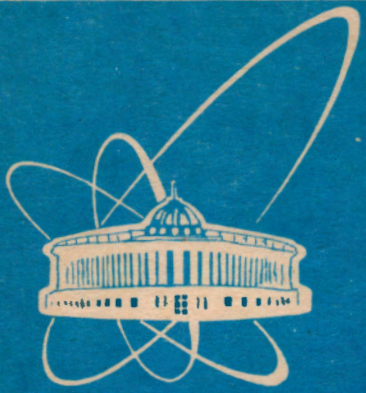

объединенный
институт
ядерных
исследований
дубна

E10-94-514

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

COMMUTATION INFORMATION IN GALOIS FIELDS $GF(2^m)$

Submitted to the Eleventh International Symposium on Applied Algebra, Algebraic Algorithms and Error Correcting Codes, Paris, July 17—21, 1995 (AAECC 11)

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

One of the most fundamental functions in data processing is data switching (or line switching). The role of this function is essentially increasing in modern digital systems. It is well known that the switching speeds of computer devices are approaching a limit, any further improvement of computer throughput can be due to increasing the number of bits processed simultaneously. In a crossbar switch every input port can be connected with a free output port without blocking. The crossbar network can provide any-to-any routing with very small delay, but becomes increasingly costly for large numbers of processors. The reason for the cost is that the switching elements, may have to be complex, and there are N^2 of them, where N - the number of processors. There are two possibilities to make the crossbar network [1]. One is a network using dual port memories. Number of dual port memories is number of input multiplied by number of output. It is not practical to use dual port memory modules for large scale network. A crossbar 2×2 switch is widely used for the construction of networks [1,2] . But the cost of network is proportional to $Nlog(N)$; and the transit delay, to $log(N)$.

As noted in [3], there are no switches with reasonable cost and performance which have prevented the growth of large multiprocessor systems. The crossbar switch [4] is complicated also.

A new method of synthesis of multipoint full crossbar switches is based on the transformation of input information, considered as a binary unitary code, to codes over which simple logical operation is executed. Then this information is decoded. This method allows one to construct full crossbar switches having useful properties. One class of switches has correcting capabilities. This method allows one to construct switches for analog and light pulses also. Two methods of the synthesis are described: switches over Galois field GF(2^m) and with correcting capability. These ones have modulo structure, small delays and simple logical structure is needed for their real-

ization. And besides, the switches have algebraic structure what makes their synthesis easier.

2 Crossbar switches over Galois field GF(2^m)

In this section we consider the method of switches construction which is based on the properties of Galois field GF(2^m) widely used for the construction of effective error correcting codes which correct $t \leq n/4$ mistakes in n-binary words (BCH-codes). Other areas of application of finite fields are known. The authors have shown that using the algebraic theory of BCH-codes one can construct coordinate processors and majority coincidence circuits for large inputs [5,6].

2.1 Some rules of execution of operation in Galois field GF(2^m)

In order to make this abstract easier for the readers who are not familiar with the rules of the execution operations over GF(2^m) elements, necessary information on these rules is present below which has no claim on completeness and generality. Concrete examples are given for $m = 4$. As is known, Boolean algebra is the theoretical base of modern computer technology where the discrete function is equal to 0 or 1 in the binary case. As shown in [7,8], Boolean algebra is the particular case of Galois algebra. Operations in Galois algebra are determined over the main field GF(2) having two elements 0 and 1 and the extended field to m-th power having (2^{m-1}) nonzero elements which are considered as m-bits words of cyclic codes. The number of nonzero elements of the finite field GF(2^m) is equal to some power of its characteristics, i. e. $n = 2^{m-1}$. The number of different elements of the field is called its order. All elements of the finite field can be obtained by means of irreducible polynomials which tables are presented in [9]. We have chosen the irreducible polynomials of the 3rd ($X^3 + X + 1$) and 4th degrees ($F(X) = X^4 + X + 1$) ($m = 4$). It should be noted that sign "+" will be used to denote modulo 2 additions. For $m = 4$ the number of nonzero elements equals 15. Among these elements there are four linearly independent (basis) elements:$a^0 = 1000$, $a = 0100$, $a^2 = 0010$ and $a^3 = 0001$. One of these elements, a^1 , is the root of the polynomial $F(X)$. Then each nonzero

element can be presented as a degree of element a^1 . This means that the multiplicative group of the finite field is cyclic in character. The least positive number n, for which $a^n = a^0 = 1$, is referred to as the order of element a^1. If the order of element a^1 equals n, elements $a^0, a^1, a^2 ... a^{n-2}, a^{n-1}$ are different. Thus, in our example $n = 15$. Therefore, e.g., $a^{30} = (a^{15})^2 = a^0$; $(a^{31}) = a^{15}a^{16} = a^1$, etc. Taking into account that a^1 is the root of the polynomial $F(X)$, the remaining elements of GF(2^4) can be obtained from the equation $a^4 + a + 1 = 0$, i.e. $a^4 = a + 1 = 1100$; $a^2 + a^1 = a^5 = 0110$ and so on. There are at all 15 nonzero elements over GF(2^4) Galois field: $a^0 = 1000$; $a^1 = 0100$; $a^2 = 0010$; $a^3 = 0001$; $a^4 = 1100$; $a^5 = 0110$; $a^6 = 0011$; $a^7 = 1101$; $a^8 = 1010$; $a^9 = 0101$; $a^{10} = 1110$; $a^{11} = 0111$; $a^{12} = 1111$; $a^{13} = 1011$; $a^{14} = 1001$; $a^{15} = 1000 = 1$. It is also convenient to present the field elements A and B as polynomial of degree $m - 1$. For $m = 4$ we have: $A = A^0a_0 + A^1a_1 + A^2a_2 + A^3a_3$ and $B = B^0a_0 + B^1a_1 + B^2a_2 + B^3a_3$, where $A_0 \div A_3$ and $B_0 \div B_3$ are equal to 0 or 1. Thus, if element $A = a^8$, then $A_0 = A_2 = 1$ and $A_1 = A_3 = 0$. Addition and subtraction operations in the field GF(2^m) are equivalent and carried out by modulo 2 addition. The operation of multiplication of two elements is carried out in the same manner as that of ordinary numbers with the difference that this operation is performed by modulo irreducible polynomial.

2.2 Scheme of the one-bit $n \times n$ crossbar switch

The following method for switch construction is suggested. Input bits arriving from n ($n = 2^{m-1}$) senders are presented as unitary code one-bit words which is encoded to Galois field elements in increasing order of their degrees. The outputs of the encoders are connected to the first group of the inputs of the modulo 2 adder. The addresses of the receivers are supplied to the second group of inputs according to the table of addition of two elements in GF(2^m). The outputs of the senders are numbered as 1 to 2^{m-1}. Then the results of addition are decoded. Each decoder has m inputs and 2^m outputs. The output 0 is not used and the corresponding outputs of the decoders are connected. For example, to send one bit from input 2 (position a^1) to receiver 5, it is necessary to deliver the control word a^3 . We get $a^1 + a^3 = a^5 = 0101$ at the outputs adder. As a result, we get a pulse corresponding to a logical one at 5th output of the binary decoder. Fig.1 gives a typical scheme of one (first in order) unit.

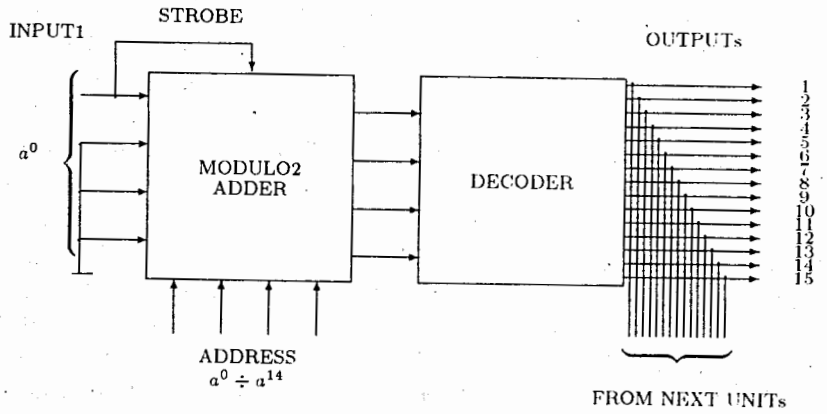

Figure 1: One unit's scheme of one-bit 15×15 crossbar switch

It is clear that the scheme of the encoders is very simple. They are the sources of logical one and logical zero. The delay T of a switch can be calculated from the relation $T = 2T_D + 2T_A$, where T_D - is the delay of the AND-gate which is in the decoder and T_A - the delay of a modulo 2 adder which equals the two delay of the AND-gate. Then the total delay is $4T$ AND-gate. It should be noted that next approach can be suggested to construct a switch having inputs (outputs) which are multiple a degree of two as Galois field GF(2^m) has 2^{m-1} nonzero elements: to use the modular nature of the Galois field. For example, to construct a one-bit 16x16 switch, the GF(2^5) field can be used instead of the Galois field GF(2^4) and modulo 2 can be minimized. It is clear the switches over Galois field GF(2^m) have not possibility of errors correction. The number of control(addresses) bits needed for N-bit $n \times n$ switch is equal to $N \times n \times log_2(n)$. One can recommend the following next irreducible polynomials for the construction of switches for n = 31, 63, 127, 511 and 1023: $X^5 + X^2 + X + 1$; $X^6 + X + 1$; $X^7 + X + 1$; $X^8 + X^4 + X^3 + X^2 + 1$; $X^9 + X + 1$ and $X^{10} + X^3 + 1$.

3 Crossbar switches with error correcting capabilities

As known the error correcting codes are wide-spread used in automatics and computing technique.These codes are applied especially effectively for checking errors in memory units. As we show below the error correcting codes can be effectively used in **suggested** by us switches for construction a new class of switches with correcting capabilities.

3.1 Use of Hamming codes

Hamming binary codes are widely used in technique. It is the reason to research switch's structure in which, for instance (7,4) Hamming code, the generated polynom of which is $g(X) = 1 + X + X^3$, is used. Then generated matrix G has the following form:

$$\begin{matrix} g(X) \\ Xg(X) \\ X^2g(X) \\ X^3g(X) \end{matrix} \begin{pmatrix} 1101000 \\ 0110100 \\ 0011010 \\ 0001101 \end{pmatrix} = G.$$

The remaining code vectors are obtained through summation of G-matrix strings in any possible combinations. Let us now enumerate all code vectors in order to information symbols were ascendly ordered by binary numbers [10]. Coding table which in our case is used for obtaining addresses of delivers has the view shown in table 1.

Table 1

N	Code vector	
	Check symbols	Information symbols
1	110	1000
2	011	0100
3	101	1100
4	111	0010
5	001	1010
6	100	0110
7	010	1110
8	101	0001

N	Code vector	
	Check symbols	Informatiion symbols
9	011	1001
10	110	0101
11	000	1101
12	010	0011
13	100	1011
14	001	0111
15	111	1111

The parity check matrix which is written through the vector $h(X) = 1 + X + X^2 + X^4$ has the form

$$H = \begin{pmatrix} 0010111 \\ 0101110 \\ 1011100 \end{pmatrix}$$

The block-diagram of one-bit 15×15 -switch is shown in fig.2.

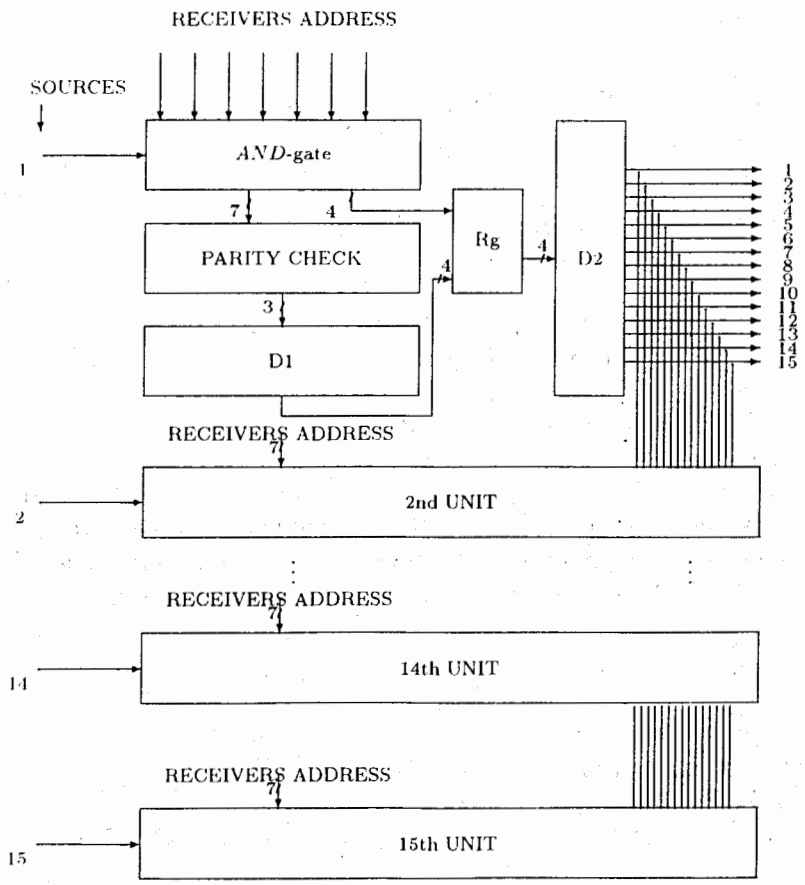

Figure 2: Block-diagram of one-bit 15×15-switch with correcting capabilities. D1,D2- decoders; Rg - register.

The switch has 15 one-type logic layouts the first of which is shown more refining. The signal from the first source is passed to the strobing inputs of elements AND group, the codes of delivers addresses are passed to the second inputs of this group. The outputs of the AND elements are joined to the inputs of calculating syndrom layout. Besides information bits are passed to the inputs of decoder. Outputs of decoder 1 are joined to the set inputs of register but outputs of decoder 2 are joined to counting inputs of this register. Let us shortly see the principe of the switch's work. Let us suppose that we need in transformation simultaneously signals from the 1st source to the 15th receiver, from the 2nd source to the 14th receiver, from the 14th one to the 1st one and from the 15th one to the 2nd one. The signals form remaining sources equal zero. As the result we have the table of communications:

Source number	Receiver number	Receiver address
1	15	11111111
2	14	00100111
⋮	⋮	⋮
14	1	11010000
15	2	01101000

Let us see the signal path from 1st source to 15th receiver on conditions that the error appeared in the 7th bit during the transmission of the address. In other words the code address 1111110 will be on the outputs of the group of the elements AND instead of the 111111. Then the code 1000 which suits to 7th column of the matrix H will be obtained on the outputs of the layout syndrom calculation. The correction of the register value is fulfilled by means of the decoder D1. As a result, the data from source 1 is transformed through the decoder D2 without error to the receiver 15. In the same way we can check if the other channel of the switch works correctly. Thus switch works without errors even if up to one error in each address (15 errors in total) appeares simultaneously. N such modules are needed for the construction of N-bit switch. The codes of the addresses are common for the appropriate bits in this case. We can use the modified Hamming code instead of the usual code of Hamming however in this way we need in complement logic elements for implementation of a switch. We can use the binary Hamming code as it

is frequently done in practice. Such approach does not change the structure of a switch. Let us estimate the switch speed T_C:

$$T_C = T_{AND} + 2 \cdot T_D + T_P + T_{Rg},$$

where T_{AND} - delay of AND-gate; T_D - delay of a decoder; T_P - delay of a parity checker and T_{Rg} - the time of a register switching. Rough estimation shows that $T_C = 10 \times T_{AND}$. If the scheme of the full decoding of coding words which have the error are used instead of the classic scheme, the speed of the switch will essentially increase due to increasing of the number of logic elements. Each channel has only one group of elements AND and 7 decoders. Besides, a signal at the first output, for instance, must appear not only when there appears the code 1111111 on its inputs, but the codes with mistakes: 0111111, 1011111 and, etc. In this case $T_c \doteq 3 \cdot T_{AND}$ and it is very small.

3.2 Use of Bose-Chaudhuri-Hocquenghem (BCH)-codes

BCH-codes are one of the most important classes of random-error-correcting codes which are known. The implementation of such codes permits one to construct switches with wide functional possibilities. Together with correcting capabilities the fast detection of appearance of a catastrophic number of errors during transformation and switching data is possible. In general the logic layout of such switch is the same as shown earlier, but the majority coincidence scheme is added for detecting a large number of errors. So far as the problem is to construct fast switches we suggest to use well-known table of methods of BCH-code error decoding [11,12].

We consider the use of such codes for construction of the switches with error correcting capabilities on the example of BCH-codes with $m = 4$ and $t = 3$, where t - set of independent errors within the block of n bits. Consider the finite field GF(2^4). The generator polynomial for $t = 3$ is $g(X) = (1 + X + X^4)(1 + X + X^2 + X^3 + X^4)(1 + X + X^2)$. The parity check matrix of the (15,10)-code may be [9]

$$H^T = \begin{pmatrix} 1000 & 1000 & 1000 \\ 0100 & 0001 & 0110 \\ 0010 & 0011 & 1110 \\ 0001 & 0101 & 1000 \\ 1100 & 1111 & 0110 \\ 0110 & 1000 & 1110 \\ 0011 & 0001 & 1000 \\ 1101 & 0011 & 0110 \\ 1010 & 0101 & 1110 \\ 0101 & 1111 & 1000 \\ 1110 & 1000 & 0110 \\ 0111 & 0001 & 1110 \\ 1111 & 0011 & 1000 \\ 1011 & 0101 & 0110 \\ 1001 & 1111 & 1110 \end{pmatrix}$$

With $m = 4$, t $= 3$ $m \times t = 4 \times 2 + 2 = 10$ check bits are required. We also suggest to use the well known W.Peterson theorem. For any BCH(2.1.d) code and any j such that $1 \leq j \leq n - 1$, the $j \times j$ matrix

$$L_t = \begin{pmatrix} S_1 & 1 & 0 & 0 & 0 & \cdots & 0 \\ S_3 & S_1^2 & S_1 & 1 & 0 & \cdots & 0 \\ S_5 & S_1^4 & S_3 & S_1^2 & S_1 \cdots & 0 & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ S_{2t-1} & S_{2t-2} & S_{2t-3} & S_{2t-4} & S_{2t-5} & \ddots & S_t \end{pmatrix}$$

is nondegenerate if the power symmetric functions S_j depend on t or $t + 1$ field elements, it is singular if the S_j depend on fewer than $(t - 1)$ different field elements. In other words, to calculate t it is necessary to calculate the determinant L_t in Galois field. The expressions for the determinants for $t = 1 \div 3$ are given in table 2.

Table 2

t	$detL_t$
1	S_1
2	$S_1^3 + S_3$
3	$S_1^6 + S_1^3 S_3 + S_1 S_5 + S_3^2$

We see that det L_t can be zero or any field value. Therefore, the logic expressions for majority coincidence scheme (MCS) containing n inputs and t outputs have the form [6]

$$\begin{aligned}
output_1 &= detL_1 \vee detL_2 \vee detL_3 \ldots detL_j \vee \ldots detL_t \geq 1 \\
output_2 &= detL_2 \vee detL_3 \ldots detL_j \vee \ldots detL_t \geq 2 \\
output_3 &= detL_3 \ldots detL_j \vee \ldots detL_t \geq 3 \\
\vdots \quad &\quad \vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots \\
output_j &= detL_j \vee \ldots detL_t \geq j \\
\vdots \quad &\quad \vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots \\
output_t &= detL_t \geq t
\end{aligned} \tag{1}$$

If to the MCS described by Eqs. (1) we add simple schemes for analyzing determinants for 0s and 1s, we can also obtain rigorous equalities, as shown in fig.3. It should be noted that such devices can be realized in practice by using fast tabular methods. The circuit shown in fig.3 was designated so that it can be realized for large numbers t by using PROMs containing $2m$ address inputs. Note that the length of a word and complexity of the error correcting layout essentially increase with increasing of t. It is the reason that we limited ourselves to the case when $t = 2$. The simple and fast methods for searching for places of errors are well known for this case [11,12].

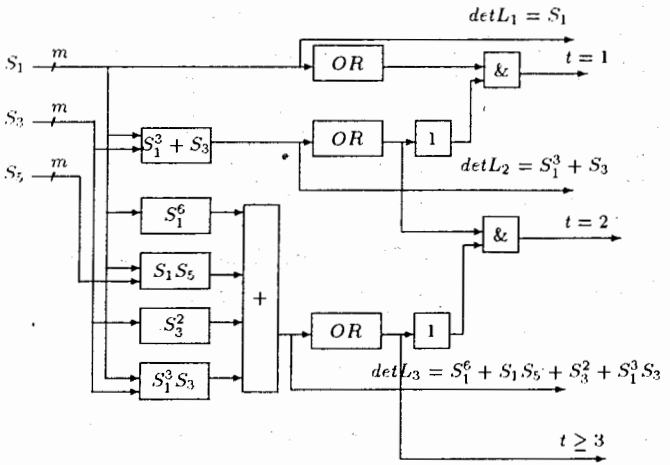

Figure 3: Circuit for calculating determinants in the field GF(2^m) for $t = 1 \div 3$.

Example. We assume the code word (addresses of a receiver) in question to be A = 100000111100000. With two errors occurring at the fifth and the tenth positions from the left, the received sequence takes the form 100010111000000. Computing S_j for $j = 1 \div 3$ it can be obtained $S_1 = a^4 + a^9 = a^{14}$; $S_3 = a^{12} + a^{27} = 0$; $S_5 = a^5 + a^0 = a^{10}$. $detL_1 = a^{14}$; $detL_2 = a^{12}$; $detL_3 = 0$. The general form of the quadratic error is [12] $X^2 + S_1 X + (S_1^3 + S_3)/S_1$. Transformation form $X + S_1 \times Y$ yields $Y^2 + Y + d = 0$, where $d = (S_1^3 + S_3)/S_1$. The roots are find by only d_i. If roots corresponding to all possible d_i's are stored in memory as a table, they can be directly obtained. The root X_1 is $X_1 = S_1 \times Y$. Only Y_1 can be stored, however, since $X_2 = S_1 + X_1$. For our example $d = a^0$ and $Y_1 = a^5$. Then $X_1 = S_1 Y_1 = a^4$.

Thus the scheme of two errors correction and detection of the large number of errors work in accordance with the follows algorithm.

$detL_1 = 0$ - error are absent.

$detL_1 \neq 0$, but $detL_2 = 0$. We have one error and S_1 is the coordinate of it.

$detL_2 \neq 0$, but $detL_3 = 0$. We have two errors and their coordinates can be fast calculated through the table.

$detL_3 \neq 0$. The solving is $t \geq 3$.

4 Conclusion

The main difference of the switches described in this paper is that not information but addresses of an active source are switched. As a result, these switches have a simple modulo structure. The absence of a bus which requires high power current switches, the correcting possibility, small delays and advanced technology make it possible to construct effective switches with many inputs for multiprocessor systems and high-speed networks and communication with wide functional possibilities.

References

1. Mount R.P. Present and future computer architectures. In: 1987 CERN school of computing. CERN 88-03. P. 151-182. Geneva, 1988.

2. Nomachi M., Sasaki O., Fujjii H., Ohska T.K. A large scale switch type event builder. In: Computing in High Energy Physics '92. CERN 92-07. P. 188-191. Geneva, 1992.

3. Veselovskiy G.G., Karavai M.F., Kuznechik S.M. Avtomatika i Telemehanika. 1989. No.2. P.3-20.

4. *SN74ACT8841* digital crossbar switch. Texas Instruments Data Book.

5. Nikityuk N.M. The method of syndrome coding and its application for data compression and processing in high-energy physics. In: 6th Int. Conf. Applied Algebra, Algebraic Algorithms and Error Correcting Codes (AAEC-6). Rome, Italy, July 4-8, 1988. Ed. T. Mora. (Lecture Notes in Computer Science. Ed. by G.Goos and Hartmanis). Springer-Verlag. P. 324-335.

6. Nikityuk N.M. The syndrome coding technique and its application to fast event in high-energy physics experiments. Physics of Particles and Nuclei. 1993. Vol. 24. No.1. P. 77-106. (Formerly Sovet Journal of Particles & Nuclei).

7. Pradham D.K. Theory of Galois switching functions and their application. IEEE Trans. on Comput. 1978. V. C-27. No. 3. P. 239-248.

8. Benjauthrit B., Reed I. Galois switching functions and their applications. IEEE Trans. on Computers. 1976. V. C-25. No. 1. P. 78-86.

9. Peterson W. W. Error-correcting codes. 1964. M.I.T. Press. New York - London 1961.

10. Harkevitch A.A. Borba s pomechami. Nayka, Moscow 1965. P.225.

11. Okano H., Imai H. Construction method of High-speed decoders using ROM's for BCH and RS-codes. IEEE Trans. on Comput. 1987. V. C-36. No. 10. P. 1165-1170.

12. Polkinghorn F. Decoding of double and triple error correcting Bose-Chaudhuri codes. IEEE Transactino on Information theory. 1966. Vol. IT-12. No.4. P. 480-481.

Received by Pyblishing Department
on December 28, 1994.

Александров И.Н. и др. E10-94-514

Коммутация информации в полях Галуа $GF(2^m)$

Предложен новый тип многоустойчивых полных коммутаторов с алгебраической структурой. Эти коммутаторы имеют маленькие временные задержки и могут переключать одновременно поток данных с n входов на m выходов. Представлены также коммутаторы, корректирующие ошибки. Коды Хэмминга и БЧХ-коды использовались для построения указанных коммутаторов. Предложена конкретная схема многоустойчивого полного коммутатора с кодом Хэмминга.

Работа выполнена в Лаборатории высоких энергий ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна, 1994

Alexandrov I.N. et al. E10-94-514

Commutation Information in Galois Fields $GF(2^m)$

The new type of multipoint full crossbar switches with algebraic structure is suggested. Such switches have small delays and possibility of switching simultaneously any n inputs and m outputs. The swiches with error-correcting possibilities are also supported. Hamming and BCH error-correcting codes were used for construction of such switches. The concrete sheme of 15×15 multipoint full crossbar switch with Hamming code is given.

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

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