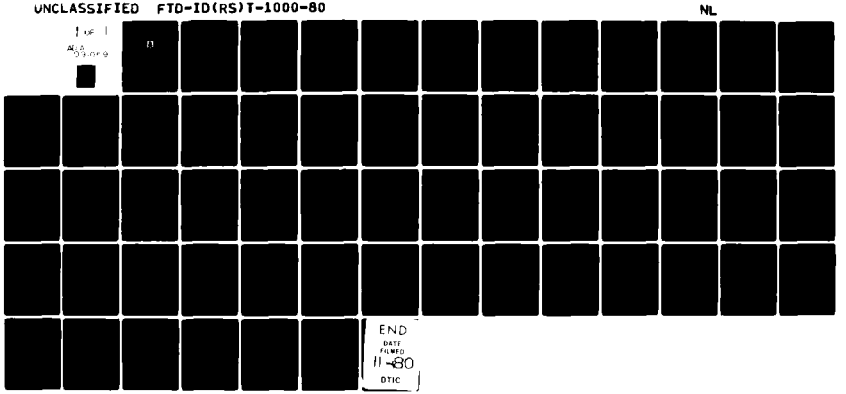
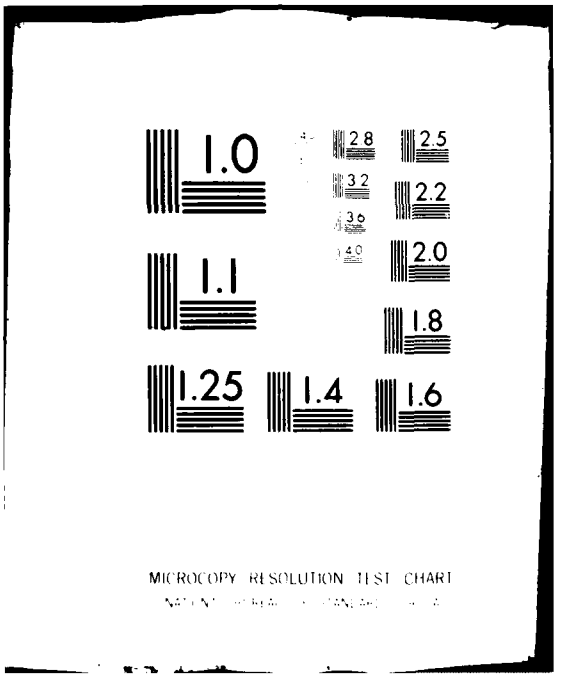
AD-A091 069 FOREIGN TECHNOLOGY DIV WRIGHT-PATTERSON AFB OH F/G 12/1
CORRECTIVE ARITHMETIC CODES IN THE RESIDUAL CLASS SYSTEM,(U)
JUL 80 S B FAYN
UNCLASSIFIED FTD-ID(RS)T-1000-80 NL



END
DATE FILMED
11-80
DTIC


1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6
MICROCOPY RESOLUTION TEST CHART

FTD-ID(RS)T-1000-80

(4)

FOREIGN TECHNOLOGY DIVISION

CORRECTIVE ARITHMETIC CODES IN THE RESIDUAL CLASS SYSTEM

by

S. B. Fayn

EDITED TRANSLATION

(14) FTD-ID(RS)T-1000-80

(11) 30 July 1980

MICROFICHE NR: FTD-80-C-001032

(12) 60

(6) CORRECTIVE ARITHMETIC CODES IN THE RESIDUAL CLASS SYSTEM

By: S. B. Fayn

(10)

English pages: 57

(21) Edited trans. of Source: Voprosy Vychislitel'noy Tekhniki (USSR) Vol VII, Issue 2 p60-91, 1966, by 27

Country of origin: USSR
Translated by: Carol S. Nack
Requester: FTD/TQTA
Approved for public release; distribution unlimited.

Accession For	
NTIS GRA&I	X
DDC TAB	
Unannounced	
Justification	
By	
Distribution/	
Availability Codes	
Dist.	Avail and/or special
A	

PREPARED BY:

TRANSLATION DIVISION
FOREIGN TECHNOLOGY DIVISION
WP-AFB, OHIO.

141600

U. S. BOARD ON GEOGRAPHIC NAMES TRANSLITERATION SYSTEM

Block	Italic	Transliteration	Block	Italic	Transliteration
А а	*А а*	A, a	Р р	*Р р*	R, r
Б б	*Б б*	B, b	С с	*С с*	S, s
В в	*В в*	V, v	Т т	*Т т*	T, t
Г г	*Г г*	G, g	У у	*У у*	U, u
Д д	*Д д*	D, d	Ф ф	*Ф ф*	F, f
Е е	*Е е*	Ye, ye; E, e*	Х х	*Х х*	Kh, kh
Ж ж	*Ж ж*	Zh, zh	Ц ц	*Ц ц*	Ts, ts
З з	*З з*	Z, z	Ч ч	*Ч ч*	Ch, ch
И и	*И и*	I, i	Ш ш	*Ш ш*	Sh, sh
Й й	*Й й*	Y, y	Щ щ	*Щ щ*	Shch, shch
К к	*К к*	K, k	Ъ ъ	*Ъ ъ*	"
Л л	*Л л*	L, l	Ы ы	*Ы ы*	Y, y
М м	*М м*	M, m	Ь ь	*Ь ь*	'
Н н	*Н н*	N, n	Э э	*Э э*	E, e
О о	*О о*	O, o	Ю ю	*Ю ю*	Yu, yu
П п	*П п*	P, p	Я я	*Я я*	Ya, ya

*ye initially, after vowels, and after ъ, ь; e elsewhere.
When written as ё in Russian, transliterate as yё or ё.

RUSSIAN AND ENGLISH TRIGONOMETRIC FUNCTIONS

Russian	English	Russian	English	Russian	English
sin	sin	sh	sinh	arc sh	$\sinh^{-1}$
cos	cos	ch	cosh	arc ch	$\cosh^{-1}$
tg	tan	th	tanh	arc th	$\tanh^{-1}$
ctg	cot	cth	coth	arc cth	$\coth^{-1}$
sec	sec	sch	sech	arc sch	sech^{-1}
cosec	csc	csch	csch	arc csch	csch^{-1}

Russian	English
rot	curl
lg	log

1000

CORRECTIVE ARITHMETIC CODES IN THE RESIDUAL CLASS SYSTEM

S. B. Payn

Designations. We will use Latin and Greek letters to designate whole numbers, $p_1, p_2, \ldots, p_n$ - prime numbers (sometimes reciprocal prime numbers in pairs),

$$P = p_1 p_2 \cdot p_3 \ldots p_n, \quad P_i = \frac{P}{p_i}, \quad Q_{i+1} = p_1 p_2 \ldots p_i$$
$$Q_1 = 1, \quad R_i = p_i \cdot p_{i+1} \ldots p_n, \quad (i = 1, 2, \ldots, n),$$

w_i - the group of remainders of reciprocal primes with modulus p_i, Π - the group of remainders of reciprocal primes with modulus P.

Further designations will be introduced in the appropriate place.

§1. Representation of Numbers in the Residual Class System

Let $0 \leqslant A < P$, $\alpha_1, \alpha_2, \ldots, \alpha_n$ - the smallest nonnegative remainders of number A with moduli of $p_1, p_2, \ldots, p_n$, respectively. We will call the expression

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_n) \qquad (1)$$

the representation of number A in the residual class system (abbreviated SOK).

The numbers $p_1, p_2, \ldots, p_n$ are called the bases of the system, and the numbers $\alpha_1, \alpha_2, \ldots, \alpha_n$ - the numbers of the given representation with moduli $p_1, p_2, \ldots, p_n$, respectively.

The whole numbers in the range [0, P) and representations (1) are in a one-to-one correspondence.

Actually, the remainders from division by the given numbers $p_1, p_2, \ldots, p_n$ are uniquely defined, on one hand, and on the other hand, we know that the system of comparisons

$$x \equiv \alpha_i \pmod{p_i} \quad (i = 1, 2, \ldots, n) \qquad (2)$$

has the unique solution

$$x \equiv A \pmod{P}, \quad 0 \leqq A < P, \tag{3}$$

with reciprocal prime numbers $p_1, p_2, \ldots, p_n$.

Remark. If the number A satisfies the inequality

$$0 \leqq A < R_i, \tag{4}$$

it is uniquely determined by the remainders for the moduli $p_i, p_{i+1}, \ldots, p_n$. Therefore, with condition (4), along with representation (1) we can write

$$A = (\alpha_i, \alpha_{i+1}, \ldots, \alpha_n). \tag{4'}$$

Analogously, with the condition

$$0 \leqq A < P_i \quad \text{or} \quad 0 \leqq A < Q_{n-i+1}, \tag{5}$$

we can write

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_n) \tag{6}$$

or

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_{n-i}). \tag{6'}$$

Unlike the position system, the arithmetic operations in the residual class system are performed step-by-step. Thus, if

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_n),$$
$$B = (\beta_1, \beta_2, \ldots, \beta_n)$$

are the representations of numbers A and B in the SOK, then if we consider $|\alpha_i \pm \beta_i|$ and $|\alpha_i \cdot \beta_i|$ to be the smallest nonnegative remainders of these numbers for the moduli $p_i\ (i = 1, 2, \ldots, n)$, the numbers $A \pm B$ and AB can be represented as

$$A \pm B = (\alpha_1 \pm \beta_1, \alpha_2 \pm \beta_2, \ldots, \alpha_n \pm \beta_n)$$
$$AB = (\alpha_1 \cdot \beta_1, \alpha_2 \cdot \beta_2, \ldots, \alpha_n \beta_n). \qquad (7)$$

This assertion is a direct result of the fact known from number theory that from

$$A \equiv \alpha_i \pmod{p_i}, \quad B \equiv \beta_i \pmod{p_i} \quad (i = 1, 2, \ldots, n)$$

it follows that

$$A \pm B \equiv \alpha_i \pm \beta_i, \quad A \cdot B \equiv \alpha_i \beta_i \pmod{p_i}$$
$$(i = 1, 2, \ldots, n).$$

Obviously, if we do not make certain limitations, the results of (7) of the execution of arithmetic operations in the SOK are only

obtained with precision down to the terms which are multiples of P.

Now we will consider the possibility of performing division operations in the SOK:

We know that the classes of remainders of reciprocal primes with the prime modulus p_i form, by multiplication, a finite group π_i, of order $p_i - 1$. The numbers 0, 1, 2, ..., $p_i - 1$ can be used as the representations of this group. If we consider representation (1) with the condition $\alpha_i \neq 0$ (i = 1, 2, ..., n), these representations also form, by multiplication, the finite group Π, which is the direct sum of the groups π_1, π_2, ..., π_n. Obviously, the order of group Π is equal to

$$\varphi(P) = (p_1 - 1)(p_2 - 1)\cdots(p_n - 1),$$

where $\varphi(P)$ is the Euler function.

It follows from the aforementioned that if we limit ourselves to elements of group Π, i.e., representations (1) do not contain zeroes, division can be carried out in the SOK step by step.

Let A and B be elements of group Π, and

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_n), \quad \alpha_i \neq 0$$
$$B = (\beta_1, \beta_2, \ldots, \beta_n), \quad \beta_i \neq 0 \qquad (i = 1, 2, \ldots, n)$$

- their representations in the SOK. The ratio A/B will be assigned the value

$$\frac{A}{B} = (\gamma_1, \gamma_2, \ldots, \gamma_n), \qquad (8)$$

where γ_i (i = 1, 2, ..., n) are defined as the solutions of the comparisons

$$\gamma_i \equiv \frac{\alpha_i}{\beta_i} \pmod{p_i} \quad (i = 1, 2, \ldots, n).$$

These comparisons also only have unique solutions when $\beta_i \neq 0$, while α_i can also assume the value 0. Therefore, we can eliminate the limitation $\alpha_i \neq 0$.

Two cases are possible:

1) the number A is evenly divisible by B; in this case, representation (8) gives the true value of the fraction A/B.

2) the number A is not evenly divisible by B; in this case, we will call representation (8) the formal representation of the fraction A/B (see [2]).

Representation (8) yields a certain whole number C in both cases. In the first case, we have C = A/B, and in the second - $C \equiv \frac{A}{B} \pmod{P}$.

We will consider the case when A is evenly divisible by B, but there are zeroes in β_i. Suppose, for example, that $\beta_i = 0$, where i is fixed. Then it fcllows from the divisibility of A by B that $\alpha_i = 0$, as well. Thus, with step-by-step division in the i-th step of representation (8), the undefined value 0/0 is present. In this case, since, obviously, A/B < P_i, the number C is determined by the representation

$$C = (\gamma_1, \gamma_2, \ldots, \gamma_{i-1}, \gamma_{i+1}, \ldots, \gamma_n)$$

(see (5), (6)). The problem of finding the numbers γ_i will be solved in the next section.

Later we will give examples illustrating the possibilities of using formal representaticns of the fractions.

Now we will consider a method of representing negative numbers in the SOK.

As we already pointed out, the set of representations (1) uniquely defines the numbers in the range [0, P). We will divide this range into two parts [0, P/2) and (P/2, P). We will call them the first and second halves of the given range, respectively. We will stipulate that the numbers in the first half are considered to be nonnegative, and the numbers in the second half - negative. Here the numbers in the first half are nomologous. We will equate the number A in the second half to the negative number A - P. Obviously, the set of all negative numbers fills the range (-P/2, 0) when $P/2 < A < P$. We will assign the representation of the number $A = -|A|$ in supplementary code. More precisely, if $A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ $0 \leqslant A < P/2$, then $-A = (p_1 - \alpha_1, p_2 - \alpha_2, \ldots, p_n - \alpha_n)$. This representation is equivalent to the number P - A, which belongs to the second half of the range [0, P).

Thus, we can speak of a one-to-one correspondence between the set of representations (1) and the numbers in the range $(-P/2, P/2)$[1].

Footnote: [1]We will point out that P/2 remains outside the consideration when P is even. End footnote

If the representation $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ is given, it suffices to

determine which half of the range [0, P) this number belongs to in order to establish the sign of the number which it represents.

Now, having defined the rational operations on the relative numbers in the residual class system, we can state the theorem.

Theorem 1. If the inequality

$$|f(x, y, z, \ldots)| < \frac{P}{2}, \qquad (9)$$

holds for the rational integer function f(x, y, z, ...) with rational coefficients (although they have a formal representation in the SOK) in a certain range of change in the variables, the values of this function for the indicated values of the arguments are calculated the same in the SOK, if we consider the results of the calculations to be the absolute least remainder with respect to modulus P. Here there are no values which would make the intermediate results go outside the range [0, P/2).

For the proof it suffices to note that if we compare the result of calculating the rational integer function in the SOK with respect to modulus P with its absolute least remainder, this absolute least remainder also gives us a single value of the function in the assigned range.

We will consider an example illustrating the calculation of an integral polynomial in the SOK.

Example 1. Let the bases of the system be $p_1 = 3$, $p_2 = 5$, $p_3 = 7$, $p_4 = 11$; then $P = 1155$.

We will calculate the value of the integral polynomial

$$f(x) = \frac{1}{2}x^3 - 5x^2 - \frac{23}{2}x$$

at $x = 13$.

In the decimal system, the calculations give us $f(13) = 104$, so that condition (9) of theorem 1 is satisfied.

We will write the representations of the numbers upon which we will operate: $x = 13 = (1, 3, 0, 2)$,

$$\frac{1}{2} = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right) = (2, 3, 4, 6) \quad \text{(formal representation)}$$

$5 = (2, 0, 5, 5)$,

$-5 = (1, 0, 2, 6)$ (supplementary code)

$$+\frac{23}{2} = \frac{(2, 3, 2, 1)}{(2, 2, 2, 2)} = (1, 4, 1, 6) \quad \text{(formal representation)}$$

$-\frac{23}{2} = (2, 1, 6, 5)$ (supplementary code).

We will perform the following operations in succession:

1) $x^3 = 13^3 = (1^3, 3^3, 6^3, 2^3) = (1, 2, 6, 8)$

(here the result exceeded the limits of the range [0; 1155),

2) $\frac{1}{2}x^3 = (2, 3, 4, 6)\ (1, 2, 6, 8) = (2, 1, 3, 4)$

(here we obtained the formal representation for the fraction $13^3/2$)

3) $-5x^2 = (1, 0, 2, 6)\cdot(1^2, 3^2, 6^2, 2^2) =$

$= (1, 0, 2, 6)\ (1, 4, 1, 4) = (1, 0, 2, 2),$

4) $-\frac{23}{2}x = (2, 1, 6, 5)\cdot(1, 3, 6, 2) = (2, 3, 1, 10)$

(we also have the formal representation here)

5) $\frac{1}{2}x^3 - 5x^2 - \frac{23}{2}x =$

$$\begin{array}{r} (2, 1, 3, 4) \\ +(1, 0, 2, 2) \\ (2, 3, 1, 10) \\ \hline (2, 4, 6, 5) = 104 \end{array}$$

The same result as in the decimal system was obtained.

§2. Conversion of Numbers from the Residual Class System into the

Position System and Back

In order to convert numbers from the SOK into the position system, A. Svoboda [2] recommends the following method, called the method of orthogonal bases.

Assume that the bases of the system of SOK $p_1, p_2, \ldots, p_n$ are assigned. We will calculate the values (in the position system) of the numbers

$$\begin{aligned} B_1 &= (1, 0, 0, \ldots, 0), \\ B_2 &= (0, 1, 0, \ldots, 0), \\ &\cdots\cdots\cdots \\ B_n &= (0, 0, 0, \ldots, 1), \end{aligned}$$

in advance. The numbers $B_1, B_2, \ldots, B_n$ are called orthogonal bases. If $A = (\alpha_1, \ldots, \alpha_n)$ is the representation of the number A in the SOK, in order to find its value in the position system, it suffices to calculate the expression

$$x = \alpha_1 B_1 + \alpha_2 B_2 + \cdots + \alpha_n B_n. \qquad (10)$$

Obviously, $A \equiv x \pmod{P}$.

If we search for the number A in the range [0, P), obviously

$$A = x - r_A \cdot P, \qquad \text{where} \quad r_A = \left[\frac{x}{P}\right].$$

According to I. Ya. Akushkiy, the number r_A is called the rank of the number A.

We will consider an example.

Example 2. We will consider the bases of the system to be the same as in example 1.

Then $B_1 = 385$, $B_2 = 231$, $B_3 = 330$, $B_4 = 210$.

Let A = (2, 4, 6, 5); then

$$x = 2 \cdot 385 + 4 \cdot 231 + 6 \cdot 330 + 5 \cdot 210 = 4724$$
$$r_A = \left[\frac{4724}{1155}\right] = 4, \quad A = 4724 - 4 \cdot 1155 = 104.$$

The advantage of the method of orthogonal bases is the simplicity of equation (10).

We will consider the generalized position system (system OPS) [1], in which the n-step number A is represented as

$$A = a_1 + a_2 Q_2 + \cdots + a_n Q_n \qquad (11)$$

where

$$\frac{Q_{i+1}}{Q_i} = p_i; \quad Q_1 = 1, \quad (i = 1, 2, \ldots, n).$$

Footnote: [1] A. Svoboda [2] calls this system a system with a mixed base. End footnote

Then it follows from (11) that

$$A = a_1 + a_2 p_1 + a_3 p_1 \cdot p_2 + \cdots + a_n p_1 \cdot p_2 \cdots p_{n-1};$$

the numbers p_1, p_2, ..., p_n are the bases of the generalized position system.

If we also assume here that the numbers a_i are the numbers 0, 1, 2, ..., $p_i - 1$, the volume of the range of numbers represented in this system is equal to $P = p_1 \cdots p_n$.

It is obvious that the ordinary position system is obtained from the generalized system if we set

$$\frac{Q_{i+1}}{Q_i} = p, \; Q_1 = 1 \; (i = 1, 2, \ldots, n-1),$$

where p is the base of the ordinary position system.

The procedure of successively obtaining the values of the numbers in the corresponding representations can be realized by the following process:

1) A is divided by p_1: this gives us $[A/p_1] = A_1$ and $A - A_1p_1 = a_1$

2) A_1 is divided by p_2: this gives us $[A_1/p_2] = A_2$ and $A_1 - A_2p_2 = a_2$

.....

i) A_{i-1} is divided by p_i; this gives us $[A_{i-1}/p_i] = A_i$ and $A_{i-1} - A_ip_i = a_i$

.....

n) A_{n-1} is divided by p_n; this gives us $[A_{n-1}/p_n] = A_n$ and $A_{n-1} - A_np_n = a_n$. (12)

Let the numbers $p_1, p_2, \ldots, p_n$ serve as the simultaneous bases of the SOK and the OPS [generalized position system]. We will number the steps in the SOK and the OPS in the same order when the intervals of the change in the numbers of the steps with the same values coincide.

We will write the representation of the number A in the OPS as

follows

$$A = [a_1, a_2, \ldots, a_n]. \qquad (13)$$

The ranges of the numbers uniquely represented in the systems SOK and OPS thus constructed obviously coincide. Therefore, we can speak of a one-to-one correspondence between the set of representations (2) of numbers in the SOK and the set of representations (13) of numbers in the OPS. The problem consists of establishing the process of realizing this correspondence.

We will show that algorithm (12) can be successfully used to convert the representation of numbers from the SOK into the OPS, whereupon the entire conversion procedure can be realized in the SOK.

Actually, the comparisons below follow from equations (12)

$$\begin{aligned} A &\equiv a_1 \pmod{p_1}, & 0 \leqq a_1 < p_1, \\ A_1 &\equiv a_2 \pmod{p_2}, & 0 \leqq a_2 < p_2, \\ &\cdots\cdots\cdots\cdots\cdots\cdots \\ A_{n-1} &\equiv a_n \pmod{p_n}, & 0 \leqq a_n < p_n, \end{aligned}$$

where a_1, a_2, ..., a_n are the digits of the OPS. Thus, the problem of finding the digits of the OPS is reduced to finding the remainders of the numbers A, A_1, A_2, ... A_{n-1} according to the moduli p_1, p_2, ..., p_n, respectively. This problem is realized in the SOK as

follows: let

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$$

be the represented numbers A in the SOK and

$$p_1 < p_2 < \cdots < p_n. \qquad (14)$$

1) obviously, $A \equiv \alpha_1 \pmod{p_1}$, which means that $a_1 = \alpha_1$;

2) the integer $A_1 = \frac{A - \alpha_1}{p_1}$ obviously satisfies the condition $A_1 < R_2$; therefore (see (4), [4']), it is determined by its last (n - 1) numbers of representaticn in the SOK. We will calculate these numbers in the SOK

$$A_1 = \frac{A - \alpha_1}{p_1} = (\alpha_2^{(1)}, \alpha_3^{(1)}, \ldots, \alpha_n^{1}),$$

where

$$\alpha_i^{(1)} \equiv \frac{\alpha_i - \alpha_1}{p_1} \pmod{p_i}, \quad i = 2, 3, \ldots, n,$$

which means that

$$a_2 = \alpha_2^{(1)}; \qquad (15)$$

3) analogously to above, we will obtain

$$A_2 = \frac{A_1 - \alpha_2^{(1)}}{p_2} = (\alpha_3^{(2)}, \alpha_4^{(2)}, \ldots, \alpha_n^{(2)}),$$

where

$$\alpha_i^{(2)} \equiv \frac{\alpha_i^{(1)} - a_2}{p_2} \pmod{p_i}; \quad i = 3, 4, \ldots, n,$$

which means that

$$\alpha_3^{(2)} = a_3,$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$i+1) \qquad A_i = \frac{A_{i+1} - a_i}{p_i} = (\alpha_{i+1}^{(i)}, \alpha_{i+2}^{(i)}, \ldots, \alpha_n^{(i)}),$$

where

$$\alpha_{i+k} \equiv \frac{\alpha_{i-1}^{(i-1)} - a_i}{p_i} \pmod{p_{i+k}}: \quad k = 1, 2, \ldots, n-i,$$

which means that

$$\alpha_{i+1}^{(i)} = a_{i+1} \qquad (15)$$

$$\cdots\cdots$$

n) finally

$$A_{n-1} = \frac{A_{n-2} - \alpha_{n-1}}{p_{n-1}} = (\alpha_n^{(n-1)})$$

and

$$a_n = \alpha_n^{(n-1)}.$$

Thus, we obtained all of the values of the representation of the number A in the OPS.

If we eliminate condition (14), the process is somewhat complicated because it becomes necessary to convert the values of the OPS obtained into the sCK.

As we can see, the conversion process can be realized in the arithmetic unit operating in the SOK system.

We will consider the opposite problem. Suppose that we know the representation (13) of the number A in the CPS. It is necessary to find its representation in the SCK (1).

We will consider that we know the representations of the numbers $Q_1, Q_2, \ldots, Q_n$ in the SCK. Specifically, let

$$\begin{aligned}
Q_1 &= (1, 1, \ldots, 1),\\
Q_2 &= (0, Q_2^2, \ldots, Q_2^n),\\
Q_3 &= (0, 0, Q_3^3, \ldots, Q_3^n),\\
&\cdots\cdots\cdots\cdots\\
Q_n &= (0, 0, \ldots, 0, Q_n^n).
\end{aligned}$$

Then it follows from formula (11) that the values $\alpha_1, \alpha_2, \ldots, \alpha_n$ can be found from the comparisons

$$\begin{aligned}
&\alpha_1 \equiv a_1 (\mathrm{mod}\, p_1),\\
&\alpha_2 \equiv a_1 + a_2 Q_2^2 (\mathrm{mod}\, p_2),\\
&\alpha_3 \equiv a_1 + a_2 Q_3^2 + a_3 Q_3^3 (\mathrm{mod}\, p_3),\\
&\cdots\cdots\cdots\cdots\cdots\cdots\\
&\alpha_n \equiv a_1 + a_2 Q_n^{(n)} + \cdots + a_n Q_n^n (\mathrm{mod}\, p_n).
\end{aligned} \tag{16}$$

Obviously, these calculations can also be realized in the arithmetic unit operating in the SOK.

If we consider the numbers $a_1, a_2, \ldots, a_n$ in system (16) to be known, the solution of this system gives us the values $a_1, a_2, \ldots, a_n$.

We will consider some examples.

Example 3. We will consider the bases of the system to be the same as in example 1. Again, let A = (2, 4, 6, 5). We will find the representation of this number in the OPS. Obviously, we will have $Q_1 = 1$, $Q_2 = 3 = (0, 3, 3, 3)$, $Q_3 = 3 \cdot 5 = (0, 0, 1, 4)$; $Q_4 = 3 \cdot 5 \cdot 7 = 0, 0, 0, 6)$.

We will successively find the digits of the OPS according to system (15).

1) $a_1 = \alpha_1 = 2$;

2) $A_1 = \frac{(2, 4, 6, 5) - (2, 2, 2, 2)}{3} = \frac{(0, 2, 4, 3)}{3} =$

$= \frac{(\ , 2, 4, 3)}{3} = \left(\ , \frac{2+2\cdot 5}{3}, \frac{4+2\cdot 7}{3}, \frac{3}{3}\right) =$

$= (\ , 4, 6, 1), \quad a_2 = 4;$

3) $A_2 = \frac{(\ , 4, 6, 1) - (\ , 4, 4, 4)}{5} = \frac{(\ , 0, 2, 8)}{5} =$

$= \frac{(\ , \ , 2, 8)}{5} = \left(\ , \ , \frac{2+4\cdot 7}{5}, \frac{8+2\cdot 11}{5}\right) =$

$= (\ , \ , 6, 6), \quad a_3 = 6;$

4) $A_3 = \frac{(\ , \ , 6, 6) - (\ , \ , 6, 6)}{11} = (\ , \ , 0, 0),$

$a_4 = 0.$

Thus, the representation of the OPS has the form

$$A = [2, 4, 6, 0] = 2 + 4\cdot 3 + 6\cdot 15 + 0\cdot 105 = 104.$$

We will conduct the opposite procedure, i.e., we will find the representation of the number A = [2, 4, 6, 0] in the residual class system. According to system (16), we will obtain

$$A = (2, 2+4\cdot 3, 2+4\cdot 3+6\cdot 1, 2+4\cdot 3+6\cdot 4+0\cdot 6) = (2, 4, 6, 5).$$

The method found for converting the representations of the numbers into the OPS from the SCK and back can be used for performing operations of comparing numbers in the arithmetic unit operating in the SOK.

In order to perform the operation of comparing two numbers A and B assigned in the SOK, it suffices to convert these numbers into the OPS, then compare the values of the representations obtained, going from the higher-order digits to the lower-order ones (in our designations, from right to left). If A > B, the first nonzero difference in the values will be positive, and vice versa.

We already pointed out that when performing operations on relative numbers, the problem of determining the sign of the number is reduced to determining which of the two halves of the range $[0, P)$ – the number belongs to the first $[0, P/2)$, or the second $(P/2, P)$. This problem is solved by comparing this representation with the representation of the number $[P/2]$. (For a fixed system of bases, the representation of the number $[P/2]$ can be stored in the memory of both the SOK and the OPS).

The problem of whether the number A, with representation (1), belongs to intervals $[0, Q_{i+1})$, $i = 1, 2, \ldots, n-1$ is also interesting. In order to solve this problem, it suffices to obtain the representation of the number A in the OPS (13). The following results are completely obvious:

when $a_n = 0$, we will have $0 \leqslant A < Q_n$

when $a_n = a_{n-1} = \cdots = a_{n-l} = 0$ we will have $0 \leqslant A < Q_{n-l}$. (17)

In §1, when considering the possibilities of performing the operation of division in the SOK, we noted the case when A is divided evenly by B, but the numbers of the representation of B also contained zero numbers. It was shown that in this case, the quotient C is determined by the digits which do not contain the undefined value 0/0. It is necessary to reveal these undefined values in order to subsequently perform operations with whole-number representations. Obviously, in order to do this it suffices to find the remainders of the number C with respect to the moduli of those digits in which this indeterminancy exists.

We will describe one possible method of solving this problem in the SOK.

Suppose the undefined value 0/0 is present in the digits for the moduli p_i ($i = i_1, i_2, \ldots, i_k$). If we designate

$$C=\left(\gamma_1, \ldots, \gamma_{i_1-1}, \frac{0}{0}, \gamma_{i_1+1}, \ldots, \gamma_{i_2-1}, \frac{0}{0}, \gamma_{i_2+1}, \ldots, \gamma_n\right), \qquad (18)$$

the problem consists of finding the numbers $\gamma_{i_1}, \gamma_{i_2}, \ldots, \gamma_{i_k}$.

In order to find these numbers, we will convert the

(n - k)-digit representation (18) into the OPS with the bases

$$p_1, p_2, \ldots, p_{i_1-1}, p_{i_1+1}, \ldots, p_{i_k-1}, p_{i_k+1}, \ldots, p_n.$$

We will obtain

$$\begin{aligned} C = [c_1, \ldots, c_{i_1-1}, c_{i_1+1}, \ldots, c_{i_k-1}, c_{i_k+1}, \ldots, c_n] = \\ = c_1 + c_2 p_1 + \cdots + c_{i_1+1} p_1 \cdot p_2 \cdots p_{i_1-1} + \cdots + \\ + c_n p_1 \cdot p_2 \cdots p_{i_1-1} p_{i_1+1} \cdots p_{n-1}. \end{aligned}$$

We will assume that we know the representations for the moduli $p_1, p_2, \ldots, p_n$ in the SCK:

$$\begin{aligned} p_1 &= (0, p_1^2, p_1^3, \ldots, p_1^n) \\ p_2 &= (p_2^1, 0, p_2^3, \ldots, p_n^n) \\ &\cdots\cdots\cdots\cdots \\ p_n &= (p_n^1, p_n^2, \ldots p_n^{n-1}, 0). \end{aligned}$$

Then the unknown digits can be found from the comparisons

$$\begin{gathered} \gamma_i \equiv c_1 + c_2 p_1^i + \cdots + c_{i_1+1} \cdot p_1^i p_2^i \cdots p_{i_1-1}^i + \cdots (\mathrm{mod}\, p_i) \quad (19) \\ i = i_1, i_2, \ldots, i_k. \end{gathered}$$

Here it can become necessary to convert the digits of the repesentation of the generalized position system into the SOK.

These calculations are simplified if we have representations in the SOK for all possible products comprised of the assigned moduli and taken one by one, two by two, etc.

The conversion of the representation of a number from the SOK into the OPS can be realized by the method of orthogonal bases.

First, we will point out that the operation of adding numbers in the OPS is done in the same way as in ordinary position systems: by the successive addition of the numbers, from smaller-order to higher-order, with the ordinary operation of inter-digit carryover, if the result of addition does not go outside the range [0, P). If the result of addition exceeds the range, i.e., a higher-order digit overflowed, it is necessary either to broaden the range, or to eliminate the consideration of overflow and take the remainder with respect to modulus P as the result of the operation.

Suppose that for a fixed system of bases which satisfies condition (14) we know the representations of the orthogonal bases in the OPS

$$B_i = [0, 0, \ldots, b_i^i, \ldots, b_n^i] \quad (i = 1, 2, \ldots, n).$$

We will consider the orthogonal numbers

$$\alpha_i B_i \equiv (0, 0, \ldots, 0, \alpha_i, 0, \ldots, 0) \quad (i = i, 2, \ldots, n) \quad (0 < \alpha_i < p_i).$$

The comparison sign has been provided here, since the product $\alpha_i B_i$ can be larger than P. We will designate

$$B_{i\alpha_i} = (0, 0, \ldots, 0, \alpha_i, 0, \ldots, 0).$$

Thus,

$$\alpha_i B_i \equiv B_{i\alpha_i} \pmod{P},$$
$$0 \leqq B_{i\alpha_i} < P.$$

We will find the representation in the OPS for the numbers $B_{i\alpha_i}$. In view of condition (14), the number $\alpha_i < p_i$ can serve as a value in each of the digits with the bases

$$p_i, p_{i+1}, \ldots, p_n.$$

Therefore, the calculation of the product $\alpha_i B_i$ is reduced to multiplication by a one-digit number, i.e., it is equivalent to α_i successive additions. If we do not consider the overflow of the n-th digit in this case, as a result of the calculations we obtain the representation in the CPS for the number $B_{i\alpha_i}$ instead of $\alpha_i B_i$.

Let the representations obtained be

$$B_{i\alpha_i}[0, 0, \ldots, 0, b^i_{i\alpha_i}, b^i_{i+1,\alpha_i}, \ldots, b^i_{n,\alpha_i}],$$
$$\alpha_i = 1, 2, \ldots, p_i - 1, \quad i = 1, 2, \ldots, n.$$

Now we can use formula (10) to convert the representation of the number A < P from the SOK into the OPS. If

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_n),$$

then

$$A \equiv \alpha_1 B_1 + \alpha_2 B_2 + \cdots + \alpha_n B_n \equiv$$
$$\equiv B_{1\alpha_1} + B_{2\alpha_2} + B_{3\alpha_3} + \cdots + B_{n\alpha_n} (\mathrm{mod}\ P).$$

Again, if we do not consider possible overflows in the n-th higher-order digit when calculating the latter sum, we will immediately obtain a precise representation of the number A in the OPS.

For an illustration, we will again consider example 3.

First we will find the representation of the orthogonal bases in the OPS

$$B_1 = [1, 3, 4, 3],$$
$$B_2 = [0, 2, 1, 2],$$
$$B_3 = [0, 0, 1, 3],$$
$$B_4 = [0, 0, 0, 2].$$

we will have

$$A = (2, 4, 6, 5) \equiv 2 B_1 + 4 B_2 + 6 B_3 + 5 B_4 =$$
$$= B_{12} + B_{24} + B_{36} + B_{45} =$$

$$\begin{array}{r} [2, 1, 2, 7] \\ [0, 3, 5, 8] \\ +[0, 0, 6, 7] \\ [0, 0, 0, 10] \\ \hline A = [2, 4, 6, 0]. \end{array}$$

§3. Corrective Arithmetic Codes in the Residual Class System

In digital computers, information must ordinarily undergo a long series of different transformations before the final result is obtained. In order for this result to be reliable, extremely rigid requirements are imposed on the reliability of digital computers. Different methods of checking are used to provide the reliability of the operation of computers. The use of codes with redundancy, so-called corrective codes, is considered to be the most promising.

Corrective codes intended for data transmission systems are widely known today. The most popular are the Heming codes, which use the "parity" method of checking. Codes suitable for checking arithmetic operations were proposed by Brown [3]. However, it is difficult to use these codes in practice because computers operate in position systems of calculation, whereas check operations should be made on residues. This contradiction will not take place if the computer itself operates in the residual class system. I. Ya.

Akushskiy [1] was the first to draw attention to this situation, recommending that the corrective properties of SOK be studied.

The codes considered below can be used for correcting errors which arise during data transmission and when performing arithmetic operations in the SOK.

We will consider the representations

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_n) \qquad (1)$$

to be a code combination, where the digits α_1, α_2, ..., α_n can be assigned as binary, like in the multiposition representation. We know that a certain whole number from the range [0, P) corresponds to each representation (1). We will use the representations (1) corresponding to the numbers included in a certain part of the assigned range [0, P) for transmitting data or performing arithmetic operations. This makes it possible to correct the errors.

We will assume that representations (1), which satisfy the condition

$$0 \leqq A < \frac{P}{p_n} = P_n. \qquad (2)$$

are used for transmitting messages or performing arithmetic operations.

In this case, the number A is uniquely defined by a $(n-1)$-digit representation $(\alpha_1, \alpha_2, \ldots, \alpha_{n-1})$ (see (5), §1). Therefore, the digit with the base p_n can be considered to be redundant. According to Heming, the effectiveness of the code or the degree of the drop in the capacity of the channel is detefined as the value R, equal to the ratio of the whole number of transmitted binary symbols to the minimum number of symbols necessary for transmitting this same information.

We will assume that a binary channal is used for data transmission. We will also assume that the bases of the SOK $p_1, p_2, \ldots, p_n$ are successive prime numbers. With this coding, $[\log_2 P_i] + 1$ binary symbols on base $p_i > 2$ are needed for transmitting the number α_i; one symbol is necessary for transmitting the number α_1. The total number of binary symbols necessary for transmitting representations (messages) (1) will be equal to

$$1 + \sum_{i=2}^{n} ([\log_2 p_i] + 1),$$

and the minimum number of symbols necessary for transmitting the same message will be equal to

$$1 + \sum_{i=2}^{n-1} ([\log_2 p_i] + 1).$$

We will obtain the following relationship for the value of R:

$$R = \frac{1 + \sum_{i=2}^{n} ([\log_2 p_i] + 1)}{1 + \sum_{i=1}^{n-1} ([\log_2 p_i] + 1)}.$$

We will evaluate this expression. We have

$$R = 1 + \frac{[\log_2 p_n] + 1}{1 + \sum_{i=2}^{n-1} ([\log_2 p_i] + 1)}$$

when n > 3, obviously,

$$R < 1 + \frac{\log_2 p_n + 1}{\sum_{i=1}^{n-1} \log_2 p_i} = 1 + \frac{\log_2 p_n + 1}{\log_2 (p_1 \cdot p_2 \cdots p_{n-1})} =$$

$$= 1 + \frac{\log_2 p_n}{\log_2 P_n} + \frac{1}{\log_2 P_n}.$$

The inequality

$$p_n^{(n-1)/2} < p_1 \cdot p_2 \cdots p_{n-1} = P_n.$$

is proven in number theory. Taking its logarithm, we obtain

$$\frac{n-1}{2} \log_2 p_n < \log_2 P_n$$

or

$$\frac{\log_2 p_n}{\log_2 P_n} < \frac{2}{n-1}.$$

Using this inequality, we will obtain

$$R < 1 + \frac{2}{n-1} + \frac{2}{n-1} \cdot \frac{1}{\log_2 p_n}. \qquad (3)$$

We will study the corrective capabilities of the codes in question with one redundant digit for base p_n.

First we will define the concept of error. We will consider a single error to be the distortion of any one number of the n-digits of representation (1), whereupon the distortion is limited only by the value of the base. We will consider a k-fold error to be the distortion of k numbers of the representation.

Let $A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ be the transmitted message; we will introduce the designation $\overline{A} = (\overline{\alpha}_1, \overline{\alpha}_2, \ldots, \overline{\alpha}_n)$ for the received message.

We will consider message $\overline{A} = (\overline{\alpha}_1, \overline{\alpha}_2, \ldots, \overline{\alpha}_n)$ to be error-free if

$$0 \leqq \overline{A} < P_n.$$

And if

$$\overline{A} \geqq P_n,$$

we will consider the corresponding representation to be erroneous.

We will point out that it is always true that $0 \leqslant \bar{A} < P$ and, as stipulated, $0 \leqslant A < P_n$.

If message (1) is transmitted, in general, any of the possible P messages can be received. Here P_n of these will be received as error-free messages, and $P - P_n$ - as erroneous. Thus, the number $P - P_n$ shows the total number of detectable possible errors. The ratio $\frac{P - P_n}{P} = \frac{p_n - 1}{p_n}$ can serve as the measure of the detection capacities of a code. It is interesting to calculate the number of detected single, double, etc., errors.

We will assume that the bases of the SOK are arranged in order of increasing value, i.e.,

$$p_1 < p_2 < \cdots < p_n.$$

Then the number $P_1, P_2, \ldots, P_n$ will satisfy the inequalities

$$P_1 > P_2 > \cdots > P_n. \qquad (5)$$

We will show that in this case, the presence of the redundant base p_n is sufficient for detecting all single errors.

Theorem 1. Let $A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, $0 \leqslant A > P_n$ be the

transmitted message or the precise result of performing arithmetic operations in the SOK, and $\overline{A} = (\overline{\alpha}_1, \overline{\alpha}_2, \ldots, \overline{\alpha}_n)$ - the message received or the result of the calculation obtained in the SOK. Then, if $\overline{\alpha}_i = \alpha_i$ when $i \neq k$ and $\overline{\alpha}_k \neq \alpha_k$ $(i = 1, 2, \ldots, n)$, we have the inequality

$$\overline{A} \geqq P_m,$$

which detects the error.

Proof. First, we will point out that the distortion of the digit $k_k (\overline{\alpha}_k \neq \alpha_k)$ means the addition of the value lB_k to the number A, where $B_k = (0, 0, \ldots, 0, 1, 0, \ldots, 0)$ with the units of the k-th digit, i.e.,

$$\overline{A} \equiv A + l B_k \pmod{P}, \quad 0 < l < p_k,$$

but $B_k = sP_k$, where $0 < s < p_k$, which means that

$$\overline{A} \equiv A + ls P_k \pmod{P}.$$

But since $ls = qp_k + r$, $0 < r < p_k$, and $p_k P_k = P$, we obtain

$\overline{A} \equiv A + r P_k \pmod{P}$, where $0 < r < p_k$. Further, we will point out that when $0 \leqq A < P_n$, the number $A + rP_k$ is located in the range $[0, P)$ and, furthermore, we have (5).

Therefore, the inequality

$$A + r P_k \geqq P_n.$$

holds in the range [0, P). Whence it follows that

$$\overline{A} = A + r P_k \geqq P_n,$$

which also had to be proven.

We will look at an example. We will consider the bases to be the same as in the examples from the preceding sections: $p_1 = 3$, $p_2 = 5$, $p_3 = 7$, $p_4 = 11$, $P_n = 105$.

Example 4. Let A = 1 = (1, 1, 1, 1) be the transmitted message and A = (1, 1, 4, 1) - the received message. Calculating the value of A, we will obtain

$$\overline{A}=(1, 1, 4, 1)=1\cdot 385+1\cdot 231+4\cdot 330+1\cdot 210=2146\equiv 991>105;$$

which means that an erroneous message has been received.

The process of detecting the inaccuracy of the representation obtained can be realized in the arithmetic unit operating in the SOK. Actually, for this purpose it suffices to convert the representation obtained into the generalized position system with the same bases as in this SOK (see (17), §2). If the higher-order digit of the CPS a_n

turns out to be equal to zero, the representation is error-free, but if $a_n \neq 0$, the representation obtained is erroneous. Thus, in the example in question, the representation of the number A in the OPS has the form

$$\overline{A} = 1 + 0.3 + 3.15 + 9.105,$$

so that $a_n = 9 \neq 0$.

It is very obvious that if we know the digit of the SOK which contains the error, it is easy to correct. Actually, if we use A_k to designate the (n - 1)-digit repesentation obtained from expression (1) by subtracting the digit a_k, in view of conditions (2) and (5), we will have $A_k = A$ when $k = 1, 2, \ldots, n$. Therefore, if we know that the error occurred in the digit with the number k, $\overline{A}_k = A_k = A$. In order to obtain the number $A = A_k$, it suffices to subtract the value P_k from it until a number which lies in the range $[0, P_k)$ is obtained. If it is necessary to determine the number a_k in a device operating in the SOK, we proceed in the same manner as in §2 with the detection of the undefined values of the form 0/0.

The possibilities of correcting a single error for a code with one redundant digit for base p_n are determined by the following theorems.

Theorem 2. Assume that $0 \leqslant A < P_n$ and that we know that only a single error is possible in the received message $\overline{A} = (\overline{\alpha}_1, \overline{\alpha}_2, \ldots, \overline{\alpha}_n)$. Let $\overline{A}_k = (\overline{\alpha}_1, \overline{\alpha}_2, \ldots, \overline{\alpha}_{k-1}, \overline{\alpha}_{k+1}, \ldots, \overline{\alpha}_n)$ designate an (n - 1)-digit representation obtained by the subtraction of the number $\overline{\alpha}_k$. Then, if $\overline{A}_k \geqslant P_n$, the number α_k of the received representation is error-free.

Proof. First we will point out that the inequality $\overline{A}_k \geqslant P_n$ is possible when k < n, since we have $P_k > P_n$ in this case. Further, inequality $\overline{A}_k \geqslant P_n$ means that the (n - 1)-digit representation of $\overline{A}_k$ contains an erroneous number. However, by definition only one error is possible; therefore, number α_k is error-free.

Corrolary. When the inequality $\overline{A}_k \geqslant P_n$ holds for the values n = 1, 2, ..., (n - 1), the number of the n-th digit is erroneous.

Actually, according to the theorem, in this case the numbers α_k, k = 1, 2, ... (n - 1) will be error-free, which means that the number of the n-th digit α_n will be erroneous.

The case discussed in the corrolary is possible. This follows from theorem 3.

Theorem 3. Suppose that only a single error is possible in the

received message in $\overline{A} = (\overline{\alpha_1}, \overline{\alpha_2}, \ldots, \overline{\alpha_n})$. If the value of $\overline{A}$ satisfies one of the irequalities

$$P_n \leqq \overline{A} < P_{n-1} \qquad (6)$$

or

$$P_{n-1}(p_{n-1} - 1) + P_n < \overline{A} < P, \qquad (6a)$$

the number of the n-th digit will ne erroneous.

Proof. According tc the corrclary of theorem 2, in case (6) it will suffice to show that the equation

$$\overline{A_k} = \overline{A}$$

holds for values of k = 1, 2, ..., (n - 1).

But these equations follow rrom conditions (6), (5) and the uniquaness of the representation of the numbers of the range $[0, P_k)$ in the SOK.

We will prove the theorem ror case (6a). We know that the distortion of the digit a_k means the additicn of the value $r_k P_k$ to the number A, where $0 < k \leqq n$, $0 < r_k < p_k$, or, equivalently, the value

$$P_k(p_k - r_k),$$

where

$$0 < k \leqq n, \quad 0 < r_k < p_k.$$

We will have

$$P_{n-1} < P_{n-2} < \cdots < P_2 < P_1,$$

whence

$$P + P_n - P_{n-1} > \cdots > P + P_n - P_1.$$

Based on condition (6a), we can write

$$\bar{A} > P - P_{n-1} + P_n > \cdots > P - P_1 + P_n$$

or, in other words

$$\bar{A} > P_{n-1}(p_{n-1} - 1) + P_n > \cdots > P_1(p_1 - 1) + P_n.$$

Furthermore, it is obvious that

$$P_k(p_k - 1) \geqq P_i(p_k - r_k), \quad 0 < k \leqq n - 1, \quad 0 < r_k < p_k,$$

therefore,

$$\bar{A} > P_k(p_k - r_k) + P_n, \quad k = 1, 2, \ldots, (n - 1), \quad 0 < r_k < p_k.$$

This means that none of the numbers of the form

$$A + P_k(p_k - r_k), \quad 0 \leqq A < P_n, \quad k = 1, 2, \ldots, (n-1)$$
$$0 < r_k < p_k$$

can satisfy inequality (6a). Consequently, it is impossible to have an error in the digits with numbers k = 1, 2, ..., (n - 1); this is equivalent to a confirmation.

Theorem 3a. If we also assume that there are no even numbers among the bases, the error will also be in the n-th digit when the received message A satisfies the inequality

$$\frac{P}{2} - \frac{P_{n-1}}{2} + P_n < \overrightarrow{A} < \frac{P}{2} + \frac{P_{n-1}}{2}.$$

The proof follows from the inequalities

$$\frac{p_i - 1}{2} P_i < \frac{p_n - 1}{2} P_n < \frac{p_n + 1}{2} P_n < \frac{p_i + 1}{2} P_i.$$

We can approach the study of the correcting capacities of a code with one redundant digit somewhat differently.

Let

$$A = (\alpha_1, \alpha_2, \ldots \alpha_n)$$

be the transmitted message, and

$$\overline{A} = (\overline{\alpha}_1, \overline{\alpha}_2, \ldots, \overline{\alpha}_n)$$

- the received message.

We will call the representation

$$\Delta = \overline{A} - A = (\overline{\alpha}_1 - \alpha_1, \overline{\alpha}_2 - \alpha_2, \ldots, \overline{\alpha}_n - \alpha_n)$$

the distortion vector.

By definition,

$\overline{A} = A + \Delta$, where $\Delta = 0$ when $\overline{A} = A$ and $\Delta \neq 0$ when $\overline{A} \neq A$. If we obtain an erroenous message $\overline{A} \geqq P_n$, we can construct the total system of possible distorting vectors.

Lemma 1. Let $\overline{A} \geqq P_n$ be the received message. Then the numerical values of all the possible distorting vectors are

$$\overline{A}, \overline{A} - 1, \overline{A} - 2, \ldots, \overline{A} - (P_n - 1). \qquad (8)$$

Proof. Actually, if $\overline{A} \geqq P_n$ is the message obtained, we are justified in assuming that any of the P_n error-free representations with numerical values of 0, 1, 2, ..., $P_n - 1$ was transmitted.

If the representation

$$0 = (0, 0, \ldots, 0)$$

was transmitted, the distorting vector Δ_0 will obviously be the vector $\overline{A}$ itself

$$\overline{A} = 0 + \Delta_0.$$

If the message

$$1 = (1, 1, \ldots, 1),$$

was transmitted, the distorting vector Δ_1 will be the vector $\overline{A} - 1$, since

$$\overline{A} = A + \Delta_1 = 1 + (\overline{A} - 1).$$

Continuing in this fashion, we obtain series (8).

In the future, we will designate system (8) as

$$\Delta_0, \Delta_1, \ldots, \Delta_{P_n - 1} \qquad (9)$$

and we will call it the total system of distortion vectors.

Lemma 2. If the erroneous message $\overline{A} \geqslant P_n$ is received, there is precisely one vector in the total system of distortion vectors (9) in which the first (n - 1) numbers are equal to zero, while the number

of the n-th digit is nonzero.

Proof. Since the numbers (9) form the total system of remainders for modulus P_n, they contain precisely one number which is a multiple of P_n; we will designate it as $\bar{A}$. The first (n - 1) digits of the vector A are obviously equal to zero; furthermore, because of condition $P_n \leqslant A < P$, if even one of its digits is nonzero, the last digit of the vector $\bar{A}$ is nonzero. The lemma is proven.

It follows from lemma 1 that the problem of correcting the error in the message obtained can be solved by the correct selection of one of the possible distortion vectors (9). If this selection is made, we will obtain the corrected message in the form of the difference $A = \bar{A} - A$. If we assume the possibility of only single errors, it is logical to begin the problem of correcting the erroneous message with the selection of the vectors from system (9) which have only one nonzero number. It follows from lemma 2 that there is always one such vector with a nonzero number in the n-th digit in complete system (9). This means that we can always assume the presence of a single error in the last digit. The conditions of theorem 3 indicate certain ranges of the value of $\bar{A}$ for which the total system of vectors (9) contains one vector each with one nonzero digit. If the number of these vectors in system (9) turns out to be greater than one, the correct selection of the necessary distortion vector can only be made

with certain additional assumptions (e.g., the assumption of the impossibility of an error in the n-th digit), or on the basis of probability considerations.

Now we will assume that the redundant base p_n satisfies the condition

$$p_n > p_i \cdot p_j \quad 1 \leqq i < j < n \qquad (10)$$

and we will study the distribution of distorting vectors with one nonzero digit in the range [0, P).

Remember that the vectors with a (n - 1)-th zero digit have numerical values which are multiples of $P_i = \frac{P}{p_i}$, while the total system of distortion vectors (9) fills a range of numbers of length P_n.

Lemma 3. With condition (10), not more than one vector which is a multiple of P_i, i = 1, 2, ..., (n - 1) can fall in each range of length P_n.

Proof. To prove this statement, it will obviously suffice to show that

$$|k P_i - l P_j| > P_n$$

when

$$i \neq j, \quad i < n, \quad j < n, \quad 0 < k < p_i, \quad 0 < l < p_j.$$

We have

$$k P_i - l P_j = k \frac{P}{p_i} - l \frac{P}{p_j} = \frac{P}{p_i p_j} (k p_j - l p_i). \qquad (11)$$

It has been stipulated that $p_n > p_i p_j$; therefore

$$\frac{P}{p_i p_j} > \frac{P}{p_n} = P_n.$$

Furthermore, since the numbers k and l satisfy the conditions $0 < k < p_i$, $0 < l < p_j$, and p_i and p_j are coprime when $i \neq j$, we obtain

$$k p_j - l p_i \neq 0.$$

Whence it also follows from equation (11) that

$$|k P_i - l P_j| > P_n |k p_j - l p_i| > P_n$$

which had to be proven.

Theorem 4. We will assume that the error-free transmission of the numbers of the n-th digit is provided. When there is one redundant digit for base p_n which satisfies condition (10), the single error in the message received can always be corrected.

Proof. For the proof it suffices to note that according to lemma 3, only one number which is a multiple of P_i ($i \neq n$) can exist among the numbers of the whole system of distorting vectors (9).

Lemma 4. We will assume that condition (10) is satisfied.

The complete system of distorting vectors (9) contains two different vectors with a $(n-1)$-th zero digit only when the value of $\overline{A}$ for which it is comprised belongs to one of the ranges of the type

$$[k_i P_i;\ k_i P_i + P_n), \qquad (12)$$
$$i = 1, 2, \ldots, (n-1),\ k_i = 1, 2, \ldots, p_i - 1.$$

Proof. First we will show that system of vectors (9) contains two vectors with a $(n-1)$-th zero digit. According to lemma 2, one vector which is a multiple of P_n is always present in system of vectors (9). It also contains a vector which is a multiple of P_i ($i \neq n$), since the system of vectors (9) which fills the interval of length P_n has the point A, which lies in interval (12) (also length P_n) for its right end, while its left end is the vector $k_i P_i$.

We will demonstrate the opposite. If the system of distortion vectors (9) contains a vector of the form $k_i P_i$ ($i \neq n$) instead of a vector which is a multiple of P_n, the value of $\overline{A} = A_0$ lies to the right of the point $k_i P_i$ at a distance which does not exceed P_n.

Therefore, $\bar{A}$ lies in the range of the form (12).

We will use D to designate the set of points belonging to all the ranges (12).

Theorem 5. We will assume that condition (10) is satisfied, and that only a single error is possible in the received message $\bar{A}$. Then, if $\bar{A}$ does not belong to set D and $\bar{A} \geqslant P_n$, the number of the n-th digit is erroneous.

Proof. The proof follows from lemmas 4 and 2.

This theorem is a generalization of theorem 3 with the additional condition (10).

We will study the possibility of detecting double errors for a code with one redundant digit p_n.

Remember that we consider a double error to be the simultaneous distortion of two different numbers of a representation by a value which is limited to the corresponding base. This means that if $A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ is the transmitted message and $\bar{A} = (\bar{\alpha}_1, \ldots, \bar{\alpha}_n)$ is the received message with a double error, then

$$\overline{A} = A + \Delta_{ij},$$

where

$$\Delta_{ij} = (0, 0, \ldots, 0, \delta_i, 0, \ldots, 0, \delta_j, 0, \ldots, 0)$$
$$(\delta_i \neq 0, \ \delta_j \neq 0)$$

is the distorting vector.

Lemma 5. Suppose that (n - 2) digits have been fixed in the representation

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$$

and two digits, e.g., α_i and α_j $(i < j)$, assume all possible values $(0 \leqq \alpha_i < p_i, \ 0 \leqq \alpha_j < p_j)$. Then the numerical values of the representations obtained are distributed in $p_i p_j$ nonintersecting intervals

$$[k P_{ij}; \ (k + 1) P_{ij}),$$

where

$$P_{ij} = \frac{P}{p_i p_j}, \tag{13}$$
$$k = 0, 1, 2, \ldots, (p_i p_j - 1).$$

Proof. First of all, it is obvious that the $p_i p_j$ representations defined in the conditions are possible in all. We will show that none

of the intervals (13) can receive two different such representations. Indeed, let

$$A = (\alpha_1, \ldots, \alpha_{i-1}, \alpha_i, \alpha_{i+1}, \ldots, \alpha_{j-1}, \alpha_j, \alpha_{j+1}, \ldots, \alpha_n)$$

and

$$\overline{A} = (\alpha_1, \ldots, \alpha_{i-1}, \overline{\alpha_i}, \alpha_{i+1}, \ldots, \alpha_{j-1}, \overline{\alpha_j}, \alpha_{j+1}, \ldots, \alpha_n)$$

whereupon at least one of the differences $(\delta_i = \overline{\alpha_i} - \alpha_i, \delta_j = \overline{\alpha_j} - \alpha_j)$ is nonzero.

We will consider the difference

$$\overline{A} - A = (0, \ldots, 0, \delta_i, 0, \ldots, 0, \delta_j, 0, \ldots, 0)$$

It is obvious that the number $\overline{A} - A \neq 0$ can be divided by P_{ij}, i.e., $|\overline{A} - A| = sP_{ij}$, where $0 < s < p_i p_j$. This means that the distance between the numbers $\overline{A}$ and A is not smaller than the length of each of the intervals (13). Thus, all $p_i p_j$ numbers fall in different intervals (13), which had to be proven.

We can obtain a theorem analogous to theorem 4 from this lemma.

Theorem 6. We will assume that the error-free transmission of the numbers of the n-th digit has been provided. Then, when there one redundant digit with respect to base p_n which satisfies condition (10), the double error in the message received can always be

detected.

Proof. The proof follows from lemma 5, since because of condition (10), we have

$$P_{ij} = \frac{P}{p_i p_j} > \frac{P}{p_n} = P_n \qquad \text{and } \overline{A} > P_n,$$

and this is equivalent to a confirmation.

Now we will compute the number of detectable double errors in the presence of one redundant digit for the largest of the bases p_n.

We will consider the number of digits i and j to be fixed, with $i < j$.

The set of numbers $\overline{A}$ corresponding to all possible values of the distortion vectors A_{ij} obviously consists of $(p_i - 1)(p_j - 1)$ elements. According to lemma 5, all of these numbers fall in different ranges of length P_{ij}.

The erroneous representations with double errors which fall in the working range $[0, P_n)$ are undetectable.

We will calculate their number. There are

$$s_{ij} = \left[\frac{P_n}{P_{ij}}\right] = \left[\frac{p_i p_j}{p_n}\right]$$

nonintersecting ranges of length P_{ij} in range [0, P). Here, if j = n, $s_{in} = p_i$, and if $j \neq n$, $s_{ij} < p_i$ and s_{ij} nonintersecting ranges of length P_{ij} do not completely cover the interval $[0, P_n)$. Whence it follows that the numerical values of the transmitted messages A ($A < P_n$) belong to one of the intervals

$$[(k-1)P_{ij};\ kP_{ij}),\quad k = 1, 2, \ldots, s_{ij},$$

or

$$[s_{ij}P_{ij};\ P_n).$$

Now it is already clear that there are not less than $s_{ij} - 1$ and not more than s_{ij} distortion vectors Δ_{ij}, leaving the number $\tilde{A} = A + \Delta_{ij}$ in the range $[0, P_n)$. For example, there will be s_{ij} if $j \neq n$ and A is located in the interval $[s_{ij}P_{ij}, P_n)$, while there will be $s_{ij} - 1$ if

$$A = lP_{ij}\ (0 \leqq l \leqq s_{ij}).$$

Thus, the number of detectable double errors will not be less than

$$(p_i - 1)(p_j - 1) - s_{ij} = (p_i - 1)(p_j - 1) - \left[\frac{p_i p_j}{p_n}\right].$$

This means that out of the total number of possible double

errors

$$\sum_{1 \leq i < j \leq n} (p_i - 1)(p_j - 1),$$

not less than

$$\sum_{1 \leq i < j \leq n} \left((p_i - 1)(p_j - 1) - \left[\frac{p_i p_j}{p_n} \right] \right) \geqq$$

$$\geqq \sum_{1 \leq i < j \leq n} \left((p_i - 1)(p_j - 1) - \frac{p_i p_j}{p_n} \right).$$

will be detectable.

We will reduce this expression to a form more convenient for calculation:

$$\sum_{1 \leq i < j \leq n} (p_i - 1)(p_j - 1) - \frac{1}{p_n} \sum_{1 \leq i < j \leq n} p_i p_j =$$

$$= \left(1 - \frac{1}{p_n} \right) \sum_{1 \leq i < j \leq n} p_i p_j - \sum_{1 \leq i < j \leq n} (p_i + p_j) + \binom{n}{2} =$$

$$= \left(1 - \frac{1}{p_n} \right) \sum_{i=1}^{n-1} p_i \sum_{j=i+1}^{n} p_j + \binom{n}{2} - (n-1) \sum_{i=1}^{n} p_i. \quad (15)$$

Footnote: [1]Here $\binom{n}{k}$ is the binomial coefficient. End footnote

The last term was obtained here as a result of the following transformations:

$$\sum_{1\leq i<j\leq n}(p_i+p_j)=\sum_{j=2}^{n}\sum_{i=1}^{j-1}(p_i+p_j)=$$

$$=\sum_{j=2}^{n}p_j\sum_{i=1}^{j-1}1+\sum_{i=1}^{n-1}(n-i)\,p_i=(n-1)\sum_{i=1}^{n}p_i.$$

It suffices to find the percentage ratio of numbers (15) and (14) in order to obtain an idea of the proportion of detectable double errors.

We computed this ratio with the assumption that the codes in question are used in a computer operating in the SOK with an upper boundary of the numerical range of $10^{11}-10^{12}$. The simple numbers

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37,

were approximately used as the bases, where we considered the base 37 to be redundant. The calculations showed that with this coding, 96.9o/o of the double errors can be detected. This figure differs insignificantly from $\frac{p_n-1}{p_n}$ - which serves as the measure of the detection capacities of the code in question. In this case, we will have

$$\frac{37-1}{37}=\frac{36}{37}=0,9729, \qquad \text{i.e., } 97.29\text{o/o}.$$

Now we will consider the corrective capacities of codes with two redundant bases with respect to the largest digits p_{n-1} and p_n.

First, we will show that a code with two redundant digits provides the correction of all single errors.

Theorem 4. Let $A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ be the transmitted message or the precise result of the execution of arithmetic operations in the SOK, and $\overline{A} = \overline{\alpha}_1, \overline{\alpha}_2, \ldots, \overline{\alpha}_n)$ - the received message or the result obtained from the calculations in the SOK. Then a single error can always be corrected if the numerical values A of the transmitted messages or the precise results of the calculations in the SOK satisfy the inequalities

$$0 \leqq A < p_1 \cdot p_2 \cdots p_{n-1}.$$

Proof. We will consider all possible (n - 1)-digit representations A_i, comprised from the message received by subtracting one number (there will be n of them). If there are no errors in the message received, for all values of i = 1, 2, ..., n there will be

$$\overline{A_i} = A_i = A < p_1 \cdot p_2, \ldots, p_{n-1}.$$

If the message received contains a single error, then all the (n - 1)-digit representations of $\overline{A}_i$ except one will contain one

erroneous number; therefcre (thecrem 1), the inequality

$$\overline{A_i} > p_1 \cdot p_2 \ldots p_{n-2}$$

will be satisfied for the (n - 1)-th value of i, and for only one value of i, e.g., for i = k, will

$$\overline{A_k} < p_1 \cdot p_2 \ldots p_{n-2}.$$

The latter corresponds tc the case when the errcneous number has been subtracted; but then, since $A < p_1 \cdot p_2, \ldots, p_{n-2}$, we obtain

$$\overline{A_k} = A_k = A.$$

Now we can already find the true value of the number α_k. In crder to dc this, it suffices tc find the remainder from the division of A_k by p_k.

The calculation of the values of the (n - 1)-digit representations of $\overline{A}_i$ is not a complex problem. If the value of $\overline{A}$ has already been calculated, and we must begin with this, then

$$\overline{A_i} = \overline{A} - r p_i, \qquad \text{where} \qquad r = \left[\frac{\overline{A}}{p_i}\right].$$

Thus, in order to calculate the values of A, it suffices to subtract the number p_i from $\overline{A}$ until a number lying in the range $[0, p_i)$ is obtained.

The presence of two redundant digits is also sufficient for detecting any double error. Actually, in the presence of a double error, at least one of the $(n-1)$-digit representations will contain a single error and, therefore, the numerical value of $\bar{A}_i$ corresponding to it will be outside the range $[0; p_1 \cdot p_2 \ldots p_{n-2}]$.

Again we will assume that a binary channel is used for data transmission and we will estimate the number of redundant binary symbols for a code with two redundant digits, i.e., for the code of corrective single or detectable double errors.

Remember that in the codes in question, a single error means the possibility of distortion of not only one binary symbol, but also distortion of the number α_i by a value $\leqslant p_i - 1$, which can correspond to the distortion $[\log_2 p_i]$ of the binary symbols.

For the codes in question, the number of redundant binary symbols is equal to

$$r = [\log_2 p_{n-1}] + [\log_2 p_n] + 2.$$

In order to estimate this number as $n \to \infty$, we will use the asymptotic formula of analytical number theory

$$p_n \sim n \ln n, \qquad \text{i.e.,} \qquad \lim_{n\to\infty} \frac{p_n}{n \ln n} = 1.$$

Therefore,

$$\log_2 p_n \sim \log_2 n + \log_2 \ln n.$$

Thus, as $n \to \infty$, we obtain

$$r \sim \log_2 p_{n-1} p_n \sim \log_2 n (n-1) + \log_2 (\ln n \ln (n-1))$$

or

$$r \sim 2 \log_2 n.$$

I will take this opportunity to thank to I. Ya. Akushkiy for his constant attention to my work and his helpful comments.

Bibliography

1. И. Я. Акушский. Машинная арифметика в системе остаточных классов. Труды четвертого всесоюзного математического съезда, Л., 1961.

2. A. Svoboda, Rational Numerical System of Residual Classes, Stroje Na Zpracovani Informaci, Sbornik V, 1957.

3. D. Brown, IRE Trans., 1960, EC—9.