UDC 681.3.012 (045)

I.A. Zhukov, Dr. Sci. (Eng.)
V.A. Gumenyuk, Cand. Sci. (Eng.)

APPLICATION OF INDIVISIBLE «*M* OUT OF *N*» CODES IN HIGHLY PRODUCTIVE PARALLEL COMPUTING SYSTEMS

NAU Institute of Computer Technologies, e-mail: iit@nau.edu.ua

The motives for beginning the automatic checkup problem in computers are considered. The "M out of N" codes applications possibilities, related to indivisible ones, are analyzed in highly productive parallel computing systems. A new scheme of computing operations organization is proposed with taking into account the peculiarities of such coding.

Introduction

The mass parallel computing processes realization problem. This problem appeared due to the requirements increasing for the computing means productivity – these requirements exceeded the physical possibilities of a single processor using Von Neumann principle [1; 2].

The requirements increasing for the computing means productivity in the period since 1995 up to this day is given in fig. 1.

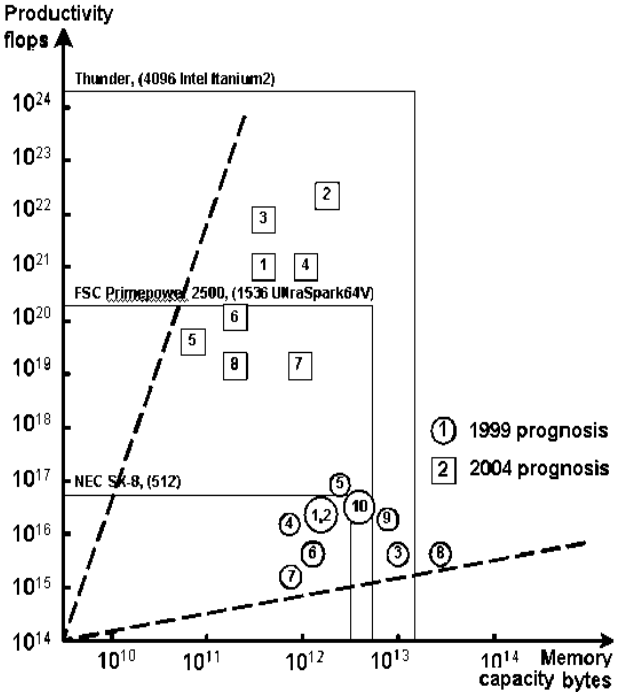

Fig. 1. The requirements increasing for the computing means productivity in the period since 1999 up to this day is given:
1 – designing airplains and rockets; *2* – creating medicines for hard illnesse of the mankind; *3* – prognostics the weather and nature cataclyasms; *4* – increasing the power stations work; *5* – increasing the reliability of cars and simulating their clashes; *6* – 3-D magnetoplasmadynamics; *7* – fundamental science investigations; *8* – economic analysis of a human activity; *9* – ecology; *10* – application for war purposes

This problem realization continues to be actual one due to the constant increase the requirements to the computing means productivity and the fundamental and applied problems decisions reliability, united by the notion «Grand challenges». Nowadays there are two basic structures of highly productive computing systems: multiple computer complexes on the base of microprocessors with fixed memory distribution upon processors (distributed memory) and multi-PUs with allocated memory in the process of computing (allocated memory).

The object of the work is evaluation the possibilities of the parallel computing systems real productivity increasing and the reliability level of their computing results on the basis of indivisible «M out of N» codes as well.

The target setting

The analysis of the real productivity decrease of the highly productive computing systems with the increasing the processors number *N* [1; 2] allows to determine two causes for such regularity:

– spatial, concerned with the processors underloading, as a result of data delay while transmitting them from the memory to the processor or from one processor to another one;

– temporal, concerned with the processors underloading, as a result of data absence, supplied by the other computing processors (data synchronization).

With increasing *N* up over 32 difficulties appear with multi-PUs constructing due to the double commutator delay (processor-memory and memory-processor transferring) inclusion into the execution time of each processor operation.

The problem with data synchronization in large tasks appears in highly productive computing systems due to necessity to develop a very complex on-line program, in which every parallel process is fulfilled in its devoted processor, RAM volume, data transfer channel, etc. The attempt to use memory locations for data synchronization may lead to a false result of the task decision, since it is very likely to use outdated data instead of updated ones.

The next conclusions concerning the directions of the problem decision are made in [1; 2]. For decision the spatial task of the data supplying it is necessary, particularly, work for:

– increasing productivity of each microprocessor;

– excluding the cache coherent problem out of the complex (Cray machines have no cache);

– excluding the time delays of the information transferring between the executing device and RAM out of the operation execution time;

– increasing throughput between the primary and secondary memories.

For decision the temporal data correlation it is necessary the following:

1) the data synchronization and computing processes paralleling problems should be decided by hardware means during the task fulfilling process;

2) conceptually exclude RAM;

3) exclude the processor time loss for interruptions and processes synchronization.

The systems with small number components, in which it is not difficult to transmit the full-size data by parallel method, are not considered here.

The significant time delays in buses and commutators, connecting computers in a system (e.g., fig. 2 illustrates a ring architecture with full connection via chords – Chordal Ring), leads to the necessity to transfer data (operands) via the system bus (commutators) in parts. After accumulation the whole bytes in the input registers their authenticity is tested by a checkup circuitry. The possibility of an alternative nonbinary digital information presentation in the system is analyzed in the paper.

Methods of the decision

There are interconnection between the real computing system throughput and its checking circuit causing timing overhead.

The computing results reliability in microprocessors is usually ensured by parity checking and using ECC codes. The digital information checking methods in some up-to-date most highly productive microprocessors are given in table.

As it is clear from the table, in the most cases the code with parity checking is applied which, as it is known, is generated by addition one redundant (checking) bit to the group of information digits presenting a simple (irredundant) code.

If while transferring the receiving device discovers that the checking bit value in the received word does not correspond to the word sum parity, then this is interpreted as an error indication.

The minimal code distance $d_{\min}=1$. Simultaneously appearing two or any other even number of errors is not discovered by the code with parity checking.

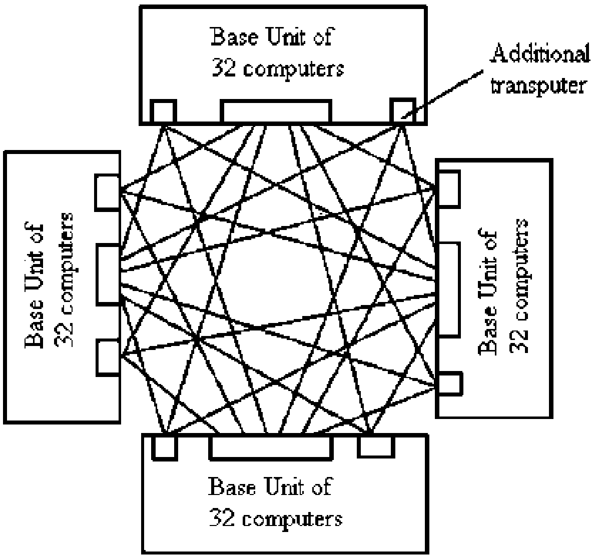

Fig. 2. The ring architecture with full connections via chords (Chordal Ring)

Checking methods in highly productive microprocessors

Units of computer	Most highly productive microprocessors				
	SPARC64 V1	UltraSparc III2	Itanium 2	Power 4	PA-8700
ALU	Parity + repeating	No	No	No	No
Registers	Parity + repeating	No	No	No	No
TLB	Parity + excluding	Parity	Parity	Parity	Parity
I-cache of the first level	Parity + excluding	Parity	Parity	Parity	Parity
I-cache tags of the first level	Doubling + excluding	Parity	Parity	Parity	Parity
D-cache of the first level	ECC + excluding	Parity	Parity	Parity	ECC
D-cache tags of the first level	Doubling + excluding	Parity	Parity	Parity	Doubling

Notes. Repeating is the operation automatic repeating. Eexcluding is the defected microprocessor switching off without reloading. Parity is the parity checkin.

Logic elements with paraphase outputs, similar to those shown in fig. 3, *a*, *b*, are used for constructing parity checking circuits. The shown circuits realizes the module 2 adding operation (reference designation is M2) for the binary variables *x* and *y*. A byte parity checking circuit is shown in fig. 3, *c*.

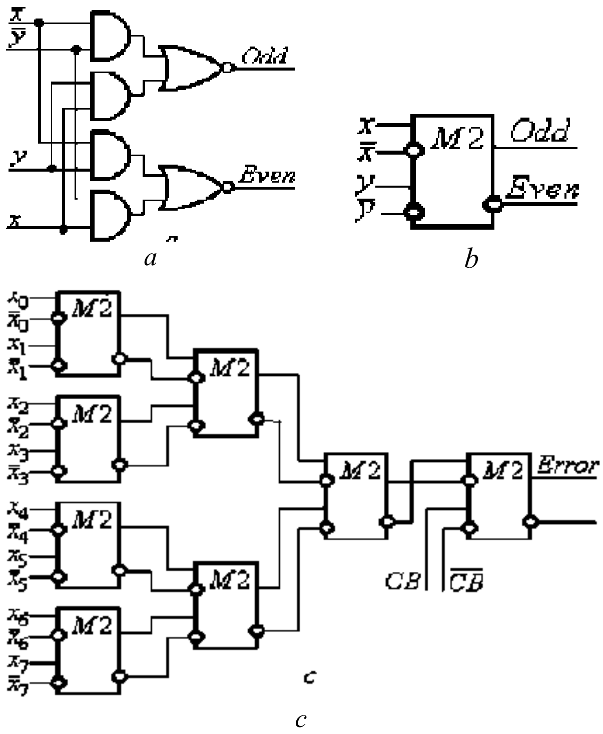

Fig. 3. Parity checking circuits:
a, b – logic elements with paraphrase outputs, similar to those shown; *c* – a byte parity checking circuit is shown

The structure of the parity checking circuit is multistage, i.e. the word is divided into some bit groups, each of them is parity checked by the direct method (the first stage), then the second group is formed from the first stage groups parities which are considered as usual binary bits and parity checked, and so on up to the final parity checking of the whole word. In the last stage the byte parity is compared with the value of the Control Bit CB.

The principal drawbacks of such checking method are insufficient level of the probability to discover the most error categories and a large time delay provided by the parity checking circuit (fig. 3).

It is established in [3] that the traditional method of the digital information presentation in computers, in which both digits of a binary position are presented by one flip-flop (interdependent presentation), has led to loss of the positional numeration natural check-up ability that, in its turn, stipulated for using different redundant code constructions in computers permissive to discover or correct errors. So, in computer each itch unit the redundant information is constructed by a definite law C_i with respect to the main information. While processing the main information depending on its view, its overpatching A_i by definite laws and rules K_i, mismatching with law A_i, the main information changes (except the main information and checking automata doubling case) the checking information is altered as well. At that, the transformation is being done in such a way that the codes, obtained as a result of performing both autonomous operations (A_i at the main and K_i at the check information), should mutually correspond to initial law C_i.

Thus, the principal logic law of the computer unit hardware checking organization while information traditional presentation is defined [4] as follows:

$$\left.\begin{aligned} &I_{\text{o}i} \equiv (C_i)\, I_{\text{КО}i};\\ &I_{\text{o}i}\,/A_i \equiv (C_i)\, I_{\text{КО}i}\,/\,K_i\,, \end{aligned}\right\}$$

where $I_{\text{o}i}$ and $I_{\text{КО}i}$ are the main and checking information in ith unit respectively; $I_{\text{o}i}$ / A_i is the main information after fulfilling operation A_i with it; $I_{\text{КО}i}$ / K_i is the checking information after transforming it by law K_i; symbol ≡ (C_i) means that the information is comparable by law C_i.

While using such hardware checking construction principle it is impossible to provide faultless functioning checking apparatuses due to the next causes [3]:

– checking system is constructed using elements with fault passport intensities identical with checked unit elements;

– the work power modes of checking system elements don't differ essentially from the work modes of checked elements, since the modulo checking procedure is based on the computing operation fulfillment using residue method simultaneously with fulfillment the checked operations;

– the checking elements reservation leads to the computing system some parameters worsening.

The probability D_l for getting correct (trustworthy) result in the computing system is defined in [3] as

$$D_l = 1 - (1 - P_{kl} * P_{nl} * P_{ml}) * R_l, \qquad (1)$$

where P_{kl} is the probability of the checking apparatuses faultless work; P_{nl} is the probability of the computing system hardware checking coverage; P_{ml} is the methodic probability of the errors discovering of category l; R_l is the probability of the errors beginnings ошибок of category l (l = 1, 2, …, Ψ).

From expression (1) it follows, that the problem cannot be decided on the basis of the digital information traditional presentation: with interdependent presentation 0 and 1 in one bit position by one flip-flop, since conditions P_{kl} = 1, P_{nl} = 1, P_{ml} = 1 are not provided. While these conditions are provided, D_l = 1 independently of R_l values, or R_l = 0 (that, naturally, is practically impossible).

Since with modulo value increasing (while modulo checking), probability P_{ml} increases insignificantly and this is practically completely compensated by inevitable at cally completely compensated by inevitable that probability P_{kl} decreasing.

As an alternative to the traditional binary information presentation in computing systems may be used the digital information presentation by codes «*M* out of *N*», in which *N* is the number of positions (each of which is presented by a separate flip-flop) in the number position; *M* is the number of «units» in these positions (the rest $N - M$ positions contain «zeros»). Thus, the ratio «zeros» and «units» is fixed in the frame of the given code. Such codes are indivisible since the code word number positions cannot be divided into information and redundant ones.

Changing the correspondence between single-digit code combinations and digits of chosen numeration base *p* lead to formation another alphabets.

As it follows from works [3; 5], with using code «*M* out of *N*», the problem of automatic checking in computing systems may be practically decided. Let us shortly concentrate on some important peculiarities of such codes. Maximal number of combinations in one number position (it is defined the range of the numeration base value choosing) is calculated by the formula

$$C_N^M = N! \,/\, [M! * (N - M)!],$$

where $N! = N * (N - 1) * (N - 2) * \ldots * 1$.

The numeration base value *p* is chosen from the condition

$$C_N^M \geq p.$$

Values of *p*, chosen from sets $p = 2^k$ (where $k = 2, 3, 4 \ldots$) and $p = 10^m$ (where $m = 1, 2, 3 \ldots$) have the most practical interest, taking into account the necessity of adapting with present computers.

The most important characteristic, defining the code ability to discover errors, is the minimal code distance:

$$d = \min d(X, Y), \quad \text{when } X \neq Y.$$

For code «*M* out of *N*», depending of the parameter *N* parity, there are $d(X, Y)$ value next changing ranges:

– for even *N*

$2 \leq d(X, Y) \leq 2M$, when $1 \leq M \leq N / 2$;

$2 \leq d(X, Y) \leq 2(N - M)$, when $N / 2 + 1 \leq M \leq N - 1$;

– for odd *N*

$2 \leq d(X, Y) \leq 2M$, when $1 \leq M \leq (N - 1) / 2$;

$2 \leq d(X, Y) \leq 2(N - M)$, when $(N - 1) / 2 \leq M \leq N - 1$.

Thus, for code «*M* out of *N*», $d \geq 2$ in any case.

For discovering any error with response ratio *l* or less it is necessary and sufficient that the minimal code distance would be more than *l* at least by 1 [5]:

$$d \geq l + 1.$$

Consequently, code «*M* out of *N*» allows discovering any single error.

The complete set of error categories, possible in such code, is defined as follows:

– the code combination contains $M - i$ «units», where $i = 1, 2, \ldots, M$;

– the code combination contains $M + i$ «units», where $i = 1, 2, \ldots, N - M$;

– the code combination contains exactly *M* «units», out of which *i* ones are false, where $i = 2, 4, \ldots, M$, if *M* is even, or $i = 2, 4, \ldots, M - 1$, if *M* is odd.

Discovering errors of the first two categories (with code structure violation) is provided with probability $P_{ml} = 1$, and the third one with probability $P_{ml} = 0$. Consequently, reliability D_l of the digital information is completely defined by probability R_3 of the third category error formation [3]:

$$D_l = 1 - R_3.$$

Let us define the probability R_3 changing dependence of *M* changing when values *p* are fixed.

Formation of undiscovered error R_3 is possible while joint development of errors of the first two categories during time

$$t = a * \Delta t_k,$$

where Δt_k is the resolvability in time of the automatic checking; $0 < a < 1$ [5].

Increasing *M* in the range $1 \leq M \leq N / 2$, with fixed *p*, allows to decrease *N* (and accordingly reduce hardware expenditures), that leads to changing probability R_3 constituents, i.e., to changing probability of the first two categories.

Probability $R_{3,}$ when $1 < M < L$, is defined by expression

$$R_3 = \sum_{i=1}^{L} P_{\mathrm{f}\,i} * P_{\mathrm{r}\,i},$$

where $L = M$, when $N / M \leq 2$; $L = N - M$, when $N / M > 2$; $P_{\mathrm{f}\,i}$ and $P_{\mathrm{r}\,i}$ are probabilities of fallout and rise *i* symbols (e.g., «units») in each combination accordingly.

Probabilities $P_{\mathrm{f}\,i}$ and $P_{\mathrm{r}\,i}$ are defined by according expressions

$$P_{\mathrm{f}\,i} = \sum_{i=1}^{L} C_M^i * R_0^i * (1 - R_0)^{M - i};$$

$$P_{\mathrm{r}\,i} = \sum_{i=1}^{L} C_{N-M}^i * P_0^i \,(1 - P_0)^{N - M - 1},$$

where R_0 is the probability of one symbol fallout; P_0 is the probability of one extra symbol rise.

Therefore,

$$R_3 = \sum_{i=1}^{L} C_M^i * R_0^i * (1 - R_0)^{M - I} * C_{N-M}^i *$$

$$*P_0^i \,(1 - P_0)^{N - M - 1}.$$

For simplifying analysis we will consider in further that $R_0 = P_0$, then

$$R_3 = \sum_{i=1}^{L} C_M^i * C_{N-M}^i * P_0^{2i} * (1 - P_0)^{N-2i}.$$

Probability P_0 is defined be expression

$$P_0 = e^{-\lambda t Q}e = e^{-\lambda K \Delta t k Q}e,$$

where Q_e is the quantity of logical elements, which are necessary for presentation of one digit in the computer.

For the contemporary element base $\lambda \approx 10^7$ 1/s, and the time delay of pulses while propagation through IC not more than $0{,}3 * 10^{-6}$ s [3]. Therefore, when

$$K = \sum_{i=1}^{L} C_M^i * C_{N-M}^i;$$

$$R_3 \approx \sum_{i=1}^{L} C_M^i * C_{N-M}^i * (0.3\, a * Q_e)^{2i}\, 10^{-26i}.$$

From the analysis of the probability R_3 approximated values for M and N values set, it follows that such probability, defined for codes having code combinations equal quantity, decreases while value M increasing from 1 to $N - 1$, if this increasing is accompanied by decreasing N, and remaining constant, at the same conditions, if N is not changed. Due to highly unlikely of two and more false transitions 0 → 1 and the same quantity of false transitions 1 → 0 in the same code combination during time $t = a * \Delta t_k$, probability R_3, in practice, may be defined by approximated expression

$$R_3 \approx 0{,}9\mathrm{M}(N - M)\, a^2 Q_e^2 * 10^{-27}.$$

Thus, p-valent digit presentation by code «M out of N» allows practically solving the problem of computing system automatic checking. The checking and information symbols in code combinations are not marked out, the redundant information is distributed uniformly between all the positions of these combinations, does not depend on operation form, presenting possibility for checking coverage all the set of full-size numbers. Therefore it is possible to consider condition $P_{n1} = P_{n2} = 1$ (where P_{n1} and P_{n2} are probabilities of checking coverage the computing system equipment, with the expectation of discovering the first two category errors, accordingly) principally achievable.

Let us consider the possibility of fulfilling the condition $P_{k1} = \mathrm{P}_{k2} = 1$, where P_{k1} and P_{k2} are the probabilities of checking apparatuses faultless working, designed for discovering the first two categories errors accordingly.

In minimal Disjunctive Normal Form (DNF) expressions, corresponding to signs A_1 and A_2 of the number error record (e.g., operand X) when $M > 1$, define algorithms of the checking apparatuses work:

– for the first category errors

$$A_{1X} = \bigcap_{(i)} x_{0,i} \cup \bigcap_{(i)} x_{1,i} \cup \ldots \cup \bigcap_{(i)} x_{\mathrm{p}*-1,i} = \\ = \bigcup_{n=0}^{p*-1} \bigcap_{(i)} x_{\mathrm{n},i}, \qquad (2)$$

where $x_{\mathrm{n},i}$ is the signal, corresponding to «unit» in i-*th* code combination position, representing digit n in code «M + 1 out of N»,

$$\sum_{(i)} x_{\mathrm{n},i} = M + 1; \quad 0 \le n \le p* - 1; \quad p* \ne p;$$

– for the second category errors

$$A_{2X} = \bigcap_{(i)} \vec{x}_{0,i} \cup \bigcap_{(i)} \bar{x}_{1,i} \cup \ldots \cup; \\ \cup \bigcap_{(i)} \bar{x}_{\mathrm{p}**-1,i} = \bigcup_{n=0}^{p**-1} \bigcap_{(i)} \bar{x}_{\mathrm{n},i}, \qquad (3)$$

where $\bar{x}_{\mathrm{n},i}$ is the signal, corresponding to «zero» in i^{th} code combination position, representing digit n in code «M – 1 out of N»,

$$\sum_{(i)} \bar{x}_{\mathrm{n},i} = M + 1; \quad 0 \le n \le p** - 1;$$

$p** \ne p,$ when $M = N/2$, $p* = p**$.

The distinguishing peculiarity of checking devices functioning, corresponding to expressions (2) and (3), is including their elements only while errors appearing. The checking element faultless work probability $P_{\mathrm{e},l}$ is defined as

$$P_{\mathrm{e},l} = P_{e,l}^{*} (1 - R_{l,1}) + P_{e,l}^{**} R_{l,1},$$

where $P_{e,l}^{*}$ and $P_{e,l}^{**}$ are probabilities of checking elements faultless work, at absence and presence error of ratio l in the checked number position; $R_{l,1}$ is the probability of a ratio l error appearing in the given number position.

The high reliability of such devices is provided by the reliability level of the contemporary element base and natural reservation of conjunctive elements in the function of the false digital signals quantity (at failures of the part of the common circuit for error checking in code combinations, presenting a full-size number, they may be discovered in normally functioned number positions [3]).

Thus, it drops off in the necessity of obligatory accumulating n-positioned operands in the input registers of the operation unit and subsequent their checking (e.g., while parity checking, the convolution and comparing operations are fulfilled sequentially), which leads to unproductive time delays. It is appeared the possibility for organization sequential (conveyor) transferring of the both operands separate number positions (while arithmetic operations) or units of k number positions, directly to the executive device with the appropriate number positions (single-digit or k-digit) with the subsequent fixing the partly result in appropriate positions of the full-size result register. Note, that while using, e.g., code «3 out of 6» the alphabet may contain up to 20 different symbols, from which it is advisably to choose 16 (2^4). Therefore, at the fixed number presentation precision in the system, the capacity of all registers is shortened by 4 times. The circuit, realizing the declared principle, is presented in fig. 4. and include a memory (e.g., associative memory), a set of matrix executive devices without input registers (k-position logic, where $k = 1, 2, \ldots; \kappa < n$.

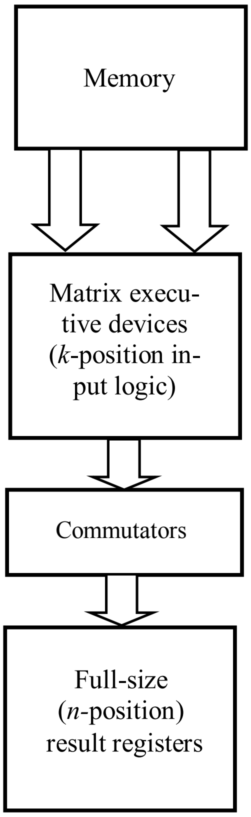
Memory
Matrix executive devices (k-position input logic)
Commutators
Full-size (n-position) result registers

Fig. 4. The simplified circuit for transferring and processing n-position operands by units with k number positions

In consequence of $P_{e,l}$ large value an element fault may be considered as a practically impossible event, therefore the natural redundancy provides additional possibilities for even more increasing checking apparatuses detectivity, since their elements do not form a successive circuit in a probabilistic sense. The action effect of such checking apparatuses in all number positions provided to be even more, since it is enough to discover an error in a single number position to recognize the checking number as erroneous one [3]. A comparison reliability value, defined for codes with different M, allows making conclusion that all the representatives of this code family provides a possibility to obtain practically reliable results of computing.

Conclusion

Code «M out of N» and appropriate computing devices structures properties analysis shows that checking function fulfilling in them is not connected with additional time delays and does not decrease the computing system common productivity. Therefore it is advisably to use codes «M out of N» in highly productive computing systems for increasing their processors productivity and also for increasing result reliability while deciding critical problems. Besides, at the fixed computing system productivity, increasing separate processors productivity allows to shorten their quantity.

Literature

1. *Бурцев В.С.* Новые подходы к оценке качества вычислительных средств// Параллелизм вычислительных процессов и развитие архитектуры суперЭВМ. – М.: Нефть и газ. 1997. – С. 28–40.

2. *Бурцев В.С.* Система массового параллелизма с автоматическим распределением аппаратных средств суперЭВМ в процессе решения задачи // Юбил. сб. тр. ин-тов Отд-ния информатики, вычислительной техники и автоматизации РАН. – М., 1993. – Т. II. – С. 5–27.

3. *Брюхович Е.И.* О проблеме автоматического контроля в ЭВМ и контролеспособности позиционных счислений // УСиМ. – 1977. – №4. – С. 71–75.

4. *Дадаев Ю.Г.* Теория арифметических кодов. – М.: Радио и связь, 1981. – 274 с.

5. *Брюхович Е.И., Гуменюк В.А.* Исследование достоверности результатов вычислений при представлении чисел в ЭВМ «M из N» кодом // Техническая диагностика электронных систем: Сб. науч. тр. – К.: Наук. думка. – 1982. –172 с.

The editors received the article on 22 Febuary 2005.

І.А. Жуков, В.О. Гуменюк

Застосування неподільних кодів “M із N” у високопродуктивних паралельних обчислювальних системах

Розглянуто проблему автоматичного контролю в комп’ютерах. Запропоновано нову схему організації обчислювальних операцій з урахуванням особливостей такого кодування.

И.А. Жуков, В.А. Гуменюк

Применение неразделимых кодов “M из N” в высокопроизводительных параллельных вычислительных системах

Рассмотрена проблема автоматического контроля в компьютерах. Предложена новая схема организации вычислительных операций с учетом особенностей такого кодирования.