application in MVL based logic design. The main idea here is based on the capacity of certain ferroelectrics to host multiple polarization states having the same energy. These polarization states are used as stable states in design of gates for MVL function realization. By forming the logical levels of the unit, these states can be easily controlled by the electric field, temperature and/or applied strain. The most evident way for this is to use the degeneracy of polarization orientation in the cubic crystal structure of perovskite oxides, Lead of Barium Titanate, PbTiO3, along with the high-symmetry polarization axes. Our first studies showed that the strained films of these materials can host a variety of logically different multilevel hysteresis loops [36], holding two, three, or even four energetically stable polarization positions. Even more fascinating opportunity can be provided by a variety of polarization topological structures, confined in the nanoscale ferroelectric samples: nanodots, and nanorods (nanopillars). The switching between different types of these structures [37–39]: vortices, skyrmions and domain patterns allow realizing even more lively logical states for MVL based logic design.

The new structure of Programmable Logic Arrays (PLA) for MVL function relegalization is considered in this paper. This new PLA's structure comes from the MVL function representation in form of generalized Reed-Muller expression [14,15]. This form is introduced in algebra based on modulo-sum and modulo-product operators and it is considered in Section 2. In contrast with the Post's algebra, the form for MVL function representation in this algebra is canonical recurrent and regular. Thanks to these properties, it is possible to create a good implementation of PLA for MVL functions. The structure of PLA based on the generalized Reed-Muller expression for MVL function is introduced in Section 3 and, in Section 4, aspects of MVL memory elements based on ferroelectric materials are considered.

2. Reed-Muller Algebra Extension for MVL Function

Functional expressions for discrete functions can be viewed as formulae specifying behavior of functions. In other words, they describe uniquely the mapping between the domain (the set where the variables take values) and the range (the set where the function takes its values) defining the function considered. A functional expression consists of symbols for variables, symbols for functions, and symbols for operations over variables and functions. To determine the meaning of symbols for operations, some algebraic structures, not necessarily identical, are imposed on the domain and the range.

One of the often-used systems for manipulation with MVL functions is algebra based on modulo-sum and modulo-product or extension of Reed-Muller algebra (modular algebra) [15,40]. It is based on a totally ordered set *M* of elements 0 < 1 < ... < m - 1 and use the operations modulo-sum (SUM) and modulo-product (PROD). This set of operations is functionally complete for MVL, and any function over *m* can be defined as a composition of SUM and PROD operations. In Table 1 these operations are defined for the radix *m* = 3.

An MVL function $f(x) = f(x_1, x_2, ..., x_n)$ of *n* variables is a logic function defined for the set {0, 1, ..., m - 1}, which meets the sets' mapping {0, 1, ..., m - 1}^{*n*} \rightarrow {0, 1, ..., m - 1}. The practical applications of MVL, in logic design, need the representation of MVL function by canonical orthogonal form. One of such forms for MVL function representation is generalized Reed-Muller expression (GRME) that is defined similarly as Reed-Muller expansion of binary function.

$\begin{array}{c} x_1 \\ x_2 \end{array}$	0	0	0	1	1	1	2	2	2
	0	1	2	0	1	2	0	1	2
$x_1 + x_2 \pmod{3}$	0	1	2	1	2	0	2	0	1
$x_1 x_2 \pmod{3}$	0	0	0	0	1	2	0	2	1

Table 1. The truth-table of operations SUM and PROD for m = 3.

A GRME of MVL function $f(x_1, x_2, ..., x_n)$ of *n* variables (with radix $m \ge 2$) according to [14,15] is defined by equation:

$$A(x) = \sum_{k=0}^{m^n-1} a^{(i)} x_1^{k_1} x_2^{k_2} \dots x_n^{k_n} \pmod{m}, \tag{1}$$

where $a^{(k)}$ are coefficients of GRME and $a^{(k)} \in \{0, 1, ..., m - 1\}$; k_i is the *i*-th digit of *m*-valued representation of parameter $k, k = (k_1, k_2, ..., k_n)_m$ where i = 1, 2, ..., n. The coefficients of GRME (1) can be represented as a vector that is named as coefficient vector $a = \left[a^{(0)} a^{(1)} \dots a^{(m^n-1)}\right]^T$.

For example, let us to consider the GRME of 3-valued (m = 3) function of 2 variables (n = 2):

$$A(x) = 1 + x_2 + 2x_1x_2 \pmod{3}.$$
(2)

The GRME of this function has coefficient vector $\mathbf{a} = [1\ 1\ 0\ 0\ 2\ 0\ 0\ 0]^{T}$ and can be represented, in generally, as follows:

$$A(x) = 1 + 1 \cdot x_2 + 0 \cdot x_2^2 + 0 \cdot x_1 + 2 \cdot x_1 x_2 + 0 \cdot x_1 x_2^2 + 0 \cdot x_1^2 + 0 \cdot x_1^2 x_2 + 0 \cdot x_1^2 x_2^2 \pmod{3}.$$
 (3)

Any value of the MVL function for specified values of its variables can be calculated based on the GRME. The calculation of the function values defined by GRME (2) for all possible values of the function variables is illustrated in Table 2. The last column of this table is interpreted as truth vector of function (2) if the values of the variables are lexicographically ordered [15]. So, the truth vector for the considered function is $x = [1 \ 2 \ 0 \ 1 \ 1 \ 1 \ 0 \ 2]^T$ according to Table 2.

<i>x</i> ₁	<i>x</i> ₂	$A(x)=1+x_2+2x_1x_2 \pmod{3}$	The Truth Table (Truth Vector) of the Function, <i>x</i>
0	0	$1 + 0 + 2 \cdot 0 \cdot 0 \pmod{3}$	1
0	1	$1 + 1 + 2 \cdot 0 \cdot 1 \pmod{3}$	2
0	2	$1 + 2 + 2 \cdot 0 \cdot 2 \pmod{3}$	0
1	0	$1 + 0 + 2 \cdot 1 \cdot 0 \pmod{3}$	1
1	1	$1 + 1 + 2 \cdot 1 \cdot 1 \pmod{3}$	1
1	2	$1 + 2 + 2 \cdot 1 \cdot 2 \pmod{3}$	1
2	0	$1 + 0 + 2 \cdot 2 \cdot 0 \pmod{3}$	1
2	1	$1 + 1 + 2 \cdot 2 \cdot 1 \pmod{3}$	0
2	2	$1 + 2 + 2 \cdot 2 \cdot 2 \pmod{3}$	2

Table 2. The truth table of the multiple-valued logicfunction (2).

The MVL function (2) in Post's algebra with the operations maximum (MAX), minimum (MIN) and literal according to [15,40] has more complex representation, i.e.,:

$$f(\mathbf{x}) = (1 \land \varphi_0(x_1) \land \varphi_0(x_2)) \lor (2 \land \varphi_0(x_1) \land \varphi_1(x_2)) \lor (1 \land \varphi_1(x_1) \land \varphi_0(x_2)) \lor (1 \land \varphi_1(x_1) \land \varphi_1(x_2)) \lor (1 \land \varphi_1(x_1) \land \varphi_2(x_2)) \lor (1 \land \varphi_2(x_1) \land \varphi_0(x_2)) \lor (2 \land \varphi_2(x_1) \land \varphi_2(x_2)),$$
(4)

where \wedge is symbol of the operation MIN; \vee is symbol of the operation MAX and $\varphi_a(x_i)$ is literal operation defined as follows:

$$\varphi_a(x_i) = \begin{cases} m-1, \text{ if } x_i = a\\ 0, \text{ if } x_i \neq a \end{cases}$$
(5)

for i = 1, 2, ..., n and a = 0, 1, ..., m - 1.

Note that a GRME of MVL function of one variable (n = 1) according to (1) is represented as:

$$A(x) = \sum_{k=0}^{m-1} a^{(k)} x^k = a^{(0)} + a^{(1)} x + a^{(2)} x^2 + \dots + a^{(m-1)} x^{m-1} \pmod{m}, \tag{6}$$

or in recurrent form:

$$A(x) = a^{(0)} + \ldots + x \cdot \left(a^{(1)} + x \cdot \left(a^{(2)} + \ldots + x \cdot a^{(m-1)}\right) \ldots\right) (mod \ m).$$
(7)

The calculation procedure of GRME's value (6) using formula (7) can be illustrated by regular and recurrent flow diagram in Figure 1.



Figure 1. Calculation of value of an MVL function of one variable based on generalized Reed-Muller expression (6).

The vector interpretation of function values and coefficients of GMRE as truth vector and coefficient vector permits representing GRME (1) in matrix form [14,15]:

$$x = \mathbf{T}_n a \qquad (mod \ m),\tag{8}$$

were *a* is the coefficient vector of GRME (1) and *x* the truth vector of the considered MVL function, matrix \mathbf{T}_n is a transformation matrix whose elements are defined considering the GRME (1) as follows:

$$T(x,i) = x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \pmod{m},$$
(9)

for $x, i \in \{0, 1, ..., m^n - 1\}$, where x_j and i_j is the *j*-th digit of *m*-valued representation of parameter *x* for j = 1, 2, ..., n.

Transformation (8) was introduced in [14,15]. Authors of [15] has studied properties of this transformation and transformation matrix T_n . Important property of the matrix T_n is computation by the recurrent procedure:

$$\mathbf{T}_n = \mathbf{T}_{n-1} \otimes \mathbf{T}(mod \ m), \tag{10}$$

where symbol \otimes denotes Kronecker product [41]; matrix **T** agrees with the GRME of one variable (6), and its elements are:

$$t(x,i) = x^{i} (mod \ m), \tag{11}$$

where $x, i \in \{0, 1, ..., m - 1\}$ (for x = 0 and i = 0 the value $0^0 \equiv 1$).

The other definition of the matrix **T** is possible by the vector $\mathbf{T}_0 = [1 \ x \ \dots \ x^{m-1}] \pmod{m}$ for $x \in \{0, 1, \dots, m-1\}$. For example, for m = 3, the matrix **T** based on vector $\mathbf{T}_0 = [1 \ x \ x^2] \pmod{3}$ is:

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 1 \end{bmatrix}.$$
(12)

GRME (1) of an MVL function of *n* variables considering the matrix procedures (8)–(10) can be defined as:

$$\mathbf{x} = \left(\begin{bmatrix} 1 \ x_1 \ \dots \ x_1^{m-1} \end{bmatrix} \otimes \dots \otimes \begin{bmatrix} 1 \ x_n \ \dots \ x_n^{m-1} \end{bmatrix} \right) a \pmod{m}, \tag{13}$$

and GRME (1) considering recurrent procedure (9) is modified into the following form:

$$A(x) = A(x_1, A(x_2, \dots, A(x_n))) (mod \ m).$$
(14)

For example, for m = 3 and n = 2, the GRME is represented by next form:

$$\begin{aligned} A(\mathbf{x}) &= (a^{(0)} + a^{(1)}x_2 + a^{(2)}x_2^2) + x_1 \cdot (a^{(3)} + a^{(4)}x_2 + a^{(5)}x_2^2) + x_1^2 \cdot (a^{(6)} + a^{(7)}x_2 + a^{(8)}x_2^2) \\ &= (a^{(0)} + a^{(1)}x_2 + a^{(2)}x_2^2) + x_1 \cdot ((a^{(3)} + a^{(4)}x_2 + a^{(5)}x_2^2) + x_1 \cdot (a^{(6)} + a^{(7)}x_2 + a^{(8)}x_2^2)) \\ &= (a^{(0)} + x_2 \cdot (a^{(1)} + a^{(2)}x_2)) \\ &+ x_1 \cdot ((a^{(3)} + x_2 \cdot (a^{(4)} + a^{(5)}x_2)) + x_1 \cdot (a^{(6)} + x_2 \cdot (a^{(7)} + a^{(8)}x_2))) (mod 3). \end{aligned}$$
(15)

The Equation (14) is used below in the design of programmable logic arrays for the calculation of MVL function values. It is worth noting that there are many algorithms for calculation of the coefficients $a^{(i)}$, for $i = 0, 1, ..., m^n - 1$, of GRME. Some of them are considered in detail in [14,15,41].

3. Logic Design of Multi-Valued Circuit

MVL is the background of the logic design of multi-valued circuits [1]. Different techniques are used to develop these circuits [8,42,43]. One of these techniques for the design of combinational logic circuits is PLA [42]. A PLA is a kind of programmable logic device: PLA has an invariable structure and new function implementation is possible by reprograming this structure. The main advantages of PLA over other logic circuits are [9]: easy design, verification and checking, far simpler layout whose development is less time-consuming, and higher switching speed. PLA structure is closely correlated with the mathematical background for the representation of logic (Boolean and MVL) functions [9,44]. The technique of PLA design for MVL has been developed as a generalization of well-known PLA based semiconductor techniques for which Boolean algebra is the mathematical background [42–44]. In Boolean algebra, disjunctive normal form is typically used for the development of PLA. Its generalization for the realization of MVL functions results into application of Post algebra for PLA design [42–44]. There are also studies of PLA design based on Reed–Muller Expression in Boolean algebra [45,46]. Such a PLA has better testability as has been shown in [44,45]. The generalization of this technique for MVL has been implemented in [9,47]. Authors of [9,47] have used GRME (1) for the design of PLA for the realization of MVL functions.

In this paper, we consider the PLA development with the use of canonical GRME (10) of MVL function. This PLA has two parts (Figure 2): memory (M) and logic block (L). The memory block has m^n inputs to program GRME coefficients values (*m* is the value of function radix and *n* is the number of the function variables). It can be interpreted as memory for the coefficient vector of GRME. Any of MVL functions can be implemented by reprogramming of GRME coefficients. These coefficients are read and transmitted to the logic block to compute the MVL function values depending on the values of variables that are incorporated into the logic block through a set of *n* external inputs of the block. In addition, the logic block has m^n inputs from the memory block.



Figure 2. Structure of the considered PLA used to realize MVL function.

The logic block consists of sum-product's homogeneous sub-blocks that are linked together to give the output. The application of the GRME in form (14) implies the structure of the logic block, which is homogeneous, recurrent, and parallel. This block consists of *n* levels (*n* is the number of function's variables). Such a structure of the logic block is due to a specific calculation of MVL function value based on GRME and can be illustrated by the matrix procedure (8) where transformation matrix T_n is computed recurrently according to (9). The *s*-th level (s = 1, 2, ..., n) of the logic block has input of the (n - s + 1)-th variable and consists of (m^{n-s}) homogeneous sub-blocks. Every one of the sub-blocks implements the calculation of GRME of one variable according to (6) and can be represented by the flow diagram in Figure 1. According to the flow diagram, the sub-block consists of (m - 1) multiplying and (m - 1) summing gates. The outputs of sub-blocks at the *s*-th level are inputs of sub-blocks at the

coefficients. This layout allows synthetizing many MVL functions expressed by GRME. Let us focus on the PLA realizing 3-valued logic functions (m = 3) of 3 variables (n = 3) to present important aspects of the proposed structure. An implemented MVL function is uniquely represented by the memorized coefficients values that can be reprogramed in the memory. The modification of these coefficients values allows changing implemented 3-valued function. The structure of such a PLA is depicted in Figure 3.

(s + 1)-th level. The first level of the logic block has m^n inputs from memory block of PLA to read the



Figure 3. Structure of the PLA for realization of 3-valued logic function of 3 variables (m = 3, n = 3).

As one can see, it is composed of two blocks that are memory and logic block. The PLA has 27 inputs that are linked with the memory block and three inputs for function variables x_1 , x_2 , and x_3 that are the inputs of the logic block. These inputs of the memory block allow programing of the coefficients of the GRME. The memory block is formed by 27 memory cells. The input of each of these cells is an external input of the PLA. The outputs of the memory block are inputs of the sub-blocks at the first level of the logic block. The logic block formed by three levels. The first level consists of nine sub-blocks that are linked to three sub-blocks at the second level. The third level is formed by one sub-block. All these sub-blocks are homogeneous and each of them has three inputs from the previous

level, one external input for variable and one output. The output of the sub-block at the last level is the output of the PLA. Every sub-block of the logic block has two multiplying and two summing nodes.

We simulated this structure of PLA (Figure 3) using Matlab Simulink 2019a [48]. Its model is shown in Figure 4. The sub-blocks of this structure were created according to the previous description, and model of one sub-block can be viewed in Figure 5. The memory block of the PLA consists of m^n cells for storing the coefficients of the GRME. The modification of these cells (reprogramming) allows implementing other MVL functions without the modification of the PLA structure. For example, implementations of two different MVL functions by the considered PLA are shown in Figure 6. The MVL function in Figure 6a is generated based on the GRME with the coefficient vector:



Figure 4. The simulated PLA of 3-valued function of 3 variables (m = 3, n = 3) in Simulink.

and the MVL function in Figure 6b is formed by the GRME whose coefficient vector looks as follows:

```
a = [101101101101101101101101101101101]^{T}.
```



Figure 5. Model of the sub-block of PLA for 3-valued function created in Simulink.



Figure 6. Simulation of one PLA used for realization of two different MVL functions in Matlab. (**a**) a function with coefficient vector $\mathbf{a} = [101112212101211012121012101]^{T}$, (**b**) a function with coefficient vector $\mathbf{a} = [101101101101101101101101]^{T}$.

These figures show how values of the output of the PLA changes depending on values of the inputs and coefficient vectors.

The total number of logic gates influences the energy consumption of a logic circuit. Therefore, the decrease in the number of summing and multiplying nodes in PLA allows reducing its energy consumption. According to equation (3) and flow diagram depicted in Figure 1, the number of product operations is m - 1 for the GRME of one variable. So, the number of multiplying nodes in every sub-block of the PLA is m - 1 too. The proposed PLA has n levels (n is the number of MVL function variables), and the *i*-th level contains m^{i-1} (i = 1, 2, ..., n) sub-blocks. It follows that the structure of this PLA includes $m^n - 1$ multiplying nodes and m^n -1 summing nodes. On the other hand, authors of works [15,40] have shown that the numbers of product and sum operators in GRME of form (1) are $n \cdot m^n$ and $m^n - 1$, respectively. The comparison of the total number of operators used for the representation of MVL function by GRME of form (1) and form (10) are shown in Table 3. The analysis of these results allows us to conclude that the transformation of GRME into form (10) and the use of this form for the technical realization of PLA can result in a programmable logic circuit with less energy consumption.

n	m = 2		<i>m</i> = 3		m = 4		m = 5	
	Equation (1)	Equation (14)	Equation (1)	Equation (14)	Equation (1)	Equation (14)	Equation (1)	Equation (14)
2	8	3	18	8	32	15	50	24
3	24	7	81	26	192	63	375	124
4	64	15	324	80	1024	255	2500	624
5	160	31	1215	242	5120	1023	15,625	3124
6	384	63	4374	728	24,576	4095	93,750	15,624
7	896	127	15,309	2186	114,688	16,383	546,875	78,124
8	2048	255	52,488	6560	524,288	65,535	3,125,000	390,624
9	4608	511	177,147	19,682	2,359,296	262,143	17,578,125	1,953,124
10	10,240	1023	590,490	59,048	10,485,760	1,048,575	97,656,250	9,765,624
11	22,528	2047	1,948,617	177,146	46,137,344	4,194,303	537,109,375	48,828,124
12	49,152	4095	6,377,292	531,440	201,326,592	16,777,215	2,929,687,500	244,140,624
13	106,496	8191	20,726,199	1,594,322	872,415,232	67,108,863	15,869,140,625	1,220,703,124
14	229,376	16,383	66,961,566	4,782,968	3,758,096,384	268,435,455	85,449,218,750	6,103,515,624
15	491,520	32,767	215,233,605	14,348,906	16,106,127,360	1,073,741,823	$457,764 \times 10^{11}$	30,517,578,124

Table 3. The numbers of product and sum operators for generalized Reed-Muller expression (1) and (14).

4. Ferroelectric Elements

The PLA depicted in Figure 2 has two main blocks that are memory block (M) and logic block (L). Logic block (L) can be realized based on two types of gates that are product and sum gates. These elements can be implemented based on different technologies, such as CMOS technology considered in [49]. Memory (M) of the PLA must contain m^n memory cells storing the coefficients of GRME. Each cell must be able to store one of *m* possible values that a coefficient of GRME can take. There is different implementation of multi-valued memory [3,31,35,36]. For the implementation of the memory cell, we need to use the system in which the non-binary logic levels are realized as a set of the energy stable or metastable states that can be switched by the applied external pulses, resulting from the electrical and magnetic fields, mechanical strains, etc. As a promising example, we consider the recently suggested multilevel polarization switching in substrate-deposited thin films of ferroelectric perovskite oxides [36]. The model ferroelectric material, PbTiO3, can be recommended for realization of gates with some (more than two) stable states. It has the pseudo-cubic structure is viewed and the technology of thin-film deposition of PbTiO3 is fairly well controlled and it can operate at room temperature. Importantly, the polarization states of PbTiO3 films crucially depend on the strain, imposed by the substrate that can have both the tensile and compressive character. Accordingly, the polarization can have either in- or out-of-plane orientations with respect to the plane of the film. The strain-temperature um-T phase diagram of the substrate-deposited single-domain PbTiO3 films depicted in Figure 7a demonstrates the hosting of three ferroelectric phases that are the phase c, the phase aa, and the phase r [50]. Bubble notations in Figure 7a indicate the location of the hysteresis loops that are depicted in Figure 7b.



Figure 7. Multilevel hysteresis switching in films of PbTiO3 (**a**) Strain-temperature phase diagram of strained PbTiO3 film; (**b**) Example of hysteresis loops.

The key point to define the stable states here is that staying in one of the thermodynamically stable states, the system can have other metastable states that are the legacy of the phases, stable in

other parts of the um-T phase diagram. The external electric field permits to switch between these states and this property allows realizing the complicate hysteresis loop with various branches and the number of the branches is interpreted as a number of stable states for multi-valued gates design and development. Therefore, the structure of the hysteresis loop depends on the energy profile and on the protocol of the field and their modification allows forming loops of complex structure. The hysteretic transitions between c- aa- and r-state can be considered as 2-branch, 3-branch, and 4-branch loops. These interpretations of hysteresis loops are presented in Figure 7b.

Four panels are shown in Figure 7b for the implementation of logics with different radix. The 2-valued logic (Boolean algebra) bases on operators with two states. The realization of these operators can be based on the definition of two stable states of hysteresis loop. Such a 2-branch hysteresis loop is shown in panel A of Figure 7b. These loops can be implemented in the phase c. The phase aa in the vicinity of the first-order transition from the phase c hosts a stable aa-state and two metastable c-states. The hysteresis loop for 3-valued logic should have three stable states. The 3-branch hysteresis loop in panel B of the Figure 7b allows interpreting three stable states and can be used for the development of gates of 3-valued logic. Based this loop 3-valued memory gate can be developed according to [36]. Such memory gates can be used for memory (M) in PLA for the realization of 3-valued function shown in Figure 3. The specific aspect of this loop is stack-wise access to all the logical levels. It is its advantage in comparison with 4-branch hysteresis loops which have restrictions for stack access between logical levels. Finally, the r-phase has two polarization components and two metastable states. The 4-branch loops can be used for the implementation of operators of 4-valued logic. These loops have two realizations which are presented in panels C and D of Figure 7.

Although several other implementations of the multilevel physical systems were proposed recently (see [36] for references), the suggested ferroelectric realization looks as highly promising, because of its outstanding simplicity and the symmetry-protected stability. Moreover, the strain-temperature reprogrammable switching architecture suggested in [36] gives even more insight into the architecture and functionality of the MVL circuits.

The ferroelectric multilevel technology operates with the highly nonvolatile logic states. Hence it requires less power than the typically used dynamic random-access memory elements where most of the power is used for the cyclic refresh of the stored information. The switching process in the ferroelectric cell is based on the displacement of the polar ions in response to an applied field, which is extremely fast and is of the order of the nanoseconds and even less. Therefore, the operational switching time can achieve the record values and is mostly limited by the rapidity of the external interconnect. At the same time, the mechanism of the switching itself is more complicated than the single-domain polarization turn and involves the domain nucleation and dynamics. The full-scale modeling of the switching process for the multilevel ferroelectric elements is now in progress.

5. Conclusions

The new structure of PLA for MVL function realization was proposed in this paper. The MVL function is represented in the form of GRME in this case. The PLA consists of two parts that are memory and logic blocks. We propose the consideration of new technology for the memory realization based on ferroelectric material. The change of coefficients in memory of PLA allows modifying the realized MVL function. The important specific of the logic block of the PLA presented in this paper is regular hierarchical and homogenous structure. This structure results from the application of GRME for description and representation of MVL function. The structure of the logical block consists of *n* levels and the *s*-th level (s = 1, 2, ..., n) includes (m^{n-s}) homogeneous sub-blocks. The sub-block is formed by multiplying and summing nodes. The technical implementation of these nodes as gates has similar parameters in comparison with the gates of operations of minimal, maximum and literal according to a theoretical evaluation in [51–53]. The advantages of GRME based PLA in comparison

with Post algebra-based PLA is the best testability. The testability of GRME based circuits has been considered, for example, in [54].

In further research, the many-outputs structure of PLA based on GRME representation of MVL function will be investigated. The GRME with fixed and mixed polarity will be considered for the MVL function representation. The type of polarity indicates the inversion operator of variables in GRME [15]. The specified variable has the same inversion for all product terms in case of fixed polarity. The inversion of the specified variable can be different for different product terms in the case of GRME with mixed polarity [15]. Studies in [15,47,55] have shown that there is a possibility to decrease the complexity of MVL-function representation depending on different types of GRME's polarities. The testability of the presented structure of the PLA should be considered too. For these purposes, the algorithms proposed in [54,56] can be considered.

Author Contributions: Conceptualization, E.Z. and I.L.; methodology, E.Z., V.L. and I.L.; software, J.R.; validation, V.L., J.R. and M.K.; formal analysis, E.Z. and P.R; investigation, E.Z., V.L., I.L. and J.R.; resources, J.R. and M.K.; data curation, M.K. and P.R.; writing—original draft preparation, P.R.; writing—review and editing, E.Z., V.L., M.K. and P.R.; visualization, M.K., J.R. and P.R.; supervision, E.Z.; project administration, V.L.; funding acquisition, E.Z. and I.L. All authors have read and agreed to the published version of the manuscript.

Funding: The Slovak Research and Development Agency (Agentúra na Podporu Výskumu a Vývoja) supported this work under the contract no. APVV-18-0027 titled "New methods development for reliability analysis of complex system".

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Hurst, S.L. Multiple-Valued Logic. Its Status and Its Future. *IEEE Trans. Comput.* **1984**, *33*, 1160–1179. [CrossRef]
- 2. Beckett, P. Towards a Reconfigurable Nanocomputer Platform. In Proceedings of the ACSAC: 7th Asia-Pacific Computer Systems Architectures Conference, Melbourne, Australia, 4–7 February 2002.
- 3. Vudadha, C.; Srinivas, M.B. Design Methodologies for Ternary Logic Circuits. In Proceedings of the ISMVL: IEEE 48th International Symposium Multiple-Valued Logic, Linz, Austria, 16–18 May 2018; pp. 192–197.
- Choi, B.; Shukla, K. Multi-Valued Logic Circuit Design and Implementation. *Int. J. Electron. Electric. Eng.* 2015, *3*, 256–262. [CrossRef]
- Surhonne, A.P.; Bhattacharjee, D.; Chattopadhyay, A. Synthesis of Multi-Valued Literal using Łukasiewicz logic. In Proceedings of the ISMVL: IEEE 48th International Symposium Multiple-Valued Logic, Linz, Austria, 16–18 May 2018; pp. 204–209.
- 6. Boole, G. An Investigation of the Laws of Thought, 1849; Dover: New York, NY, USA, 1954; reprint.
- 7. Shannon, C.E. The synthesis of two-terminal switching circuits. Bell Syst. Tech. J. 1949, 28, 59–98. [CrossRef]
- 8. Reis, A.I.; Drechsler, R. Advanced Logic Synthesis; Springer: Berlin, Germany, 2018.
- 9. Yanushkevich, S.; Miller, D.M.; Shmerko, V.; Stankovic, R.S. *Decision Diagram Techniques for Micro- and Nanoelectronic Design Handbook*; CRC Press: Boca Raton, FL, USA, 2005.
- 10. Post, E.L. Introduction to a general theory of elementary propositions. *Am. J. Math.* **1921**, *43*, 163–185. [CrossRef]
- Lukasiewicz, J. O Logice Trjwartos'ciowej, Ruch filozoficzny, 5, 1920, 170171; English Translation "On Three-Valued Logic"; Selected Works by Jan Lukasiewicz; Borkowski, L., Ed.; NorthHolland: Amsterdam, The Netherlands, 1970; pp. 87–88. (In Polish)
- 12. Bernstein, B.A. Modular representation of finite algebras. In Proceedings of the International Congress of Mathematicians, Toronto, ON, Canada, 11–16 August 1924; Volume 1, pp. 207–216.
- 13. Berlin, R.D. Synthesis of N-valued switching circuits. IRE Trans. Electron. Comput. 1958, 7, 52–56. [CrossRef]
- 14. Green, D.H. Ternary Reed-Muller Switching Functions with Fixed and Mixed Polarity. *Int. J. Electron.* **1989**, 67, 761–775. [CrossRef]
- 15. Kukharev, G.; Shmerko, V.; Zaitseva, E. *Multiple-Valued Data Processing Algorithms and Systolic Processors*; Nauka i Technika: Minsk, Belarus, 1990.
- 16. Webb, D.L. Generation of any *N*-valued logic by one binary operator. *Proc. Natl. Acad. Sci. USA* **1935**, *21*, 252–254. [CrossRef]

- 17. Epstein, G.; Frieder, G.; Rine, D.C. The development of multiple valued logic as related to computer science. *Computer* **1974**, *7*, 20–32. [CrossRef]
- 18. Keszocze, O.; Wille, R.; Drechsler, R. *Exact Design of Digital Microfluidic Biochips*; Springer: New York, NY, USA, 2017.
- 19. Komolov, V.P.; Roshal, A.S. Logic circuits with ternary parametrons. *Radiophys. Quantum Electron.* **1965**, *8*, 129–133. [CrossRef]
- 20. Edson, W.A. Frequency memory in multi-mode oscillators. IRE Trans. Circuit Theory 1955, 2, 58-66. [CrossRef]
- 21. Anderson, D.J.; Deitmeyer, D.L. A magnetic ternary device. *IEEE Trans. Electron. Comput.* **1963**, *12*, 911–914. [CrossRef]
- 22. Brusentzov, N.P.; Maslov, S.P.; Rozin, V.P.; Tishulina, A.M. The SETUN small automatic digital computer. *Vest. Mosc. Univ.* **1962**, *4*, 3–12.
- 23. Temel, T.; Morgul, A. Implementation of Multi-Valued Logic Gates Using Full Current-Mode CMOS Circuits. *Analog Integr. Circuit. Signal. Process.* **2004**, *39*, 191–204. [CrossRef]
- 24. Heung, A.; Mouftah, H.T. Depletion/enhancement CMOS for a Lower Power Family of Three-valued Logic Circuits. *IEEE J. Solid State Circuits* **1985**, *20*, 609–616. [CrossRef]
- 25. Rich, D.A. A Survey of Multivalued Memories. IEEE Trans. Comput. 1986, 35, 99–106. [CrossRef]
- 26. Yasuda, Y.; Tokuda, Y.; Zaima, S.; Pak, K.; Nakamura, T.; Yoshida, A. Realization of Quaternary Logic Circuits by *n*-channel MOS Devices. *IEEE J. Solid State Circuits* **1986**, *21*, 162–168. [CrossRef]
- Waho, T.; Chen, K.J.; Yamamoto, M. A novel multiple-valued logic gate using resonant tunneling devices. In Proceedings of the ISMVL: IEEE 29th International Symposium Multiple-Valued Logic, Freiburg, Germany, 20–22 May 1999; pp. 2–8.
- 28. Uemura, T.; Baba, T. Demonstration of a novel multiple-valued T-gate using multiple-junctions surface tunnel transistor and its applications to three-valued data flip-flop. In Proceedings of the ISMVL: IEEE 30th International Symposium Multiple-Valued Logic, Portland, OR, USA, 23–25 May 2000; pp. 305–310.
- 29. Appenzeller, J. Carbon Nanotubes for High-Performance Electronics-Progress and Prospect. *Proc. IEEE* 2008, *96*, 201–211. [CrossRef]
- Raychowdhury, A.; Roy, K. Carbon-Nanotube-based Voltage-mode Multiple-valued logic design. *IEEE Trans.* Nanotechnol. 2005, 4, 168–179. [CrossRef]
- Sedighiani, S.; Kazemi, A. An Energy-Efficient Quaternary Serial Adder for Nanoelectronics. In Proceedings of the ISMVL: IEEE 48th International Symposium Multiple-Valued Logic, Linz, Austria, 16–18 May 2018; pp. 44–49.
- 32. Umredkar, N.W.; Gaikwad, M.A.; Dandekar, D.R. Review of Quaternary Adders in Voltage Mode Multi-Valued Logic. *Int. J. Comput. Appl.* 2013, 2013, 17–21.
- 33. Nielsen, M.A.; Chuang, I.L. *Quantum Computation and Quantum Information*; Cambridge University Press: Cambridge, UK, 2000.
- 34. Mohammadi, M. Radix-independent, efficient arrays for multi-level *n*-qudit quantum and reversible computation. *Quantum Inf. Process.* **2015**, *14*, 2819–2832. [CrossRef]
- 35. Weiss, R.; Basuy, S. The Device Physics of Cellular Logic Gates. In Proceedings of the ACSAC: 7th Asia-Pacific Computer Systems Architectures Conference, Melbourne, Australia, 4–7 February 2002.
- Baudry, L.; Lukyanchuk, I.; Vinokur, V.M. Ferroelectric symmetry-protected multibit memory cell. *Sci. Rep.* 2017, 7, 1–7. [CrossRef]
- 37. Lukyanchuk, I.; Sharma, P.; Nakajima, T.; Okamura, S.; Scott, J.F.; Gruverman, A. High-Symmetry Polarization Domains in Low-Symmetry Ferroelectrics. *Nano Lett.* **2014**, *14*, 6931–6935. [CrossRef]
- Lahoche, L.; Luk'yanchuk, I.; Pascoli, G. Stability of vortex phases in ferroelectric easy-planes nano-cylinders. *Integr. Ferroelectr.* 2008, 99, 60–66. [CrossRef]
- Baudry, L.; Sené, A.; Luk'yanchuk, I.; Lahoche, L.; Scott, J.F. Polarization vortex domains induced by switching electric field in ferroelectric films with circular electrodes. *Phys. Rev.* 2014, 90, 024102. [CrossRef]
- 40. Stankovic, R.; Astola, J.T.; Moraga, C. Representations of Multiple-Valued Logic Functions. *Synth. Lect. Digit. Circuits Syst.* **2012**, *7*, 1–168. [CrossRef]
- 41. Bellman, R. Introduction to Matrix Analysis, 2nd ed.; McGraw-Hill: New York, NY, USA, 1970.
- Munirul, H.M.; Kameyama, M. Ultra-Fine-Grain Field-Programmable VLSI Using Multiple-Valued Source-Coupled Logic. In Proceedings of the ISMVL: IEEE 34th International Symposium Multiple-Valued Logic, Toronto, ON, Canada, 19–22 May 2004; pp. 26–30.

- 43. Zilic, Z.; Vranesic, Z.G. Multiple-valued logic in FPGAs. In Proceedings of the 36th Midwest Symposium on Circuits and Systems, Detroit, MI, USA, 16–18 August 1993; pp. 1553–1556.
- 44. Epstein, G. Multiple-Valued Logic Design: An Introduction; Taylor & Francis Group: Abingdon, UK, 1993.
- 45. Amaru, L.; Gaillardon, P.E.; De Micheli, G. Majority-Inverter Graph: A New Paradigm for Logic Optimization. *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.* **2016**, *35*, 806–819. [CrossRef]
- 46. Xia, Y.S.; Wang, L.Y.; Zhou, Z.G.; Ye, X.E.; Hu, J.P. Novel Synthesis and Optimization of Multi-Level Mixed Polarity Reed-Muller Function. *J. Comput. Sci. Technol.* **2015**, *20*, 895–900. [CrossRef]
- 47. Sasao, T. On the Optimal Design of Multiple-valued PLA's. IEEE Trans. Comput. 1989, 38, 582–592. [CrossRef]
- Model-Based Design: From Concept to Code. Available online: https://www.mathworks.com/products/ simulink.html (accessed on 17 September 2019).
- 49. Tarun, K.; Hashmi, M.S. Multiple valued current mode logic circuits. In Proceedings of the IMPACT: International Conference on Multimedia, Signal Processing and Communication Technologies, Aligarh, India, 24–26 November 2017; pp. 65–69.
- 50. Pertsev, N.A.; Zembilgotov, A.G.; Tagantsev, A.K. Effect of Mechanical Boundary Conditions on Phase Diagrams of Epitaxial Ferroelectric Thin Films. *Phys. Rev. Lett.* **1988**, *80*, 1988–1991. [CrossRef]
- 51. Rafiev, A.; Mokhov, A.; Burns, F.; Murphy, J.; Koelmans, A.; Yakovlev, A. Mixed Radix Reed-Muller Expansions. *IEEE Trans. Comput.* **2012**, *61*, 1189–1202. [CrossRef]
- 52. Balch, M. Complete Digital Design: A Comprehensive Guide to Digital Electronics and Computer System Architecture; McGraw-Hill: New York, NY, USA, 2003.
- 53. Fujioka, Y.; Kameyama, M.; Lukac, M. A dynamically reconfigurable VLSI processor with hierarchical structure based on a micropacket transfer scheme. In Proceedings of the International Conference on Information and Digital Technologies, Zilina, Slovakia, 5–7 July 2017; pp. 132–136.
- 54. Sasao, T. Easily Testable Realizations for Generalized Reed-Muller Expressions. *IEEE Trans. Comput.* **1997**, 46, 709–716. [CrossRef]
- Moraga, C.; Stanković, M.; Stanković, R.S. Spectral Invariant Operations in the p-valued Spectral Domain. In Proceedings of the ISMVL: IEEE 49th International Symposium Multiple-Valued Logic, Fredericton, NB, Canada, 21–23 May 2019; pp. 67–72.
- Shimabukuro, K.; Kameyama, M. Fine-Grain Pipelined Reconfigurable VLSI Architecture Based on Multiple-Valued Multiplexer Logic. In Proceedings of the ISMVL: IEEE 47th International Symposium Multiple-Valued Logic, Novi Sad, Serbia, 22–24 May 2017; pp. 19–24.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).