THE MODEL OF HUMAN SHORT-TERM MEMORY

I.J.Bereznaya, R.M.Granovskaya
Leningrad University, USSR

Summary

The short-term memory under consideration differs from other memory systems of the same sort in so far as it has the following properties: (a) storage of codes in the dynamic memory and analysis of codes are combined in time; (b) information is stored in both structures (dynamic and static) in a "squeezed" form and the degree of "squeezing" differs for dynamic and static memories: (c) codes may be restored both out of dynamic and of static storage.

The model of memory proposed is invariant relative to the object's size and to changes of the object's position on the plane, to its negative image and to the object plane turning about.

The basic requirements put to memorizing devices of computers are those which provide for an increase in their capacity and the rapidity of their action.

The increase in the capacity of the memorizing device leads now to a reduction in the rapidity of its action as the time needed to find objects of information at large increases and so does the addressing part of the code - descriptor. Hence, requirements put to memorizing devices while elaborating new computing systems, i.e., greater capacity and greater rapidity of action are essentially mutually contradictory. Human memory by far excels the best artificial storing devices in so far as flexibility, capacity, the rapidity of the selection of information are concerned.

This contradiction makes it necessary to syntheeise new memorizing devices on bionic principles, i.e. on the basis of human memory and studies of higher animals, and to transfer corresponding principles and regularities to information storage techniques. Unlike memory blocks in computers which provide for a clear understanding of the mechanisms and principles of information storage, mechanisms and principles of how memory works with man and animals have been but vaguely studied so far and the way they process information may be surmised only through a series of indirect indices and almost exclusively by way of comparison of the time-space distribution of signals bearing information at inputs and outputs of the system.

These processes differ one from the other in so far as the manner for the storage of information (neurodynamic process and stable structural changes), the time of its storage (hours and life time) and the type of the agent destroying traces are concerned. This investigation pursues the following tasks, (a) The formalisation of the main physiological and psychological data proceeding from mechanisms of functioning and properties of structural elements of the operative (short-term) human memory, (b) The construction (synthesis) of a mathematical model of the short-term memory out of neuron-like logical elements, a model which would not contradict psychological and physiological data, (c) An analysis of the work of the model in systems for recording, storage and reproduction of information, (d) A comparison of properties of the model offered with the memory of living systems by some parameters.

Certain principles of processing information in human short-term memory are used in the model: the transfer of the information arriving at the input successively to spatial distribution of traces, the combination of the process of analysis and that of storage, the storage of information in a squeezed form, which enables one to generalize it

In the model under investigation the the information arriving at the input of short-time memory (SM) is stored successively in two different forms: first, as a dynamic process on a certain neuron structure, then as a static distribution of spatial properties of the neuron structure. The size of information in the dynamic form of storage is limited. The size of SM as to the number of structural elements is also limited.

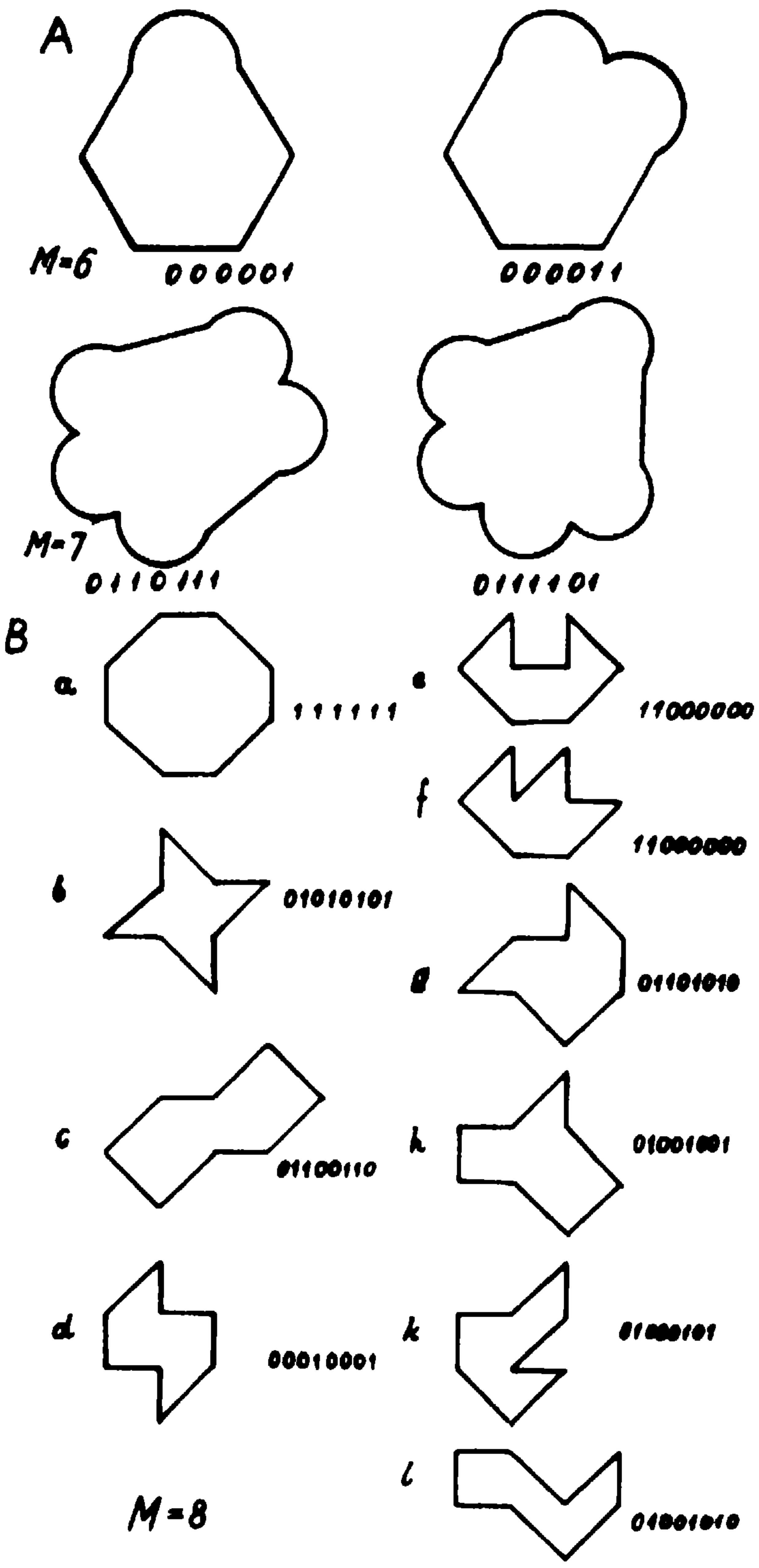
The period of fixation of information over other information may be introduced in SM, the information in the dynamic form of storage being erased. Information arrives at the input of the SM model from outputs of receptive and processing neurons (which pick out signs of the object and form the primary and secondary codes) successively in time.

Simple arbitrary geometrical figures were chosen as objects bearing information which enters SM model. Each figure is a closed broken line - contour of a polygon. The broken line consists of segments with the length of 1 and (the side and the diagonal of a square). Contiguous segments of the broken line form angles of 45°, 90° and 135°. All the figures are subdivided into ranks according to the number of their apexes.

The rule of polygon angle evaluation may be also formulated as follows: in case of clockwise trajectory of eye - tracking system while tracking the polygon angle, the interior angle of the polygon must be evaluated: in case of counter-clockwise trajectory, the exterior angle of the polygon must be evaluated. The interval is coded as one, if it includes the angle of 135° and as zero if the value of the angle is 45° or 90°. While coding, the figure contour tracking is performed clockwise and may begin at any arbitrary point. One may assume that in some sense the codes of the figures under consideration characterize the succession of concavities and convexities of their contours.

One more class of figures is used as the input objects. These figures are constructed like this. Some sides in a "right" polygon are changed to half-circles (circumferences). These half-circles are constructed on respective sides of the polygon, the latter being their diameters. The figures obtained contain both straight and curved sides. All the figures are subdivided into ranks according to the number of their sides. Fig. 1,A shows one figures of ranks V,V1 and their codes. The curved side is coded as one, the straight one is coded as zero. The only binary succession corresponds to every figure of this class.

The simplest linear uniform circuit without ramifications consisting of models of neurons is used in the model under investigation as a structure to store information in the dynamic form. This structure is called the main chain (MC).



Pig.1,A. Different rank (M=5,6,7,8) figures and corresponding codes. B, Rank M=8 figures of another class and corresponding codes.

MC is composed of successively connected on-off neurons with one input (n=1) without memory (s=1) and threshold m=1 (Pig.2). The equation of the element m is as follows:

p(t+1) = fifleCt+l) - e(t)| - m(t)] ,

$$E(x) = \begin{cases} 1, & x \geq 0, \ p(t) \text{ - output signal,} \\ 0, & x < 0, \ e(t) \text{ - input signal.} \end{cases}$$

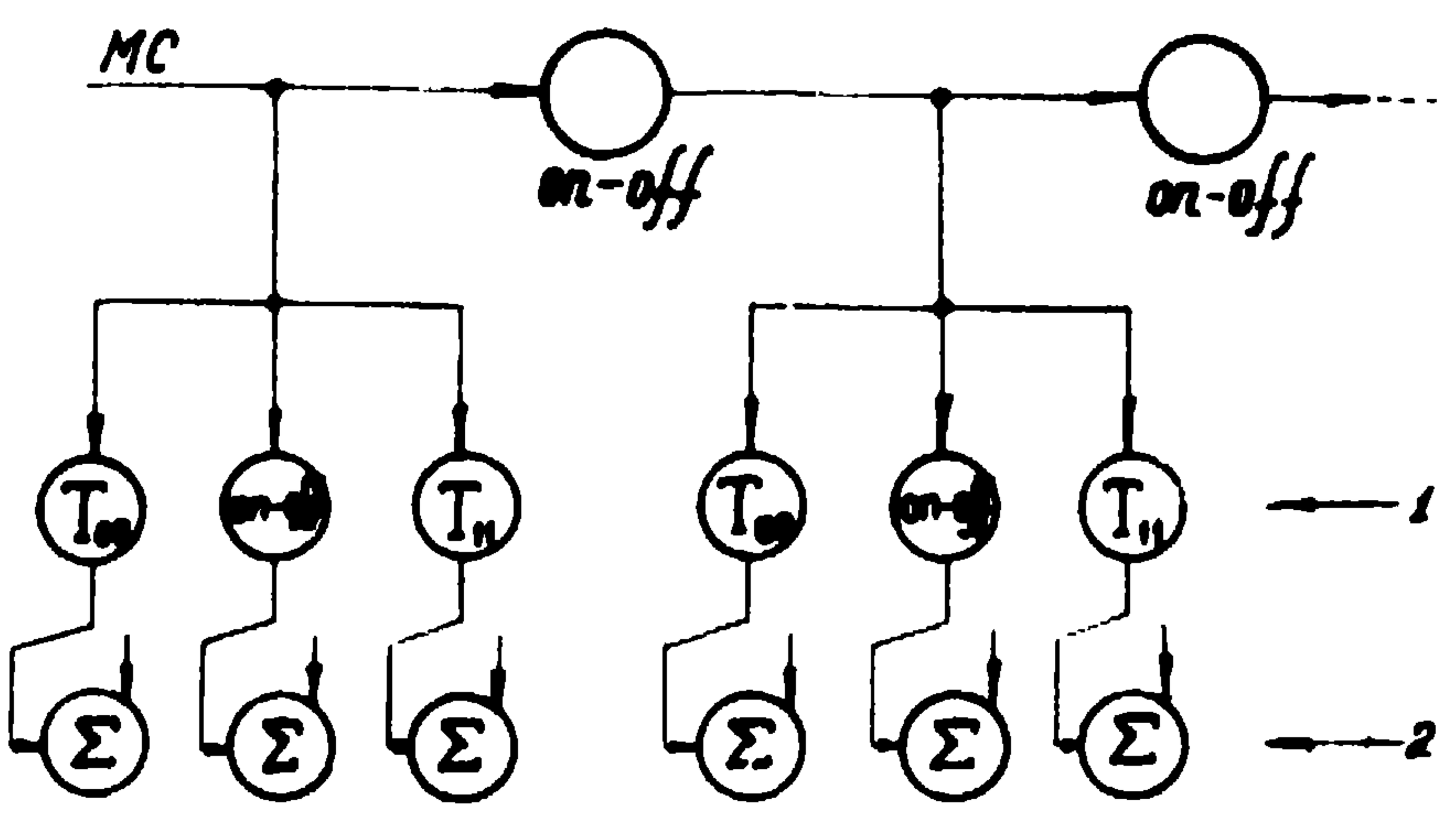


Fig.2 Main chain. Organization of static storing network (there is no restoring network at the figure)· 1 - indicators, 2 - counters.

The storage of the input code in the dynamic form is compared with the state of the elements composing MC at the given moment of time which is conditioned by the type of the input code and its transformation on every neuron of MC.

A separate symbol of the output code of every neuron is determined at a certain moment as modulo 2 sum of the input symbol for this neuron at the tact (moment) under consideration and the one proceeding to it. The value of the output symbol of any neuron in the MC depends on the structure of the chain Neuron number) and on the symbols of the input sequence.

In spite of the fact that all the neurons composing the MC are identical each of them "picks out" a definite sign of an object, the nature of the sign being de by the neuron number, i.e., with the arrangement of the MC like this the division of the functions of the analysis of the object's code along the chain takes place.

The output sequence of a MC neuron is determined at the given moment by states of certain neurons, not by those of all of them. It has been shown that the total amount of neurons whose state determines the output sequence of every neuron is always even, and the numbers of these neurons are symmetrical with respect to the middle of the chain.Whatever the number of neurons in the MC, the value of the output symbol of the n neuron necessarily depends at any moment on the state of the first neuron n tacts back from the moment under consideration and on that of the n neuron at the given moment. It has been revealed that the value of the output symbol of any neuron of the MC is connected with the number of units in the row of the so called residue modulo 2 triangle where the row number n is equal to that of the given neuron (Pig.3).

Properties of the MC as a whole sharply change when its length is changed by a unit, especially by transition from one layer of the residue modulo 2 triangle to another.

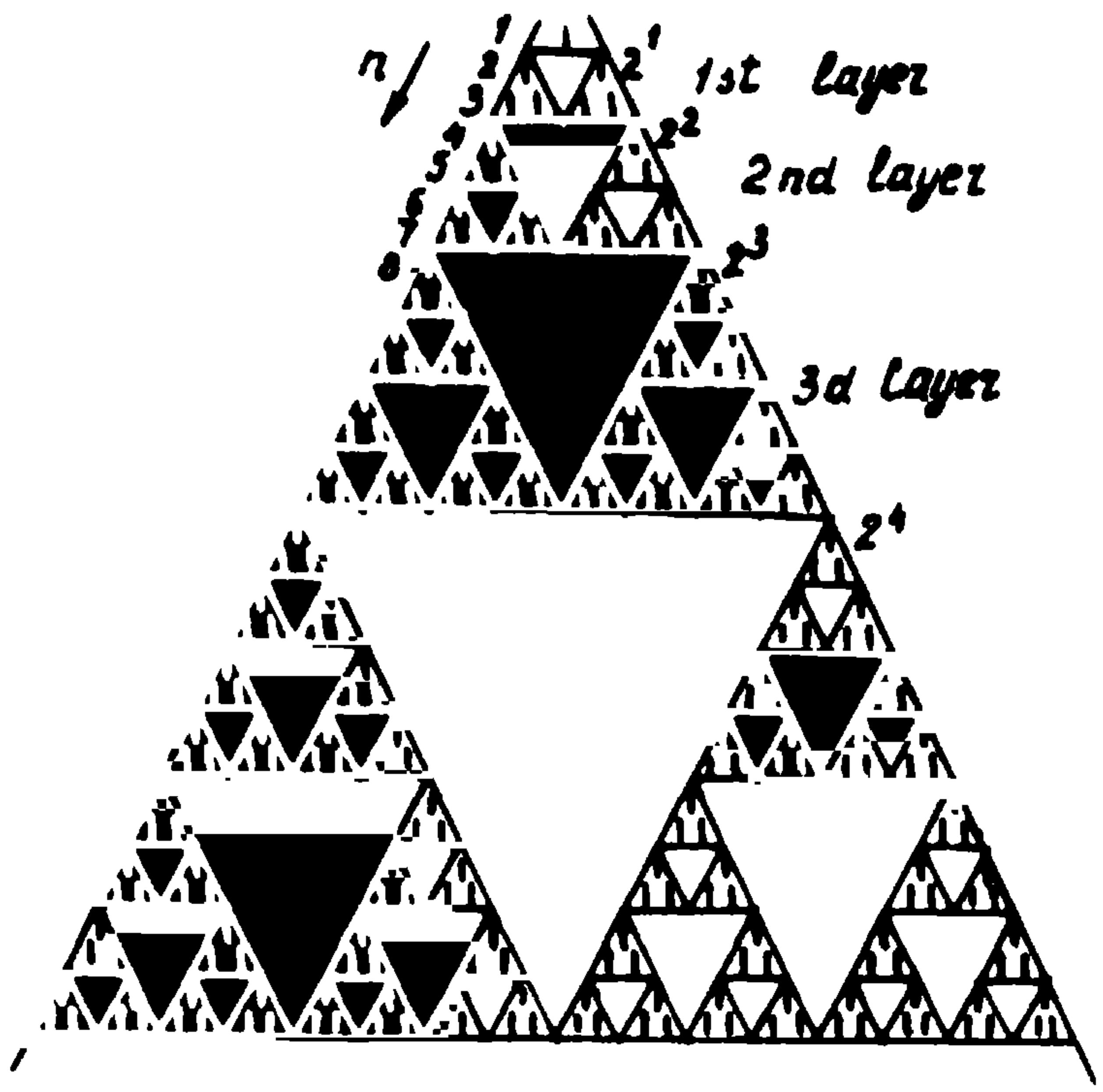


Fig. 3 A residue modulo 2 triangle.

All the input sequences for the MC can be subdivided into two categories (according to transformation in MC character) (Fig.4).non-zero and zero ones. A zero sequence is an input succession which upon entering the input of the MC at a certain moment of time will produce an output succession in a definite number of tacts after this moment consisting of zeroes only (000....0). If a zero sequence (ZS) enters the input of the MC, the information it is bearing will not go further than a certain neuron, in other words, the MC of a definite length appears to be "tuned" to a class of zero sequences, corresponding to its length, the turning being meant as the property of the MC to stem codes of a certain structure. This structure is determined by the number of ranks and the law of one-zero alteration.Thus, in case of the dynamic storage, information of the oDjects which have zero codes corresponding to them does not reach the output of the MC.

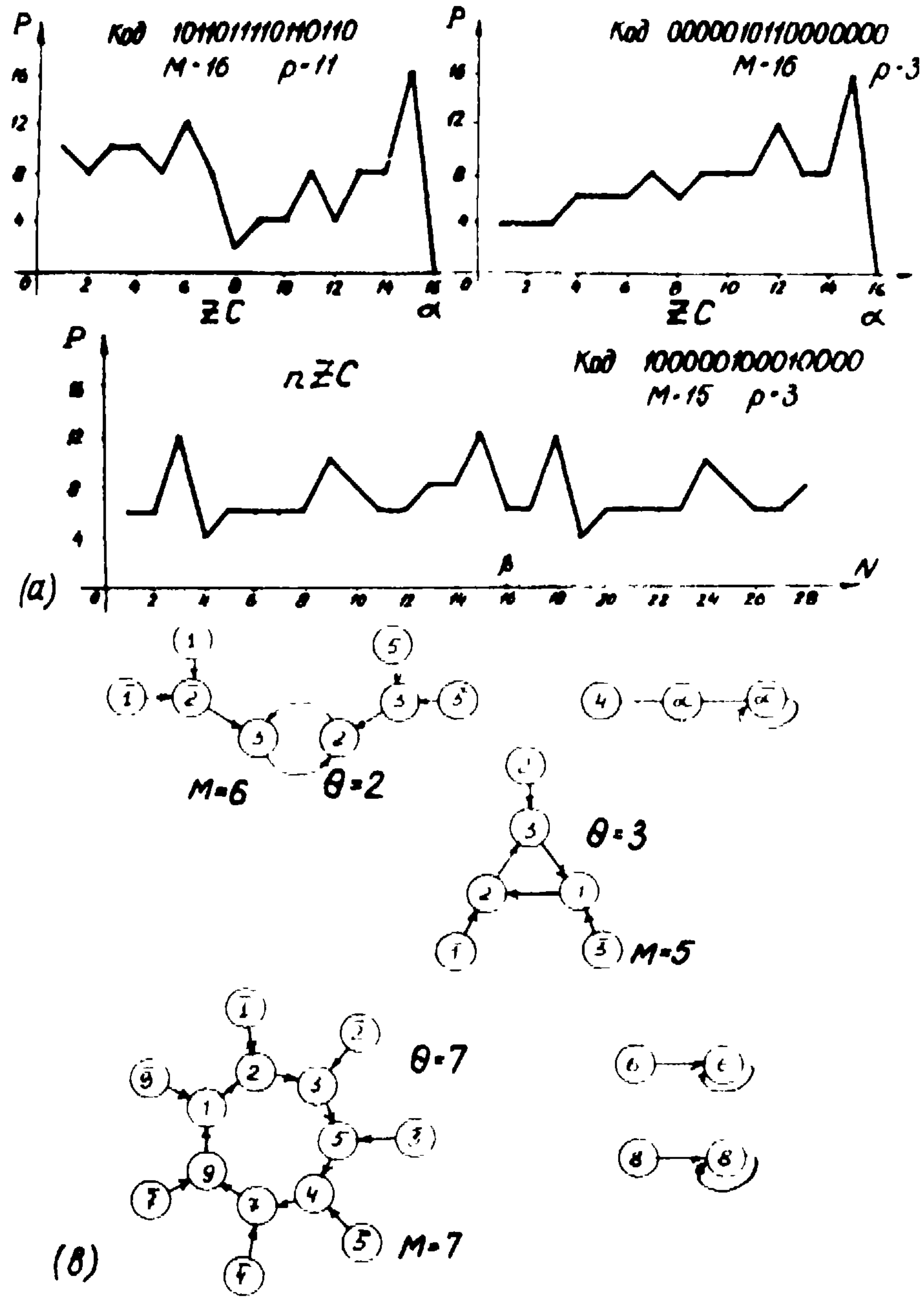


Fig.4 (a) Zero (left) and non-zero (nZC) (right) code transformation dynamic process: M - a quantity of code ranks, P - a quantity of unite in the code, N - a number of chain neurons, (b) Non-zero code transformation dynamic procesB for codes of ranks M=5,6,7.

Zero sequences are a regular system - a converging tree, every apex of which corresponds to one class of the ZS equivalence. One class of the equivalence comprises sequences obtained while tracing one and the same figure, starting points of the tracing being different. Every edge of the tree corresponds to a transformation in one MG neuron of a certain input sequence situated (placed) in the apex, whence the edge issues into a succession on its output situated in the apex where the edge enters. Pig.5 shows a n-rank tree of the ZS (n=8). Layers similar to those in the residue modulo 2 triangle can be distinguished in it. The tree of zero sequence is dichotomous except for borders between layers.

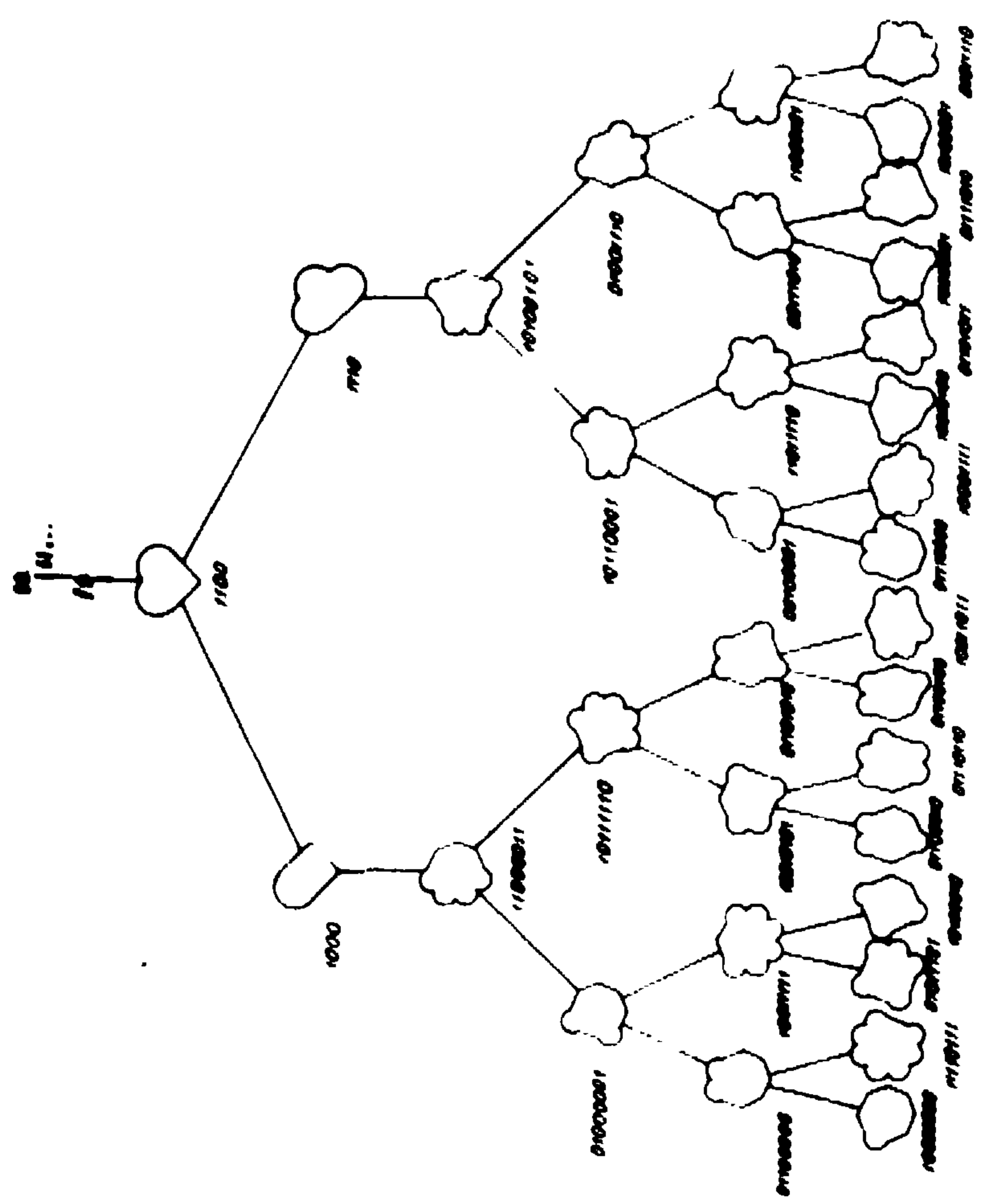


Fig.5 A zero-code transformation tree and corresponding figure transformation tree.

Sequences with equal subperiods are situated in all the apexes of one layer. If one distinguishes three successive layers in the ZS tree, the period of ZSs situated in the upper layer is twice as great as that of ZSs situated in the middle layer.

Movement along the edges of the code tree from a certain apex of the n rank towards the apex of the tree corresponds to the indicated simple transformation of the input sequence situated in this apex during its passage through on-off neurons from the first towards the n one which will have a sequence consisting of zeros only at its output .

If one examines the output sequence of any neuron of the MC the tree of zero sequences will help ascertain what sequence the given sequence will turn into after being transformed on subsequent neurons of the main chain, but there is no direct and simple answer as to transformation of what input sequence results in it. In case input codes with the period not equal to a power of two (non-zero codes) pass through the MG no neuron will yield a sequence consisting of zeros only.

In case input non-zero codes pass through the main chain, a group of successively connected neurons may be picked out of the chain of neurons. A periodic process of code transformation with a period of 0 neurons develops within the group (Fig.4). Number 0 as a characteristic of tne transformation of the input codes essentially depends on the period of the code, not on the type of it, Examples of how various types of the input sequences are transformed are given in figure 4,b.

Transformation of non-zero codes is also a regular system which may be presented as a graph having both cycles and an acyclical part (Pig.4,b).

The knowledge of laws of graph formation enables one to calculate the minimum length of the MC on which the full analysis of the input code signs takes place (code parameters given) and to determine the numbers of the neurons on which the input code appear, which is of importance for the detection and elimination of disturbances in the shape of casual code distortions in the main chain. The main chain may be compared with a set of filters in which the variety of properties of every filter is connected with the number of the corresponding neuron in the chain.

The figures of the same rank may be compared according to the number of connected successively MC neurons in which the non-periodic process of code transformation develops. Let us call this number code the structural characteristic (of the figure) - SC.

Codes with the inner subperiods correspond to figures with periodic properties. The transformation of such codes in the submitted model of SM is characterized by a smaller SC value. Besides this, the shorter the code subperiod (with the rank unchanged), the smaller the SC value. Thus, recording and analysing of a code with inner subperiods in the dynamic and static memory require less time and less structural elements (neurons of the MC and those of the static memory) than the identical processes for the figures with non-periodic properties.

The perculiarity of periodic figure code transformation in the given model of SM may be compared with the perculiarity of perceptron, storage and recognition of the so called "good" figures (Pig.1). These are known to include symmetrical, and, particularly, the figures with periodic properties belonging to "good" ones.

Figures in fig.1,B are divided into "good" (a-d) and "not good" (e-1) in accordance with presence or absence of periodic properties. It takes only one MC neuron and one corresponding group of the static memory neurons for figure a code analyses, two MC neurons for figure b code analyses, three and four MC neurone for polygon c and code analyses but seven-eight MC neurone for polygons e-1 code analyses.

Thus the better the figure the lower its SC value, which results in determining the numerical evaluation of good figure properties.

The process of the passage of the visual object's code through the MC is compared with the storage of information in the dynamic form. Let us assume that before the following sequence arrives the main chain will completely purify. The state of the MC can be characterised by a certain rectangular table, each line of which corresponds to the code at the output of a MC neuron at various moments while a column corresponds to states of the output of MC neurons at one and the same moment. If states of outputs of the initial k-neurons at t_k moment are known, the input code can be restored, i.e., if elements of the k column (**$k \leq N$, N** - the length of the MC) are known, the zero line (input sequence) can be determined. Algorithm of the code restoration out of the dynamic storage can be realized by a network consisting of the novelty (H) and the summarizing (Z) neurons. Equations of these types of neurons are shown in fig.6. The elements of MC do not possess memory, and their possessing it is not expedient, since the chain must be purified before introducing the new code. That is why the results of the analyses necessary to the organism in the future, should be transferred from dynamic storing into a static one

Σ	$p(t+1)=1\left[\sum_{i=1}^{n} S_i(t)e_i(t)-m(t)\right]$	$p(t+1)=e_1(t)e_2(t)$ $p(t+1)=e_1(t)\vee e_2(t)$ $p(t+1)=e_1(t)\bar{e}_2(t)$ $n=2$	$p=1\left[\mathfrak{D}\left(\sum_{i=1}^{n} S_i e_i - m\right)\right]$ $S_r e_r \geq m_0$ $1(x)=\begin{cases}1, & x\geq 0\\ 0, & x<0\end{cases}$	$p=\mathfrak{D}(e_1e_2)$ $p=\mathfrak{D}(e_1\vee e_2)$ $p=\mathfrak{D}(e_1\bar{e}_2)$
on-off	$p(t+1)=1\left[\sum_{i=1}^{n}\lvert S_i(t)e_i(t)-S_i(t-1)e_i(t-1)\rvert-m(t)\right]$	$p(t+1)=e(t-1)\bar{e}(t)\vee\bar{e}(t-1)e(t)$ $n=1$	$p=1\left[\sum_{i=1}^{n}\lvert\mathfrak{D}S_ie_i-\mathfrak{D}^2S_ie_i\rvert-\mathfrak{D}m\right]$	$p=\mathfrak{D}^2e\,\mathfrak{D}\bar{e}\vee\mathfrak{D}^2\bar{e}\,\mathfrak{D}e$
on	$p(t+1)=1\left\{\sum_{i=1}^{n}\left[S_i(t)e_i(t)-S_i(t-1)e_i(t-1)\right]-m(t)\right\}$	$p(t+1)=\bar{e}(t-1)e(t)$ $n=1$	$p=1\left[\sum_{i=1}^{n}(\mathfrak{D}S_ie_i-\mathfrak{D}^2S_ie_i)-\mathfrak{D}m\right]$	$p=\mathfrak{D}^2\bar{e}\,\mathfrak{D}e$
off	$p(t+1)=1\left\{\sum_{i=1}^{n}\left[S_i(t-1)e_i(t-1)-S_i(t)e_i(t)\right]-m(t)\right\}$	$p(t+1)=e(t-1)\bar{e}(t)$ $n=1$	$p=1\left[\sum_{i=1}^{n}(\mathfrak{D}^2S_ie_i-\mathfrak{D}S_ie_i)-\mathfrak{D}m\right]$	$p=\mathfrak{D}^2e\cdot\mathfrak{D}\bar{e}$
H	$p(t+1)=1\left[\sum_{i,j=1}^{n}\lvert S_i(t)e_i(t)-S_j(t)e_j(t)\rvert-m(t)\right]$	$p(t+1)=e_1(t)\bar{e}_2(t)\vee\bar{e}_1(t)e_2(t)$ $n=2$	$p=1\left[\mathfrak{D}\left(\sum_{i,j=1}^{n}\lvert S_ie_i-S_je_j\rvert-m\right)\right]$	$p=\mathfrak{D}(e_1\bar{e}_2\vee\bar{e}_1e_2)$
T_p	$p(t+1)=1\left[S_re_r-m_0-\sum_{i,j=1}^{n}\lvert S_i(t-1)e_i(t-1)-S_j(t-1)e_j(t-1)\rvert\right]$	$p(t+1)=\bar{e}_1(t)\bar{e}_2(t)\vee e_1(t)e_2(t)$ $n=2$	$p=1\left[S_re_r-m_0-\mathfrak{D}^2\sum_{i=1}^{n}\lvert S_ie_i-S_je_j\rvert\right]$	$p=\mathfrak{D}^2(e_1e_2\vee\bar{e}_1\bar{e}_2)$
T_t	$p(t+1)=1\left[S_re_r-m_0-\sum_{j=1}^{n}\sum_{i=1}^{n}\lvert S_i(t-j)e_i(t-j)-S_i(t-j-1)e_i(t-j-1)\rvert\right]$	$p(t+1)=\bar{e}(t-1)\bar{e}(t)\vee e(t-1)e(t)$	$p=1\left[S_re_r-m_0-\sum_{j=1}^{n}\sum_{i=1}^{n}\lvert\mathfrak{D}^{j+1}S_ie_i-\mathfrak{D}^{j+2}S_ie_i\rvert\right]$	$p=\mathfrak{D}^2\bar{e}\,\mathfrak{D}\bar{e}\vee\mathfrak{D}^2e\cdot\mathfrak{D}e$

Fig.6 Different types neurons equations

The model of structure of static storing is a network, consisting of groups of neuron-like elements, equal both in composition and in structure. The number of such groups corresponds to the number of on-off neurons in the chain. Each group has elements to distinguish certain kinds of situations in the input code of the corresponding neuron of MC (indicators), and element accumulating information of the number of repetitions of these situations (counters) fig.2,o. Thus each group analyses the input code of one neuron of MC and stores a certain kind of information of this code, and may be compared with static memory of each element of the structure of dynamic memory.

By code situation we meen a pair of binary symbols corresponding to the values of the neighbouring positions of the code, For analysis and storing in static memory three types of code situations at the input of each on-off neuron of MC are chosen: 1) 00 is the absence of signal difference in the absence of these in the neighbouring measures of time (situation α)- 2) 01 or 10 is signal difference (switch on - switch off type - situation β): 3) absence of signal difference in their presence during the neighbouring measures of time (situation γ).

The code arriving at the input of any neuron of MC arrives simultaneously at the inputs of the three elements of static memory, one of them responding to situation α , the other doing so to β , the third to γ (Fig.2,b). The model of neuron identity in time (T_t^{oo}) reacting only to the absence of signals at its input at two successive instants is used as the first element. The on-off neuron without memory similar to an element of MC is taken as the second element while the model of neuron identity in time T_{tt} responding only to the continuous signal at its input during two successive measures of time is taken as the third one. The output of each of the three described elements, analysing the code situation at the input of some on-off neuron of MC, is bound to the input of the corresponding element-counter (Σ) accumulating and storing information of the number of situations of the given type in the input code of this neuron. Equations of neurons of the types T_t^{oo}, T_t^{11}, Σ are given in the fig.6. Thus the static memory of every neuron of MC includes three analysing elements without memory and three accumulating elements with it.

If we take the static memory of one neuron of MC as an element of the structure of static storing, this structure will be uniform in its original state. While the code of an object passes through MC and undergoes dynamic changes, the structure of static storing becomes non-uniform, while information of code situations is being accumulated. Arrival of the code of the following object will find MC in the same state, as the previous one. But the static memory will be in a rew state by this moment.

Accumulating elements being not uniformly distributed in space corresponds to their being not uniform in time. This absence of uniformity may be observed at the input of each element of the dynamic structure. Presence and measure of this correspondence determines both the very possibility and exactness of restoring of the code of an object by its traces in static memory.

While the code passes through MC, traces of the input code of these elements are imprinted in the static memory of every neuron: elements-counters accumulate and store information of the number of pair-zeroes, pair-units and transitions from zero to unit and from unit to zero. After the code of an object has completely passed through all the neurons of MC, the properties of static memory counters reflect the general number of situations α, β, γ in the former codes; besides this some information of the order of alterations of the three types of code situations is lost in transferring the codes from dynamic into static storing.

This means that identical traces are left by the codes obtained in tracking the contour of one and the same object starting from different points of it, Besides, similar traces in static memory are left by different objects, the codes of which have the same number of situations α, β, γ. The codes of such objects will be called equivalent.

Thus, the algorithm of restoration is essentially recurrent, the maximum number of stages being equal to the number of elements in the main chain. The algorithm of restoration enables one to compare the test code with information in memory with any degree of accuracy which is determined by the chosen number of stages of the algorithm of restoration that are used.