Series: **Computer Engineering**

Solomon I. Khmelnik

Positional codes of complex numbers and vectors

Abstract

A little-known theory of positional coding of complex numbers and vectors is described; this theory may be used for the development of specialized processors. The theory is supplemented by numerous examples.

Contents

Introduction

To improve performance of complex number processing in computer systems several proposals have been made for a *single-component* representation. The essence of these proposals is to present complex numbers in such a positional number system, which allows for complex operations to be implemented on a single data element without a need for separate processing of real and imaginary components.

S. Khmelnik [2, 5, 9] was among the first to analyze and propose various positional codes for presentation and processing of complex numbers. He proposed and analyzed several positional coding systems, including those with $j\sqrt{2}$ radix and (-1+j) radix. Other early researchers in this field include D. Knuth [1] who suggested to use an imaginary radix $j\sqrt{2}$ for positional complex codes and W. Penney [4] who afterward also suggested to use a complex radix (-1+j). In his works [2,5,6,7,9,14,15] S. Khmelnik suggested techniques for coding and decoding complex numbers and apparatus for arithmetic and mathematic processing of complex numbers, and then in [3,8,14-21] – implementation of these operations in digital hardware. Last works by S. Khmelnik [14-21] are being implemented in specialized ALUs for complex number processing.

Later several authors [11-13] suggested techniques for design of complex number multipliers. They used redundant codes for complex number representation, in order to achieve more regularity in ASIC hardware. However these coding systems were never applied to implementation of other arithmetic operations, such as division.

This article only summarizes the information about various positional codes for complex numbers and vectors.

1. About the method of positional coding

In this section we shall discuss positional codes of many-dimensional vectors Z, based on their expansion in a series

$$Z = \sum_m r_m f(\rho, \ m), \qquad (1)$$

where

- m - number of position
- ρ - radix of coding – a number or a vector,
- $f(\rho, \ m)$ - base function of the number and the radix,
- r – the expansion's position, number or vector, assuming values from a bounded set $A_R = \{a_0, a_1, a_2, ..., a_j, ..., a_{R-1}\}$, containing R various values a_j.

The positional code of a vector Z, respective to this expansion, has the form

$$K(Z) = ... \sigma_m ...,$$

where σ_m - the digit that denotes the value r_m. Formula (1) includes operations of addition and multiplication. For existence of algorithms for operations with such expansions (or, which is the same, with positional codes) the operations of addition and multiplication should be associative and commutative, and also should obey the distributive law. Hence, for positional coding of some set of objects to be possible, this set should be a ring. The set of real numbers and the set of many-dimensional vectors, with determined operations of addition and multiplication by number, satisfies this requirement. For real numbers the positional systems are known. For the above indicated set of vectors a positional system with the real basis will be developed below.

The set of complex numbers is a ring, and for it positional number systems in real and complex radix will be also constructed.

For development of a positional number system for many-dimensional vectors in a vector radix, operation of vectors multiplication subject to the above named laws should be determined. In other words, an algebra in many-dimensional vector space should be determined. This will be accomplished in the following.

First we shall consider two ways of vectors coding; then we shall proceed to a more general and rigorous description of positional coding method.

2. Two ways of complex numbers codes synthesis

Positional codes of many-dimensional vectors may be constructed as a certain composition of codes of real numbers to negative radix. Here and further j is imaginary unit.

Let X_α and X_β - be real numbers, defined as binary expansions in the radix $\rho = -2$, that is

$$X_\alpha = \sum_{(m)} \alpha_m \rho^m, X_\beta = \sum_{(m)} \beta_m \rho^m .$$

The codes corresponding to these expansions are

$$K(X_\alpha) = \alpha_m, \quad K(X_\beta) = \beta_m .$$

There are two ways of joining these two codes into a code of complex number. According to the **first** of them a pair of positions α_m and β_m are notated by one digit σ_m. Thus a code

$$K(Z) = \ldots\sigma_m\ldots$$

of complex number Z = $X_\alpha + jX_\beta$ in the radix ρ=-2 with positions that assume one of the four values. The resulting code will be as follows:

$$\sigma_m \in \{0, 1, j, 1+j\}.$$

Let us consider now a complex function of a real integer argument m:

$$\rho_2 = \begin{cases} (-2)^{m/2} \text{ if } m-\text{even} \\ j(-2)^{m-1/2} \text{ if } m-\text{odd} \end{cases} \quad (2)$$

The considered code of this complex number to the radix of (-2) with complex values of positions may be regarded as a code of complex number on the radix (ρ_2) with binary positions. This code corresponds to the expansion of a complex number: $Z = \sum_m \left(\sigma_m \rho_2^m\right)$, where binary positions are $\sigma_m = \begin{cases} \alpha_m \text{ if } m-\text{even} \\ j \cdot \beta_m \text{ if } m-\text{odd} \end{cases}$. For illustration we shall show here the codes of some characteristic numbers in this system:

K(2) = 10100, *K*(-2) = 100, *K*(-1) = 101, *K*(*j*) = 10, *K*(-*j*) = 1010, *K*(2*j*) = 101000.

The **second** way consists in construction of a sequence of alternate positions α_m and β_m

$$\ldots\beta_{m+1}\alpha_{m+1}\beta_m\alpha_m\beta_{m-1}\alpha_{m-1}\ldots$$

Denoting $\alpha_m = \sigma_{2m}$, $\beta_m = \sigma_{2m+1}$, we shall rewrite this sequence in another form:

$$\ldots\sigma_{k+3}\sigma_{k+2}\sigma_{k+1}\sigma_k\sigma_{k-1}\sigma_{k-2}$$

where $k=2m$. This sequence is the binary code

$$K(Z) = \ldots\sigma_m\ldots$$

of a certain complex number Z. It is possible to show (and it will be done below), that the code obtained in such a way is a binary code in the radix $\rho = \pm j\sqrt{2}$, and the coded number is $Z = X_\alpha + \rho \cdot X_\beta$.

Thus, certain compositions of binary codes of real numbers in the radix ρ = - 2 are the codes of complex numbers. When performing the algebraic addition of complex numbers such codes may be considered simply as a set of real numbers' codes and so we may perform the same operation independently with each pair of real numbers. Operations of multiplication for such codes and division for the codes of the second type are executable . The operations of multiplication and division consist, as usual, of cycles "shift - addition".

3. Method of coding for the points of many-dimensional space

A method of coding for points of many-dimensional Euclidian space should establish a certain correspondence between these points and the codes from a certain set. This correspondence, generally speaking, is not necessarily a one-to-one correspondence. But in order to make the decoding unique, each code should correspond to an unique point of the coded space. At the same time even a bounded part of the space contains an uncountable set of points. Hence, the set of corresponding codes is also uncountable, and among them there are bound to be codes with infinite number of positions (**infinite codes**). However in practice of calculations only **final codes** are used, and the set of final codes is bounded.

To preserve a correspondence between the codes and the points of space, it would appear reasonable to divide the bounded coded domain G into a bounded set of sub-domains δ of a fixed size and configuration, so that each point of the domain G will belong to one of sub-domains δ . Then a one-to-one correspondence between set of final codes and set of the areas δ will be attainable.

Such way of coding the points of many-dimensional space is approximate. Indeed, a unique code K_i corresponds to all points $Z_j \in \delta_i$. However when decoding the code K_i we get an unique point Z_i . Denote the radius-vector of a point Z by symbol $\overline{Z}$. The difference $\Delta Z_j = |\overline{Z}_j - \overline{Z}_i|$ defines the absolute error of point Z_j coding.

For illustration let us consider fig. 1, where we see domain Z_j of the two-dimensional space, divided into sub-domains δ .

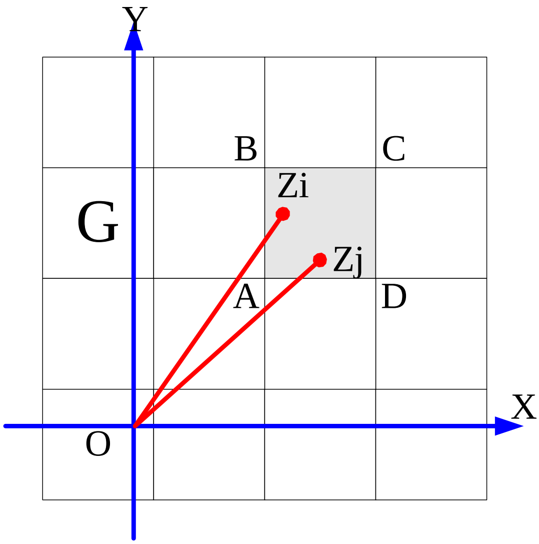

Fig. 1. Coding of two-dimensional region

On this figure one of the sub-domains δ_i = ABCD is marked out, where its lower (AD) and right (CD) borders also belong to δ_i. In δ_i we have taken a basic point Z_i and another point $Z_j \in \delta_i$. The length of segment ΔZ_j characterizes the absolute error of the point Z_j coding.

So, the stated principle of coding for points of many-dimensional space may be formulated as following:

- bounded domain G of the coded space is divided into a bounded set of equal sub-domains δ_i $(i=1, 2, \ldots, N)$, where

  $$G = \bigcup \delta_i \text{ and } \delta_i \bigcap \delta_j = \varnothing \text{ at } i \neq j;$$

- a set of final codes K_i $(i=1, 2, \ldots, N)$ is defined;
- between the sub-domains and the codes a one-to-one correspondence is established.

If these conditions are observed, it means that the **system of coding** of the domain G of many-dimensional space satisfies the **principle of coding** and the domain G is coded with **discreteness** δ. The following two lemmas are obvious.

Lemma 1. The system of coding of domain G satisfies the principle of coding if $V=NU$, and vice versa, where

U – is the δ -domain's volume,

V – is the G –domain's volume,

N - the power of final codes set.

Lemma 2. The system of coding for G - domain, satisfying the principle of coding, is called **complete** (that is, any point corresponds to a final code), **nonredundant** (that is, each point corresponds to one final code) and **approximate** (that is, a subset of points - vectors, for which module of difference does not exceed a certain value, corresponds to one final code).

Consider a set of *n*-digit codes of the form

$$K = \alpha_{n-1}...\alpha_k...\alpha_1\alpha_0, \quad (3)$$

where α_k is a digit that assumes one of the values R_k, $R_k > 1$ and is integer.

Lemma 3. If the system of coding satisfies the principle of coding, then with increase of final codes digit capacity and on retention of coding discreteness, the coding volume of the coded domain is increased in the same way as the power of final codes set, and vice versa.

Proof. The power of final codes set is

$$N_n = \prod_{k=1}^{n} R_k . \quad (4)$$

Let this set of codes satisfy the principle of coding and encode the domain G_n with discreteness δ . According to lemma 1, the number of sub-domains δ , contained in G_n, is also equal to N_n, and the G_n domain has volume

$$V_n = N_n U . \quad (5)$$

Now we shall increase the codes digit capacity by 1, that is, we add a digit α_n, which assumes one of the values R_n. It is obvious that

$$N_{n+1} = R_n N_n . \quad (6)$$

Let the new set of codes also satisfy the principle of coding and encode the domain G_{n+1} with the same discreteness δ . The number of sub-domains δ , contained in domain G_{n+1}, is equal to N_{n+1}, that is, the domain G_{n+1} has volume

$$V_{n+1} = N_{n+1} U . \quad (7)$$

Combining three last formulas, we find, that

$$V_{n+1} = R_{n+1} V_n , \quad (8)$$

that is, the direct part of the lemma is proven.

By the conditions of the inverse part of lemma, the formulas (5), (6), (8) are valid. From them follows (7), whence according to the lemma 1 we come up with the proof of the inverse part of the lemma.

We shall consider now a positional system of coding. In this system each positional code

$$K(Z) = \alpha_n \ldots \alpha_k \ldots \alpha_m$$

corresponds to a point Z of a coded many-dimensional space, which has an expansion of the following form:

$$Z = \sum_{k=m}^{n} \alpha_k \rho^{k}, \qquad (9)$$

where

ρ - is the radix of coding,

k – the position number,

α_k - the k-th position of a code (digit or quantitative equivalent, corresponding to it in the expansion), which assumes one of the values R_k.

Notice that ρ and α_k are also points of the coded many-dimensional space. The positional code is called **infinite** if $m = -\infty$, and **final** if m is limited. Number n called the **length** of positional code. If $R_k = R$, the expansion and the code are called R-expansion and R-code. So, we shall consider the value a, assuming values from the set

$$A_R = \{a_0, a_1, a_2, \ldots, a_j, \ldots, a_{R-1}\}, \qquad (10)$$

containing R different values a_j. In practice of positional coding it is essential that R is limited and is not larger than a few units.

We shall denote the positional code of a point Z on the radix ρ and write it also as follows

$$< Z >_\rho = \alpha_n \ldots \alpha_k \ldots \alpha_1 \alpha_0 , \alpha_{-1} \alpha_{-2} \ldots \alpha_m , \qquad (11)$$

placing the point between the digits zero and (-1) (index - base will not be indicated, if the base'd value is clear from the context). A vector (point) Z, in whose code $m \geq 0$, will be called ρ - **whole**. Accordingly the vectors Z that are ρ - **fractional** (**proper** and **improper**) are defined. In particular,

$$< \rho >_\rho = 10 . \qquad (12)$$

The aggregate of $< \rho, \ A_R >$ of the radix of coding ρ and the set A_R shall be called a **system of positional coding**. We shall say that a point of many-dimensional Euclidian space is **representable** in the given system of positional coding, if there is a corresponding expansion of the form (9) and a positional code of the form (11), where the digits assume values from the set (10).

We have to construct such positional systems of coding, in which any point of the given space will be representable, subject to conditions of completeness, nonredundance and approximateness, according to lemma 2.

The purpose of positional systems construction is to simplify the execution of arithmetic operations with points (vectors) of the many-dimensional space. On the other hand, the existence of positional codes, based on expansion (9), is possible only for such space, where operations of summation of vectors and multiplication of a vector by the radix ρ (which can also be a vector) are determined.

In one- and two-dimensional spaces multiplication by the radix ρ (multiplication by real or complex number) corresponds to an increase of a module vector-multiplicand by the factor $|\rho|$, that is,

$$\text{if } Z_2 = Z_1\rho, \text{ then } |Z_2| = |Z_1||\rho|. \qquad (13)$$

It should once again noted, that multiplication $Z_1\rho$ is equivalent to a shift of the code $<Z_1>$ by one position to the left *in any space*. We shall require the condition (13) to be valid also *for any coded space* and we shall prove a certain condition of a positional system's existence, using these two facts.

<u>**Theorem 1**</u>. For any point of an h-dimensional Euclidian space in which condition (13) is satisfied, the necessary and sufficient condition of its representability in the given system of positional coding is the condition:

$$|\rho|^h = R. \qquad (14)$$

Proof. Each code $<Z_2>_\rho$ of length $(n+h)$ with $m = -\infty$ may be received by shift by h positions to the left of a certain code $<Z_1>_\rho$ of length n. But according to (12) such shift is equivalent to multiplication by the radix, that is, $Z_2 = Z_1\rho^h$. Thus from formula (13) follows, that $|Z_2| = |Z_1||\rho|^h$. Hence, the linear sizes of the coded domain are increased by $|\rho|^h$ (besides the coded area, generally speaking, is turned relative to its previous position). Thus, the volumes of the areas G_n and G_{n+h} are related by the following equation:

$$V_{n+h} = |\rho|^h V_n . \quad (15)$$

Obviously, the restriction m does not change volume of the coded domain. There only appears discreteness of coding $\delta = G_{m-1}$. Taking into account (14), from (15) we obtain

$$V_{n+h} = RV_n . \quad (16)$$

Comparing (16) and (8), from lemma 3 we find that the system of positional coding at $m < -\infty$ satisfies the principle of coding, that is, owing to lemma 2, it is total, nonredundant and approximate. The theorem is proven.

4. Arithmetic systems of coding

Among positional systems of coding, such systems for which simple algorithms of addition and multiplication are applicable – are subjects of particular interest. Such indeed are the systems which we shall consider below, but first we must define them more strictly.

Definition 1. System $<\rho \quad , \quad A_R>$ of positional coding is called **arithmetic**, if following conditions are fulfilled

- number (-1) is ρ - whole,
- the sum and the product of any pairs of vectors, belonging to to set A_R, are ρ - whole.

Note, that the condition (13) may be valid also for a non-arithmetic system.

Lemma 4. If in an arithmetic positional system the vectors Z_1 and Z_2 are representable, then the vectors $-Z_1$, $-Z_2$, $Z_1 + Z_2$, $Z_1\ Z_2$ are also representable in this system.

The lemma’s validity follows from the fact that, as it will be shown below, for arithmetic positional systems there exist algorithms of arithmetic operations.

Definition 2. Positional system $<\rho, \ A_R>$ is called **normal**, if $A_R = B_R$, where $A_R = B_R$, где $B_R = \{0,1,2,...,R-1\}$.

Lemma 5. A normal number system, in which

$$R = \sum_{k=1}^{n} \alpha_k \rho^k , \quad (17)$$

$$-R=\sum_{k=1}^{w}\beta_k\rho^k, \qquad (18)$$

that is, the codes of numbers R and $-R$ are ρ-whole and have zero value of the zero digit, is an arithmetic system.

Proof. For any number from set B_R $0\le a_j\le(R-1)$. Hence, for numbers from this set the equations $-a_j=a_k-R$ and $a_j+a_k=a_m+R$, if $a_j+a_k\ge R$ are valid. Taking into account the conditions of the lemma, we conclude, that the numbers $(-a_j)$ and (a_j+a_k) are ρ-whole. Obviously, the product a_ja_k can be presented as a sum of numbers from the set B_R. By induction, owing to the existence of addition algorithm, we conclude, that such sum is also ρ-whole. Thus, the conditions of definition 1 are fulfilled. Hence, the considered system is an arithmetic one.

Lemma 6. A normal number system, in which the number R has an expansion of the form (17), and

$$R=\sum_{k=1}^{m}\alpha_k, \qquad (19)$$

is an arithmetic system.

Proof. As follows from (17) and (19), in the lemma systems, in which

$$R=\sum_{k=1}^{n}\alpha_k\rho^k=\sum_{k=1}^{n}\alpha_k.$$

are considered.

Let us consider the following algorithm:

α_3	α_2	α_1	0				carries
	α_3	α_2	α_1	0			carries
		α_3	α_2	α_1	0		carries
			α_3	α_2	α_1	$0=\langle R\rangle_\rho$	addend 1
				β_2	β_1	$0=\langle X\rangle_\rho$	addend 2
0	**0**	**0**	**0**	**0**	**0**	**0**	sum

Here the code of number R is summed the with code of some number X, whose digits are formed so that

$$\alpha_1+\beta_1=R \text{ и } \alpha_1+\alpha_2+\beta_2=R.$$

Thus, and owing to (19) the addition of digits for each column will give us number R, which forms the carry and zero digit of the sum. As a result the infinite carries and the zero sum will be formed. Hence, X=-R. Obviously, such algorithm of number -R code formation is executable for any R corresponding to the expansion (17) or, which is the same, for any code of number R of a form

$$< R >_{\rho} = \alpha_m ... \alpha_2 \alpha_1 0 .$$

The result of this algorithm is a code of number -R:

$$< -R >_{\rho} = \beta_w ... \beta_2 \beta_1 0 .$$

This code corresponds to expansion (18). Thereby the lemma is proved.

Note that the expansions (17) and (18) can be considered as a system of two power equations with unknown ρ. Solving it, we may, generally speaking, define a certain system of coding. However this method does not always lead to good results, because the given system either is not solvable analytically, or is not compatible, or gives a solution not satisfying the condition of Theorem 1, or gives a real number as the solution.

Lemmas 4, 5, 6 will be used further in the search for normal positional systems of coding.

5. Codes of real numbers

For real numbers the dimension of the coded space is h=1. Hence, for positional codes of real numbers it is necessary to observe a condition

$$|\rho| = R .$$

Positional codes of real numbers in which $\rho = R$ and the positions assume values from set B_R are widely known and widespread. Among these are usual decimal (R =10) and binary (R =2) codes. However such codes can't portray negative real numbers, so we have to use some workarounds, in particular, inverse or additional codes, which may cause certain inconvenience.

At the same time there are two ways of developing positional codes fit to portray real - positive and negative numbers. First of them consists in giving positive and negative values from the set $D_R = \{-r_1, -r_1 + 1, ..., -1, 0, 1, ..., r_2 - 1, r_2\}$ to the positions $R = r_1 + r_2 + 1, r_1 \neq 0, r_2 \neq 0,$ leaving the radix equal to R (at $r_1 = 0$ the set D_R turns into set B_R). The other way is based on the use of negative radix $\rho = -R$. Thus the digits may assume values either from set B_R, or from set D_R. So the known results related to positional coding of real numbers, may be formulated as follows.

Theorem 2. Any real positive number may be represented in the systems

$$< R, B_R >, \quad < R, D_R >, \quad < -R, B_R >, \quad < -R, D_R > .$$

So, there are four systems of real numbers coding:

the system $< R, B_R >$, for example < 5, { 0, 1, 2, 3, 4 } >;

the system $< R, D_R >$, for example < 5, { -2, -1, 0, 1, 2 } >;

the system $< -R, B_R >$, for example < -5, { 0, 1, 2, 3, 4 } >;

the system $< -R, D_R >$, for example <-5, {-2, -1, 0, 1, 2 }>.

We shall give some examples of penta-codes of numbers in the above-named systems, denoting the values -1 and -2 as $\overline{1}$ and $\overline{2}$:

1. K(16)= +31,K(-13)= -23,
2. K(16)= $1\overline{2}1$,K(-13)= $\overline{1}22$,
3. K(16)= 121,K(-13)= 32,
4. K(16)= 121,K(-13)= $\overline{1}\overline{2}2$.

Here we must draw attention to the fact that in the first of these systems the codes have " + " and "-", which are absent in all the other systems, for in them the number's sign, as well as the module, are defined by the values of code's digits.

It is important to note, that among indicated systems there are only two systems of binary coding, namely the system with numbers { 0, 1 } and radixes "2" and "-2".

6. Codes of complex numbers

We shall begin with proving some existence theorems of normal arithmetic systems of coding with complex radix, denoting imaginary unit by j.

Theorem 3. Any complex number is representable in a normal system of coding in a complex radix ρ, and this system is arithmetic, if

$$|\rho| = \sqrt{R} \tag{20}$$

and conditions (17), (19) are fulfilled.

Proof. For complex numbers the dimension of the coded space is h=2, and at any ρ the condition (13) is fulfilled. From here and from (20) follows, that the conditions of theorem 1 are fulfilled. Hence, any complex number is representable in the given system of coding. Further, the conditions (17) and (19) are the conditions of lemma 6. Hence, the given system is arithmetic.

Theorem 3 enables to reduce the proof of theorems about representability of any complex number in a normal system of coding and arithmeticality of this system to the proof that condition (19) is satisfied and ρ is a complex root of the equation (17). This very method of proof we shall use in future.

Theorem 4. Any complex number is representable in a normal system of coding at complex radix

$$<\rho = \sqrt{2}e^{\pm j\pi/2};\ \ B_2> \text{ or } <\rho = -1 \pm j;\ \ \{0,\ 1\}>$$

and this system is arithmetic.

Proof. Let us assume, that $<2>_\rho = 1100$. This condition it is equivalent to an equation $\rho^3 + \rho^2 = 2$. Its decision coincides with the condition of the given theorem. Hence, condition (17) is satisfied. Obviously the condition (19) is also satisfied, because R=2. In view of theorem 3 the given the theorem is proven. For illustration let us write several characteristical numbers in a system at the radix $\rho = (j-1)$, denoting by $\overline{\rho}$ - number adjoined to number ρ:

$K(2) = 1100$, $K(-2) = 11100$, $K(-1) = 11101$, $K(j) = 11$, $K(-j) = 111$, $K(\overline{\rho}) = 110$.

Theorem 5. Any complex number is representable in a normal system of coding at a complex radix ρ and this system is arithmetic, if

$$\rho = \sqrt{R}e^{j\varphi},\quad \varphi = \pm\ arcCos\ (-\beta/2\sqrt{R}),\quad \beta < (R,\ 2\sqrt{R})_{\min}$$

and β - a whole positive number.

Proof. Let us assume that $<R>_\rho = 1\alpha_2\alpha_1 0$, where $1 + \alpha_2 + \alpha_1 = R,\quad \alpha_2 = \beta - 1$. This condition is equivalent to an equation $\rho^3 + (\beta - 1)\rho^2 + (R - \beta)\rho = R$. Its solution gives the value that was mentioned in the theorem's conditions. In view of theorem 3 the given theorem is proven.

For illustration let us record codes of some characteristic numbers in this system, denoting by $\overline{\rho}$ - number adjoined to number ρ:

$$K(R) = 1\ \ (\beta - 1)\ \ (R - \beta)\ \ 0,$$

$$K(-R) = 1\ \ \beta\ \ 0,$$

$$K(-1) = 1 \;\; \beta \;\; (R-1),$$
$$K(\overline{\rho}) = 1 \;\; (\beta - 1) \;\; (R - \beta),$$
$$K(-\overline{\rho}) = 1 \;\; \beta,$$
$$K(-\rho) = 1 \;\; \beta \;\; (R-1) \;\; 0,$$
$$K(\rho - \overline{\rho}) = 2 \;\; \beta,$$
$$K(\rho + \overline{\rho}) = 1 \;\; \beta \;\; (R - \beta).$$

As β can assume several values with constant R, there are several types of positional codes in systems of the considered type. As an example, possible codes of number R with various R and β are presented in Table 1.

Table 1. Codes of number R.

$R \setminus \beta$	**1**	**2**	**3**	**4**	**5**
2	1010				
3	1020	1110			
4	1030	1120	1210		
5	1040	1130	1220	1310	
6	1050	1140	1230	1320	
7	1060	1150	1240	1330	1420
8	1070	1160	1250	1340	1430
9	1080	1170	1260	1350	1440

By way of illustration we shall present the codes of several characteristic numbers in a system in the radix $\rho = \frac{1}{2}\left(-1 + j\sqrt{7}\right)$, denoting by $\overline{\rho}$ - number adjoined to number ρ:

$K(2)$=1010, $K(-2)$=110, $K(-1)$=111, $K(\overline{\rho})$=101, $K(-\rho)$=1110, $K(-\overline{\rho})$=11, $K\left(j\sqrt{7}\right)$=10101, $K\left(-j\sqrt{7}\right)$=1110011.

From the systems of theorem 5 it is possible to single out the groups with a fixed value of the base's argument, for example

$\varphi = \pm 2\pi/3$, if $\beta = \sqrt{R}$, that is, at R=4, 9, 16, 25, ...;

$\varphi = \pm 3\pi/4$, if $\beta = \sqrt{2R}$, that is, at R=8, 18, 32, 50,...;

$\varphi = \pm 5\pi / 6$, if $\beta = \sqrt{3R}$, that is, at R=12, 27, 48, 75,..;

Let us consider now a positional system of a more general form.

Theorem 6. Any complex number is representable in a system of coding $<\rho = 2e^{j\pi/3}, A_4>$, $A_4 = \{0, 1, ,e^{2j\pi/3}, e^{-2j\pi/3}\}$ and this system is arithmetic.

Proof. Notice, that $(-2)^k = l_k \rho^k$, where

$$l_k = \{1, e^{2j\pi/3}, e^{-2j\pi/3}\}$$

accordingly at $k = (3m, 3m+1, 3m+2)$, where m is integer. Obviously, $l_k \in A_4$. Hence, any power of the number "-2" may be represented in the indicated system of coding by one digit. In view of theorem 2 any real number X is representable as an expansion in the radix "-2". But each digit of such expansion, being a power of number "-2" or 0, can be replaced by the term of expansion in the indicated system of coding, that is, any real number is representable in this system of coding.

Table 2. One-digit multiplication

*	0	1	c	d
0	0	0	0	0
1	0	1	c	d
c	0	c	d	1
d	0	d	1	c

Any complex number Z can be presented as $Z = u_1 + u_2 e^{2j\pi/3} + u_3 e^{-2j\pi/3}$, where u_1, u_2, u_3 - certain real numbers. In this sum all the components are representable in the indicated number system, because the cofactors of real numbers u_1, u_2, u_3 belong to the set A_4. If this system is arithmetic, then the considered sum is representable in it, and so this is true for any complex number. It remains to show that the considered system is arithmetic. For that we shall arrange tables of pairwise multiplication, summation and the table of inverting (multiplication by "-1") for figures from set A_4 - see tables 2, 3, 4. For convenience these figures are denoted by symbols 0, 1, *c, d*. As may be seen from these tables, in the considered system of coding all conditions of definition 1 are fulfilled. Hence this system is arithmetic, as it was required to show.

Table 3. One-digit summation

+	0	1	c	d
0	0	1	c	d
1	1	dc0	1d	dc
c	c	1d	d10	c1
d	d	dc	c1	c10

Table 4. Inverting a digit

x	**0**	**1**	**c**	**d**
-x	**0**	**c1**	**dc**	**1d**

We must note, that in this system the codes of complex numbers of the forme e^{jk60° with integer k have a very simple form - see table 4a. Besides, the table 4b presents the codes of numbers 2^k and $(-2)^k$ with integer k.

Table 4a. Codes of numbers e^{jk60°.

φ	0	60	120	180	240	300
код	**00**	**1d**	**0c**	**c1**	**0d**	**dc**

Now we shall only state more accurately the results obtained in section 2.

Theorem 7. Any complex number Z is representable in a positional number system $<\rho=-R,\ A_{R^2}>$, where the set A_{R^h} consists of complex numbers $r_m=\alpha_m^1+j\alpha_m^2$, and the numbers $\alpha_m \in B_R$.

In particular, there exists a system $<-2, \{0,1,j,1+j\}>$.

Table 4b. Codes of numbers 2^k *and* $(-2)^k$.

k	$(-2)^k$	2^k
-4	0.000d	0.000d
-3	0.001	0.0c1
-2	0.0c	0.0c
-1	0.d	1.d
0	1	1
1	c0	dc0
2	d00	d00
3	1000	c1000
4	c000	c000

Theorem 8. Any complex number Z is representable in a normal positional system $< \pm j\sqrt{R}, B_R >$.

For example, there exists a system $< \pm j\sqrt{2},\ \{0,1\} >$. For illustration we shall write the codes of several characteristic numbers in the system $\rho = j\sqrt{2}$:

$K(2)=10100$, $K(-2)=100$, $K(-1)=101$, $K\left(j\sqrt{2}\right)=10$,
$K\left(-j\sqrt{2}\right)=1010$.

Table 5. Binary systems of coding.

Preferred number systems	ρ	**<2>**	**<-2>**	**<-1>**	**Theorem**	**Fig.**
System 1	ρ_2	10100	100	101	Formula (2)	1
System 2	$j\sqrt{2}$	10100	100	101	Theorem 8	2
System 3	$-1+j$	1100	11100	11101	Theorem 4	3
System 4	$\frac{1}{2}\left(-1+j\sqrt{7}\right)$	1010	110	111	Theorem 5	4
	-2	110	10	11	Theorem 2	
	2	10			Theorem 2	

Obviously, for the systems from theorems 7 and 8 the condition (14) is satisfied. The proof of these theorems is based on the reasonings of section 2.

For illustration and comparison let us present the binary codes of numbers in all the considered systems of coding, including systems of coding in real (positive and negative) and complex radix - see table 5.

Further we shall dwell in more detail on the four binary systems of complex numbers - see the column «Preferred number systems» in table 5. The following figures present the first 4 values of base function for the preferred number systems.

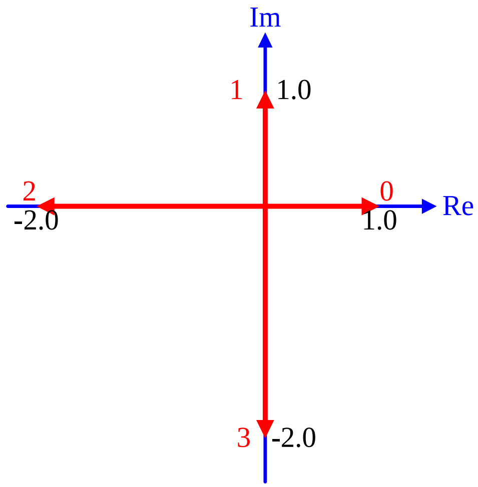

Fig. 1

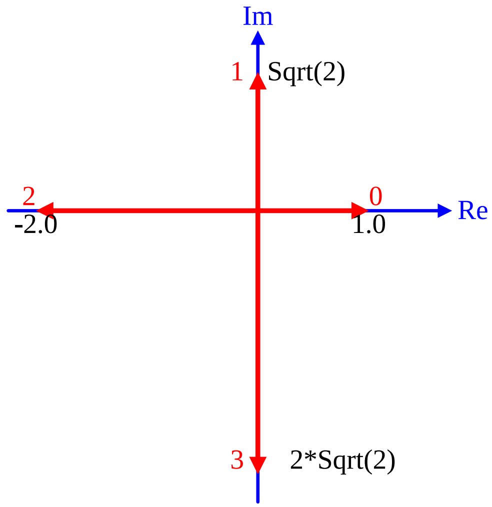

Fig. 2.

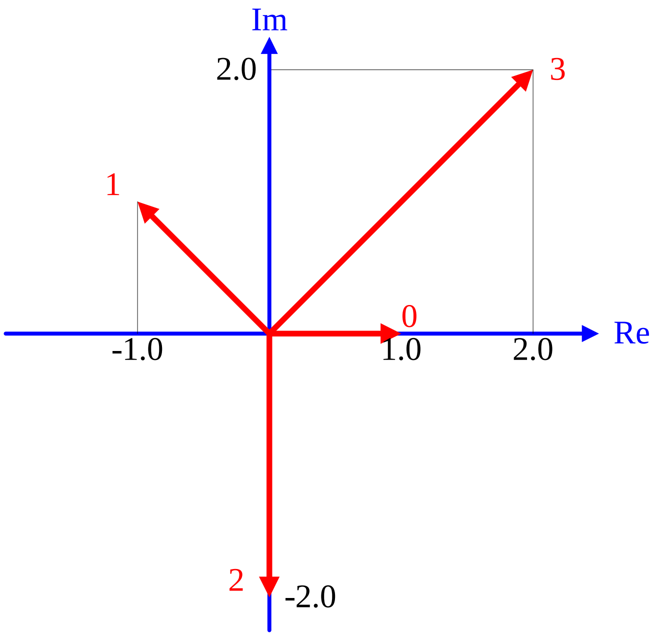

Fig. 3.

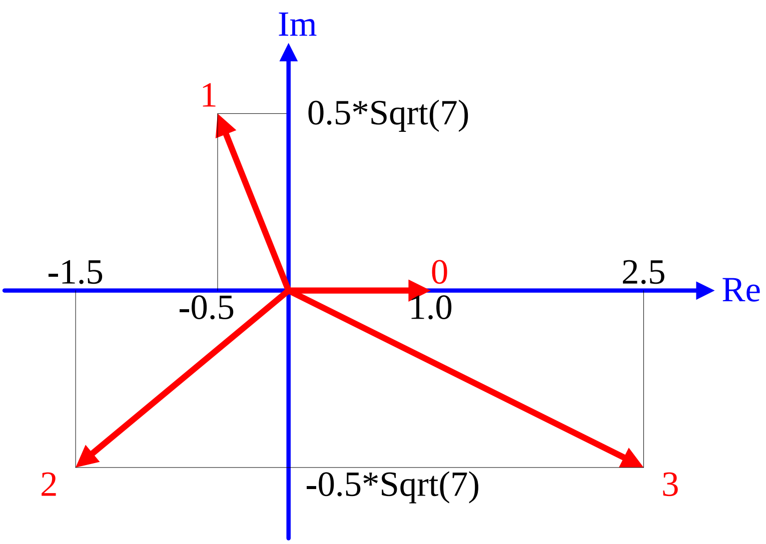

Fig. 4.

We shall also present a table of 6 four-valued codes of numbers ' 4 ' and '-4 ' in all systems of coding considered above (in this table '-1' it is designated by sign 'h').

Table 6. Four-valued coding system.

ρ	A_4	<4>	<-4>	Theorem
4	{0,1,2,3}	10		2
4	{-1,0,1,2}	10	h0	2
4	{-2,-1,0,1}	10	h0	2
-4	{0,1,2,3}	130	10	2
-4	{-1,0,1,2}	h0	10	2
-4	{-2,-1,0,1}	h0	10	2
$2e^{2j\pi/3}$	{0,1,2,3}	1120	120	5
$2e^{j\pi/3}$	{0,1,c,d}	d00	1d00	6
-2	{0,1,j,1+j}	100	1100	7
ρ_4	{0,1,2,3}	10300	100	8
$\pm 2j$	{0,1,2,3}	10300	100	8

7. Codes of many-dimensional vectors

7.1. Binary codes of vectors - method 1.

The stated method of complex numbers codes construction may be generalized and used for many-dimensional vectors coding. To do so we shall consider a set of real numbers $\{X_i\}$, each of which is described by a binary expansion in the radix ρ = -2, that is

$$X_i = \sum_{(m)} \alpha_m^i \rho^m \quad (i=1, 2,..., n).$$

To each such expansion there corresponds a code

$$K(X_i) = ...\alpha_m^i...$$

We shall consider now n-dimensional vector

$$Z = E_1 X_1 + E_2 X_2 + ... + E_n X_n, \tag{21}$$

where $\{E_i\}$ is the base of n-dimensional vector space. The set of codes $\{K(X_i)\}$ thus may be interpreted as an uniform code of the vector Z in the radix "-2". Each m-th digit of this code is represented by the set

$\{\alpha_m^i\}$ of binary digits. Denoting these sets by digits σ_m, we get the code of the vector

$$K(Z) = \ldots\sigma_m\ldots,$$

corresponding to the expansion (1), where the vector

$$r_m = E_1\alpha_m^1 + E_2\alpha_m^2 + \ldots + E_i\alpha_m^i + \ldots + E_n\alpha_m^n \qquad (22)$$

is represented by digit σ_m.

In particular, at $n = 2$ we shall get the codes of complex numbers in radix "-2," which have been considered higher. At $n=3$ we shall get the codes of three-dimensional vectors, where the digits take one of eight values:

$$r_m \in \{\,\mathbf{0, i, j, k, i{+}j, i{+}k, j{+}k, i{+}j{+}k}\,\}, \qquad (23)$$

where **i, j, k** are orts of Cartesian coordinate system. Similarly, for coding three-dimensional vectors, a function from real whole vector argument m may be introduced:

$$\vartheta_2 = \begin{Bmatrix} i(-2)^m \text{ if } m = 3k \\ j(-2)^{m-1} \text{ if } m = 3k+1 \\ k(-2)^{m-2} \text{ if } m = 3k+2 \end{Bmatrix}, \qquad (24)$$

The considered code of a three-dimensional vector in the radix ϑ_2) with vector digit values (23) may be considered a code of a three-dimensional vector in the radix ϑ_2 with binary digits. To this code there corresponds the vector's expansion $Z = \sum_m (\alpha_m \vartheta_2)$.

Similarly, for *n*-dimensional vectors coding a vector-function of real integer argument *m* may be introduced:

$$\vartheta_2^n = \begin{Bmatrix} i(-2)^m \text{ if } m = nk \\ j(-2)^{m-1} \text{ if } m = nk+1 \\ \ldots \\ k(-2)^{m-n+1} \text{ if } m = nk+n-1 \end{Bmatrix}, \qquad (25)$$

Obviously, $\rho_2 = \vartheta_2^2$, $\vartheta_2 = \vartheta_2^3$.

7.2. Binary codes of vectors - method 2.

We shall now build, as we have done earlier for complex numbers, a sequence of alternated binary digits α_m^i:

$$\cdots\alpha_{m+1}^{2}\alpha_{m+1}^{1}\alpha_{m}^{n}\alpha_{m}^{n-1}\cdots\alpha_{m}^{2}\alpha_{m}^{1}\alpha_{m-1}^{n}\alpha_{m-1}^{n-1}\cdots$$

In other notations this sequence is a binary code

$$K(Z)=\ldots\alpha_k\ldots\ldots$$

of a certain vector Z. The radix of coding is also a vector

$$\rho=E_2\sqrt[n]{2}, \qquad (26)$$

where E_2 - the second ort of the base $\{E_i\}$ of an n-dimensional vector space. The coded vector Z is defined in this case by the formula

$$Z=X_1+\rho X_2+\ldots+\rho^{i-1}X_i+\ldots+\rho^{n-1}X_n. \qquad (27)$$

7.3. Many-dimensional codes of vectors - method 2.

The positional codes of vectors (including complex numbers and multidimensional vectors) are built precisely similarly, based on incorporation of positional codes of numbers - projections of vectors at the radix $\rho=-R$, where R is integer. In this case, for example, instead of the function ρ_2 the function

$$\rho_R=\begin{Bmatrix}(-R)^{m/2} \text{ if } m-\text{even}\\ j(-R)^{m-1/2} \text{ if } m-\text{odd}\end{Bmatrix}, \qquad (28)$$

should be considered as the radix of complex numbers coding, and instead of function ϑ_2 the function

$$\vartheta_R=\begin{Bmatrix}i(-R)^{m/3} \text{ if } m=3k\\ j(-R)^{m-1/3} \text{ if } m=3k+1\\ k(-R)^{m-2/3} \text{ if } m=3k+2\end{Bmatrix}, \qquad (29)$$

should be considered as the radix of complex numbers coding, and, generally speaking, instead of function ϑ_2^n the function *n*-dimensional vectors coding we should consider the following function

$$\vartheta_R^n = \begin{cases} i(-R)^m \text{ if } m = nk \\ j(-R)^{m-1} \text{ if } m = nk+1 \\ \ldots \\ k(-R)^{m-n+1} \text{ if } m = nk+n-1 \end{cases}. \quad (30)$$

should be considered as the base of n-dimensional vectors coding. Thus, the following theorems are true.

Theorem 7a. If in n-dimensional Euclidian space with base $\{E_i\}$ an algebra is determined, then any point Z of this space may be represented in the positional number system $<\rho = -R,\ A_{R^n}>$, where the set A consists of the vectors (22), and the numbers $\alpha_m \in B_R$.

In particular, for complex numbers there exists a system $<\rho = -R,\ A_{R^2}>$, for example, a quaternary system <-2,{0,1,j,1+j}>, and for three-dimensional vectors with orts **i, j, k** – an octal system, where each digit takes values (23).

Theorem 8a. If in n-dimensional Euclidian space with base $\{E_i\}$ an algebra is determined, then any point Z may be represented in a normal positional system

$$<\rho = \pm E_2\sqrt[n]{R},\ \ B_R>. \quad (31)$$

Specifically, for R=2 we have a binary system of vector coding by radix (26). For complex numbers there exist systems $<\pm j\sqrt{R},\ \ B_R>$, for example, binary system $<\pm j\sqrt{2},\ \ \{0,\ 1\}>$, and for three-dimensional vectors with orts **i, j, k** – the exists a binary system $<\pm j\sqrt[3]{2},\ \ \{0,\ 1\}>$. In the last system we have:

<i>=1, <-i>=1001, <2i>=1001000, <-2i>=1000;
<j>=10, <-j>=10010, <2j>=10010000, <-2j>=10000;
<k>=100, <-k>=100100, <2k>=100100000, <-2k>=100000.

For three-dimensional vectors with orts **i, j, k** there also exists a quarternary number system $< \pm j\sqrt[3]{4},\ \{0,1,2,3\} >$, where $<4i>=1003000$ and $<-4i>=1000$.

Obviously, for the systems from theorems 7a and 8a the condition (14) is fulfilled.

References

1. D.E. Knuth. "An Imaginary Number System", Communications of ACM 3, No. 4, 1960.
2. S. Khmelnik. "A Specialized Computer for Operations on Complex Numbers". Questions of Radio Electronics XII, No. 2, 1962 (in Russian).
3. S. Khmelnik. "Adder of codes of complex numbers". Questions of Radio Electronics XII, No. 3, 1965.
4. Penney W., "A 'Binary' System for Complex Numbers", Journal of ACM 12, No. 2, 1965, pp. 247-248.
5. S. Khmelnik. "Positional Coding Systems for Complex Number Presentation". Questions of Radio Electronics XII, No. 9, 1966 (in Russian).
6. S. Khmelnik. "Number system with complex radixes", Part in Book: Pospelov D.A., Arithmetic basis of digital Computers, "Visshaja shkola", 1970, Moscow.
7. S. Khmelnik. "Solution of the navigation problems on a digital computer using complex numbers codes", "Problem special radio-electronic", series "Tele-mechanics and management Systems", 1971, No 6, Moscow.
8. S. Khmelnik et al. "Digital devices with microchips", "Energia", 1975, Moscow.
9. S. Khmelnik. "Computer Arithmetic of Vectors, Figures and Functions", Mathematics in Computers, 1995, Tel Aviv (in Russian).
10. Miller J. M., "Elementary Functions. Algorithms and Implementation". Birkhauser, 1997, Boston.
11. T. Aoki, H. Amada, and T. Higuchi: "Real/Complex Reconfigurable Arithmetic Using Redundant Complex Number Systems". In Proc. 13th Symposium on Computer Arithmetic, 1997.

12. Y. Chang and K. Parhi. "High Performance Digit Serial Complex Number Multiplier-Accumulator", Proc. Int. Conf. on Computer Design,1998.
13. A. M. Nielsen and P. Kornerup, "Redundant Radix Representation of Rings", IEEE Transactions on Computers, Vol. 48 (11), November 1999
14. S. Khmelnik, "A Method and System for Processing Complex Numbers". International Patent Application, WO 01/50332, 2001. . United States Patent Application No. 10/189,195, 2002.
15. S. Khmelnik, "A Method and System for Implementing Coprocessor". Canadian Patent Application, CA 02293953, 2000.
16. S. Khmelnik. "Method and System for Processing Matrices of Complex Numbers and FFT", Canadian Patent Application, CA 2339919, 2001.
17. S. Khmelnik, "Method and System for Processing Matrices of Complex Numbers and Complex Fast Fourier Transformation". International Patent Application, PCT/CA02/00295, 2002
18. S. Khmelnik, A Method and System for Processing Complex Numbers. European Patent Office, EP 1248993. Priority 12.07.01.
19. S.I. Khmelnik, I.S. Doubson, S.M. Khmelnik, A.E. Viduetsky. Arithmetic Unit for Codes by Negative Radix. Publisher «Mathematics in Computers», Israel, Printed in USA, Lulu Inc. No 93352, 146 p., 2004 (in Russian).
20. S.I. Khmelnik. Computer Arithmetic of Complex Numbers and Vectors. Theory, Hardware, Modelling. Publisher «Mathematics in Computers», Israel, Printed in USA, Lulu Inc. No 560836, 387 p., 2006 (in Russian).
21. S.I. Khmelnik, I.S. Doubson, S.M. Khmelnik, A.E. Viduetsky. Arithmetic Unit of Complex Numbers. Publisher «Mathematics in Computers», Israel, Printed in USA, Lulu Inc. No 641090, 110 p., 2007 (in Russian).