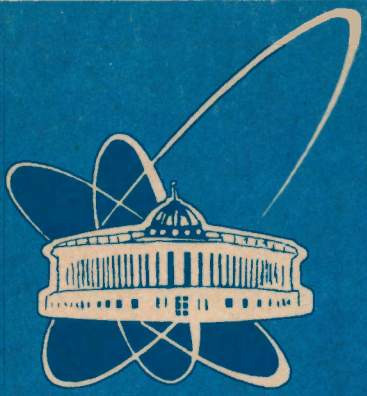

объединенный
институт
ядерных
исследований
дубна

E10-94-513

I.N.Alexandrov, V.M.Kotov, N.M.Nikityuk, R.Pose

NEW TYPE OF PROGRAMMED MEMORY WITH ALGEBRAIC STRUSTURE

Submitted to the International Symposium on Mathematical Foundations of Computer Science, August 28 — September 1, 1995, Prague, Czech Repablic

1994

In Galois fields, full of flowers
Primitive elements dance for hours
Climbing sequentially through the trees
And shouting occasional parities.
S.B. Weinstein.

1 Introduction

Possibility of an application of polynomial forms for constructing Galois fields $GF(2^m)$ switching functions (GSF), the perspective of their using for synthesis of universal dynamically programmable logic modules and modes of layouts synthesis both for completely and incompletely defined functions were considered in [1,2]. The possibility of using GSF for constructing the new types of EPROMs and PROMs is considered in this paper.

As well known EPROMs and PROMs are widespread used in modern computer technique. However it demands the using of complicated and expensive technology. They have complicated logic structure. The typical EPROM-module includes decoder, memory array, amplifier-shaper and so on. All of this technique increases delay time and cost of module. In this paper the principally new approach to the synthesis of a EPROM and PROM which is based on the representation of switching function of m arguments in elements of Galois GF(2^m) field is suggested. As shown, the representation of switching functions in the form of Galois field polynom is promising for the calculation and synthesis of EPROM. The active part of our EPROM is purely combinatorial and has no decoders, amplifiers, therefore, signal delays in them are minimal. Besides the computer algebra application for the calculation of the Galois switching functions (GSF) in the field GF(2^m) is used.

2 Basic attributes and definitions

As the Galois field GF(2^m) is a natural extension of a Boolean field, the representation of GSF as polynomials where both the variables and the coefficients are field elements has a number of fundamental advantages.

1. It is possible to perform algebraic operations on GSFs, which simplifies the problem of minimization and its formal representation.

2. In a Galois field there are operations like addition, multiplication and division which give additional advantages over Boolean fields.

3. The representation of switching functions in the form of polynomial makes it possible to use standard programming systems to calculate complicated logic structures.

4. Since the input and output states of a logic structure are coded by field elements, the following state can be represented as a polynomial function of the current state and the current output.

5. The description of multivalued and multilevel schemes is very compact, which simplifies their theoretical study [3].

The case p=2 is considered, however all results are valid for other simple p, therefore the actuality of the given direction is increasing with the creation of multivalued logic devices [2]. Let us introduce a field of coefficients GF(2) with elements 0 and 1, modulo 2 addition as a field's addition operation and a conjunction as a field's multiplication operation. In this field the operations of addition and subtraction are completely identical. Introduce over GF(2) a field GF(2^m) as a field of polynoms with coefficients from GF(2) and degree less than m. The primitive root of irreducible polynom of degree m with its all degrees will play the role of variables. The degrees of the primitive polynom cover all the field GF(2^m). The addition operation in this field is the usual operation of polynom's addition, where the coefficients are added in the field GF(2). Multiplication is a multiplication of polynoms in modulo of a polynom. The first m degrees of primitive root α^0,α,α^2, α^3,...,α^{m-1} are linearly independent and can be given as a basis of the Galois field. This means that any field 's element X can be represented in the form:

$$X = a_0 * \alpha^0 + a_1 * \alpha + \cdots + a_{m-1} * \alpha^{m-1}, \tag{1}$$

where $a_i \in 0, 1$. If we take any binary number $(a_0, a_1, \ldots, a_{m-1})$ of length m as a set of coefficients a_i from (1), we may interpret it as the element of the Galois field.

Menger has shown [4] that any switching function f(X) can be represented in the form (and in a unique way)

$$f(X) = f(0) + \sum_{i=1}^{2^m-1} G_i X^i, \tag{2}$$

$$G_i = \sum_{j=1}^{2^m-1} \alpha_j^{-i}(f(0) - f_j), \tag{3}$$

where $f_j = f(\alpha_j)$, and $\alpha_j = \alpha^j$ - j-th degree of α.

3 EPROM layout synthesis

Starting from (1), when performing raising to a power by means of multiplication of (1) by itself (appropriate number of times), one gets all the X^i from (2) in polynomial form. If for a concrete switching function coefficients G_i are calculated from formula (3) and its values with degrees of X in polynomial form are substituted into (2), after eliminating of equal values we can get a GF(2) polynom system. Each polynom will be represented as a coefficient of basic element α^i. However,in this case the values of G_i will be hardwired in layout.

We separate EPROM to the active and passive parts. The active part of EPROM changes its states in each clock but the passive one reserves all preliminary loaded adjusting coefficients and does not change its states when process of access to memory. The variable X is the input for active part of EPROM. Coefficients G_i are loaded into passive part of EPROM (file registers) in sequential mode. G_i are represented in the general polynomial form through basis as we have represented X in (1):$G_i = b_{i_0}\alpha^0 + \cdots + b_{i_{m-1}}\alpha^{m-1}$. The given expressions with the expressions for all X^i are substituted into (2). As a result we will get the required polynom for EPROM, in which X and all G_i are variables given via expansion in the basis. On its basis one can synthesize EPROM, X and all G_i being inputs for layout, but G_i are loaded from passive part of EPROM. To adjust EPROM to a concrete function it is enough to calculate values of all G_i for this function from formula (3) and to load them into file registers. The values of G_i are directed to the inputs of layout together with the values of X during operating EPROM. The expressions obtained for X^k are bulky, therefore all the calculations are performed

on computer. Using the fact that $X^k = X \times X^{k-1}$, the layout of EPROM can be simplified by means of increasing the number of cascades in it, i.e. by increasing the delay time [2]. For example, only even parity degrees of X can be realized (their expressions in average simpler), and odd parity degrees are obtained on the further cascade by means of multiplication $X^{2l+1} = YX$, where Y is outputs from layout of X^{2l} and l=1,2,...,$(2^{m-1}-1)/2$.

Example 1. For the Galois field GF(2^4) and the irreducible polynom $x^4 = x + 1$ formula (2) has the following form:

$$F(X) = F(0) + G_1X + G_2X^2 + \cdots + G_15X^15 = F(0) + [G_1X + G_2X^2 + G_3X^3] +$$
$$X^4[G_4 + G_5X + G_6X^2 + G_7X^3] + X^8[G_8 + G_9X + G_{10}X^2 + G_{11}X^3] +$$
$$X^{12}[G_{12} + G_{13}X + G_{14}X^2 + G_{15}X^3].$$

Substituting the expressions for G_i and X^k in polynomial form we get a two-cascade layout of the calculation of any 4-input switching function (4 input EPROM). The first cascade represents the calculation of the expressions in brackets, the second is the realization of the remaining operations of multiplication and addition. However, the expressions for X^3, X^{12} are still large. Using only the expressions for X, X^2, X^4, X^8, we get the following expressions for F(X):

$$F(X) = F(0) + G_1X + G_2X^2 + G_3X^2X + G_4X^4 + G_5X^4X + G_6X^4X^2 +$$
$$G_7X^4X^2X + G_8X^8 + G_9X^8X + G_{10}X^8X^2 + G_{11}X^8X^2X + G_{12}X^8X^4 +$$
$$G_{13}X^8X^4X + G_{14}X^8X^4X^2 + G_{15}X^8X^4X^2X =$$
$$F(0) + [X^8(G_8 + G_{12}X^4)] + [G_1 + X^8(G_9 + G_{13}X^4)]X + [G_2 + G_3X +$$
$$X^8(G_{10} + G_{11}X + G_{14}X^4)]X^2 + [G_4 + G_5X + (G_6 + G_7X)X^2]X^4 +$$
$$G_{15}[(X^8X^4)(X^2X)].$$

The general layout of suggested EPROM is shown on fig.1. It includes the coincidence matrix with $m + m(2^m - 1)$ inputs, file register having $m(2^m - 1)$ registers, sequential interface and group of modulo 2 adders. The coincidence matrix and modulo 2 adders are active parts of the EPROM. The sequential interface is used for loading the adjusting coefficients into file register. Two modes of suggested modules can be constructed. The first is in EPROM type and the second one is in the view of PROM. Technology of creating

file register must be energy dependent in the first mode because the data must be saved after source power turn off. However if file register is a usual trigger register or dynamic memory, all coefficients must be loaded at the beginning of the work from host computer. Since the file register does not demand high speed it may be realized by means of cheep technology. We need not file register in the second case. It is enough to program the values of logic levels (high-low) for reserving of beforehand calculated coefficients for instance through the fuse-programmable method. We can see from fig.1 that delay time T_{EPROM} is $T_{EPROM} = T_c + T_m$, where T_c is the delay time in coincidence matrix and T_m is the delay in the modulo 2 adder. As known a modulo 2 adder realized in usual AND-gates. That is the reason of high speed of suggested type of EPROMs and PROMs. The active part has polynomial structure, all expressions are known in analytical form that is the reason to use computer algebra and analytical calculation for their synthesis in hardware. The analog schemes are absent in the active part of our unitsthat lead to high reliability of suggested EPROM.

For the calculation of G_i the EPROM layout itself may be used. It is sufficient to load into registers of preserving of G_i the values of the functions F_i for all i=1,...,$2^m - 1$ and sequentially put into the inputs of layout values of $X = \alpha^{-k}$, where k=1,...,$2^m - 1$ (for $k = 2^m - 1$ value f(0) is added). On the output we get the required values of G_i with the minimal time delay. It is possible also to use systolic structures for the calculation of G_i. If for readjusting EPROM from one function to another only a small number of values of outputs is changed, changes of the volume of the calculation for (3) will be drastically reduced. So by changing one value of the output at the point α_k from the old $F_{k_{old}}$ to the new $F_{k_{new}}$ expression (3) becomes as follows: $G_{i_{new}} = G_{i_{old}} + (F_{k_{new}} - F_{k_{old}})\alpha_k^{-i}$, for all $i = 1, ..., 2^m - 1$.

4 Construction of EPROM through EPROM of a less number of inputs

With increasing m the complexity of the expression for EPROM increases, therefore it would be useful to get values of any switching function of m inputs by means of EPROM of a lesser number of inputs/outputs, in other words, working in Galois fields of lesser order. Let us consider any function

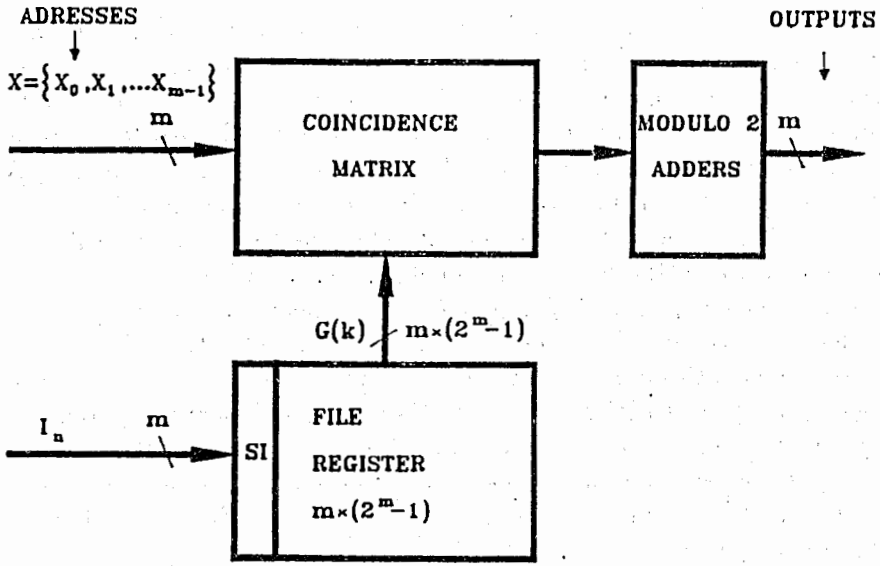

Fig.1. General block scheme of EPROM. I_n – adjusting coeficients.

F with m inputs and m outputs. The domain of its definition X is a set of all binary of m length. Let Y be a domain of values of F (the length m binary number). We assume that the inputs/outputs of the function are from left to right. Let us call for convenience m_1 left inputs/outputs lower, the remaining $m_2 = m - m_1$ inputs/outputs higher (we have $m > m_1 \geq m_2$). Partitioning X into 2^{m_2} classes K_i is as follows: element x from X belongs to K_i if its higher bits represent number i in binary form. Each class has 2_1^m elements. They differ only in lower bits. In each K_i let us define a pair of functions F_{i_1} and F_{i_2} as follows. Any x belonging to K_i may be represented in the form $(x_1, x_2, \ldots, x_{m1}, x_{m1+1}, \ldots, x_m)$, $(x_{m+1}, \ldots, x_m)$ being a constant for any x from K_i. If $y = F(x)$, then y can be also represented in the form $(y_1, y_2, \ldots, y_{m_1}, y_{m_1+1}, \ldots, y_m)$, and in accordance with the definition assume $F_{i_1}(x) = (y_1, \ldots, y_{m_1})$ and $F_{i_2} = (y_{m_1} + 1, \ldots, y_m)$. As for each K_i the high-order bits of inputs values are constant, we may take for any F_{i_l} $(l = 1, 2)$ as

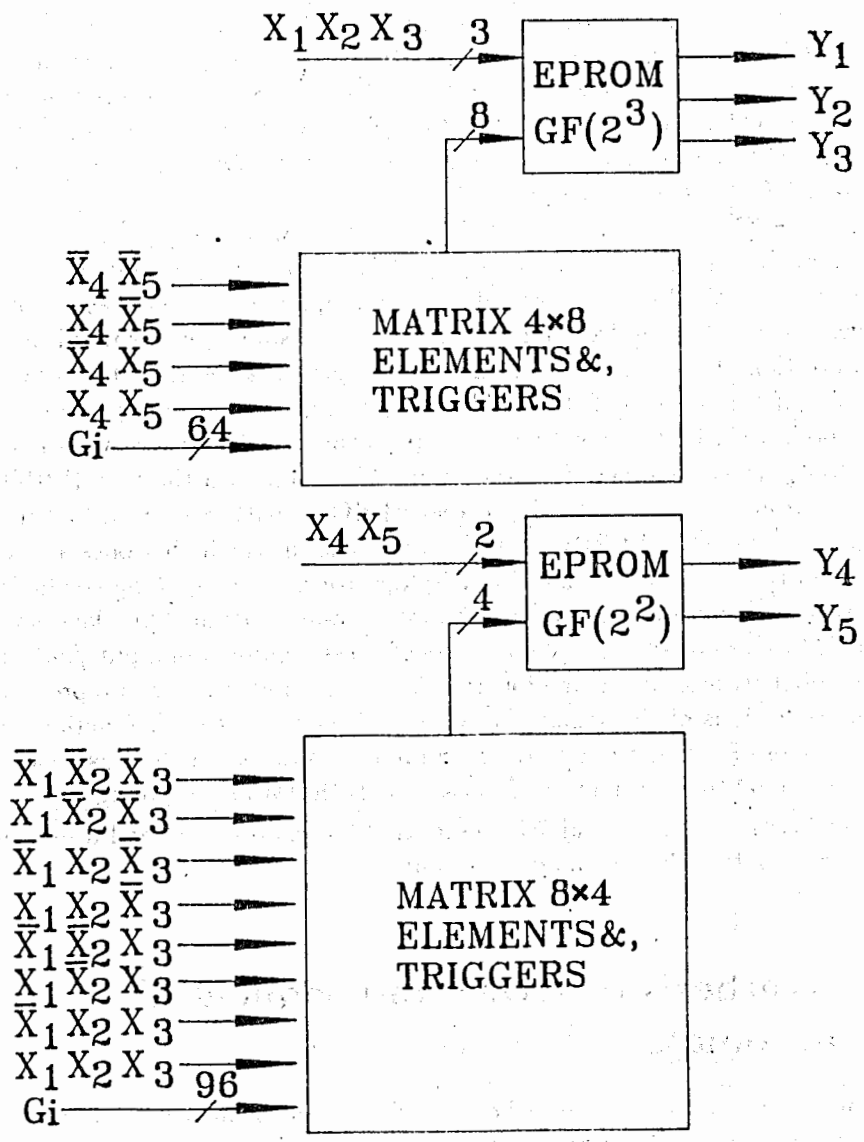

Fig.2. Global structural scheme for m=5, m_1=3, m_2=2. x_i –inputs, y_i –outputs, G_i, G_j –adjusting coefficients.

inputs only lower bits, and consequently to realize it in m_1 inputs/outputs of EPROM. For each F_{i_l} let us calculate from expression (3) coefficients $G_{i_l j}$ for $i_l = 0, \ldots, 2^{m_2} - 1$, $j = 1, \ldots, 2^{m_1} - 1$. Designate as x'_i an input element from K_i that has lower bits equaled zero. Then we may get the values of the function F (separately lower and higher bits) by formulae analogous to (2) through GF(2^{m_1} field operations:

$F_{i_l}(x) = \sum_{i=0}^{2^{m_2}-1} F(x'_i)p(i,x) + \sum_{j=1}^{2^{m_1}-1} G'_{i_l j} X^j, \quad G'_{i_l j} = \sum_{i_l=0}^{2^{m_2}-1} G_{i_l j p(i,x)},$

where $X = (x_1, \ldots, x_{m_1})$, $p(i,x)$ equal 1, when the higher bits of x coincide with i in the binary representation and equal zero otherwise. In other words, $p(i,x)$ is a term from all $(x_{m_1}+1, \ldots, x_{m_2})$ and x_k is negated if in the k-th position of the number i in the binary representation is 0. To obtain all the bits of the outputs, 2 EPROMs with m_1 inputs/outputs are necessary. The mode with EPROM with m_1 inputs/outputs and EPROM with m_2 inputs/outputs is possible. Then G_{ij} from F_{i_1} is put into the first EPROM, but G_{ij} from F_{i_2} is put into the second EPROM (with m_2 inputs/outputs), besides for the second EPROM the lower bits act as the higher ones and vice versa. In this case the memory capacity for storing the adjusting coefficients is equal to the storage for EPROM for m inputs/outputs, but layouts for EPROM themselves are simpler because of a lesser number of inputs/outputs. The global structural scheme for $m = 5, m_1 = 3$ and $m_2 = 2$ is represented in figure 2. It is obvious that the problem is simplified for a function with the number of outputs less than the number of inputs. In this case one can take m_1 equal to a number of outputs, and EPROM for obtaining F_{i_2} is not needed. So if the structural scheme in fig.2 for m=5 inputs had 3 outputs, 8×4 matrix for GF(2^2) would be absent.

5 Synthesis of ROM and sequential automata

Let us show the rules for synthesis of ROM and sequential automata as particular case of PROM in the concrete example.

Example 2. Let us calculate the scheme of sequential automata and ROM simultaneously having capacity 2^4 for bit words. The Galois field

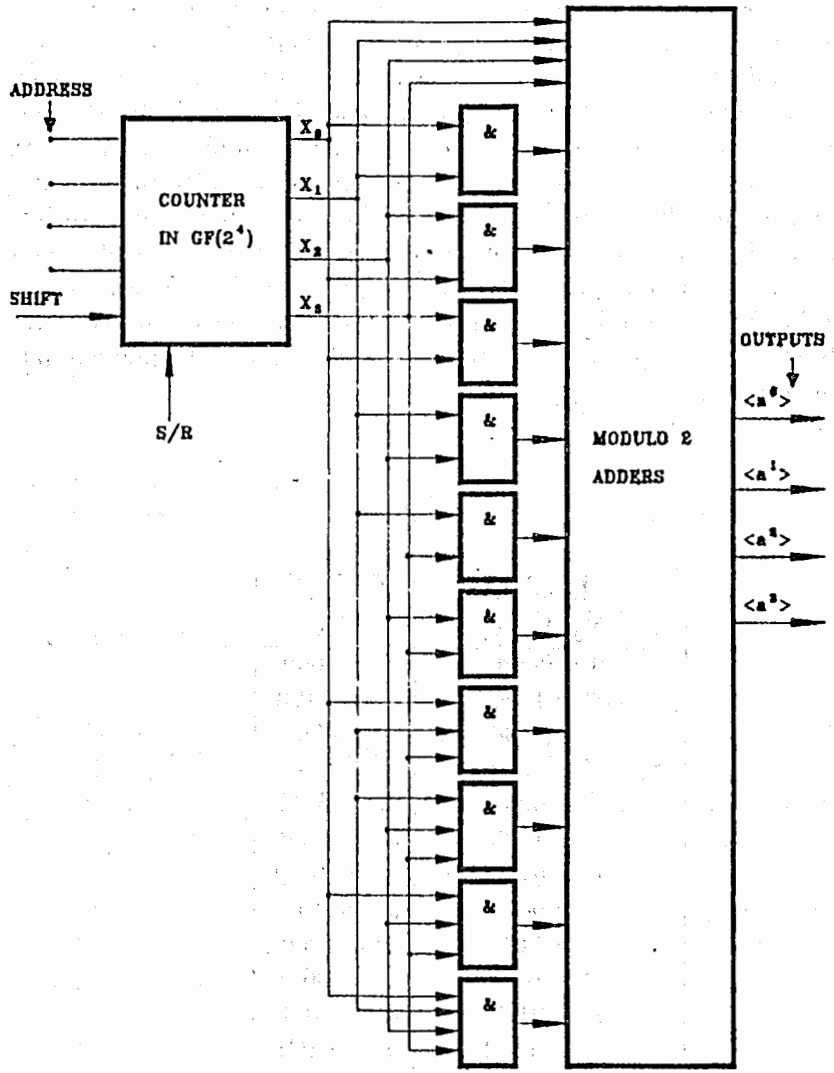

Fig.3. Block scheme of sequential automata/ROM.
S/R=1 – mode of sequential automata,
S/R=0 – mode of ROM.

$GF(2^4)$ elements generated over the irreducible polynomial $X^4 + X + 1$ arrange correspondingly in increasing order of their powers at the inputs in case of sequential automata. At the outputs we obtain the same elements in the given sequence (as shown in Table). The Galois field $GF(2^4)$ elements in increasing order of their powers can be rather simply obtained with the help of the counter in the $GF(2^4)$. It is a shift register with the logical opposite connections. If we carry unit into the low-order digit and zeros into the other ones the successive shifts of the register will give us presentation of the a^1 element powers and the root of the polynomial $X^4 + X + 1$ as they are shown in Table on the left. To construct a scheme the sequential automata and ROM it is necessary to calculate 15 coefficients in the polynomial GSF representation of 4 variables.

Table

Inputs	Outputs
$X = X_0, X_1, X_2, X_3$	$F(X)$
$0 = 0000$	$0 = 0000$
$a^0 = 1000$	$a_1 = 0100$
$a^1 = 0100$	$0 = 0000$
$a^2 = 0010$	$a^7 = 1101$
$a^3 = 0001$	$a^5 = 0110$
$a^4 = 1100$	$a^{10} = 1110$
$a^5 = 0110$	$a^{11} = 0111$
$a^6 = 0011$	$a^{13} = 1011$
$a^7 = 1101$	$0 = 0000$
$a^8 = 1010$	$a^3 = 0001$
$a^9 = 0101$	$a^{14} = 1001$
$a^{10} = 1110$	$a^3 = 1001$
$a^{11} = 0111$	$0 = 0000$
$a^{12} = 1111$	$a^8 = 1010$
$a^{13} = 1011$	$a^4 = 1100$
$a^{14} = 1001$	$a^0 = 1000$

By calculation of the $G(k)$ coefficients and elimination of similar terms on computer we get the following switching functions.

$$X_0X_1 + X_2 + X_0X_2 + X_0X_3 + X_1X_2 + X_1X_3 + X_0X_1X_3 + X_1X_2X_3 +$$

$$X_0X_1X_2X_3 < a^0 >$$

$$X_0 + X_2 + X_3 + X_1X_3 + X_0X_1X_3 + X_1X_2X_3 < a^1 >$$

$$X_3 + X_0X_1 + X_0X_3 + X_1X_2 + X_1X_3 + X_1X_2X_3 + X_0X_1X_2X_3 < a^2 >$$

$$X_2 + X_1X_3 + X_0X_1X_3 + X_0X_2X_3 < a^3 >$$

With the aid of these functions a scheme of sequential automata shown in fig.3 was obtained. Such schemes can be used to get a given sequence of binary digits for example in microprogramming control devices. It is enough to turn off feedbacks and to load parallel-in according address codes to the inputs of address counter for using this scheme.

6 Conclusion

The new variant of a programmable memory unit is suggested. This one may have higher speed and reliability in compare with already known units because an active part of such unit consist only from AND-gate elements. Suggested EPROM can be used as module of associative processors or universal dynamic programmable logic module. The basic problem is as follows: when the number m is big the complement researches are required due demand of large calculation of adjusting coefficients in GSF polynomial form and aids of optimization multistage schemes (for example the delay times and required AND-gates). The support is needed for this researches.

References

1. Aleksandrov I.N., Kotov V.M., Nikityuk N.M., Pose R. Use of Computer Algebra for Calculation Switching Functions in Galois Field GF(2^m).

"The Rhine Workshop on Computer Algebra". Karlsruhe, Germany, March 22-24, 1994,pp.216-219. Editor J.Calmet.

2. Aleksandrov I.N., Kotov V.M., Nikityuk N.M. Application of switching function in Galois field GF(2^m) for synthesis universal dynamically programmable logic modules. Be published in "Avtomatika i telemekhanika", Moscow, No.4, 1995.

3. Benjauthrit B., Reed I. Galois Switching Functions and Their Applications - "IEEE Trans. on Comput.", 1976, Vol.C-25, No 1, pp.78-86.

4. K.S.Menger. A Transform for Logic Networks. - IEEE Transactions on COMPUTERS,1969, vol.C-18, N^{0}3, p.241-250.

Received by Publishing Department
on December 28, 1994.

Александров И.Н. и др. E10-94-513

Новый тип программируемой памяти с алгебраической структурой

Предложен принципиально новый подход к синтезу EPROM и PROM, базирующийся на полиномиальном представлении переключательных функций m переменных в полях Галуа $GF(2^m)$. EPROM делится на активную и пассивную части. Активная часть EPROM меняет свои состояния в каждом такте, в пассивной хранятся все предварительно загруженные настроечные коэффициенты, не меняющие свои состояния в процессе доступа к памяти. Коды адресов являются входами для активной части EPROM. Предварительно подсчитанные по формуле Менгера настроечные коэффициенты загружаются в пассивную часть EPROM (файловый регистр) в последовательной моде. Активная часть предлагаемой EPROM содержит только AND-вентили и не имеет декодеров и предусилителей, поэтому задержки в ней минимальны. Для больших m применяется суперпозиция в полях Галуа меньшей размерности. Для вычисления коэффициентов переключательных функций применяется компьютерная алгебра.

Работа выполнена в Лаборатории высоких энергий ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна, 1994

Alexandrov I.N. et al. E10-94-513

New Type of Programmed Memory with Algebraic Structure

The principally new approach to the synthesis of EPROMs and PROMs which is based on the representation in a polynomial form of switching functions of m arguments is elements of Galois GF (2^m) field is suggested. We separate EPROM to the active and passive parts. The active part of EPROM changes its states in each clock but the passive one reserves all preliminary loaded adjusting coefficients and does not change its states in the process of time access to memory. A code of address is the input for active part of EPROM. Polynomial coefficients that are preliminary calculated through the formula of Menger are loaded into passive part of EPROM (file registers) in the sequential mode. The active part of our EPROM includes AND-gates only and has no decoders, amplifiers, therefore, signal delays in them are minimal. The superposition of Galois fields GF (2^{m1}) and GF (2^{m2}), where $m = m1 + m2$, is applied for large m. Besides, the computer algebra application for the calculation of the Galois switching functions in the field GF (2^{m}) is used.

The investigation has been performed at the Laboratory of High Energies, JINR.

Preprint of the Joint Institute for Nuclear Research. Dubna, 1994