World Applied Sciences Journal 30 (12): 1932-1938, 2014
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2014.30.12.14390

K-Value Adaptive Resonance Theory of the Neural Network for Analyzing the Operability of Computing Devices

[1]*Leonov S. Yu,* [1]*V.D. Dmitrienko and* [2]*T.V. Gladkikh*

[1]National Technical University "Kharkov Polytechnic Institute", Kharkov, Ukraine
[2]TOA "Ukraine", Kharkov, Ukraine

Abstract: In this article, the use of K-value adaptive resonance theory of the neural network for the classification of the K-value output signal of logical elements and devices of computer engineering has been discussed. This network allows automating the processes of identifying situations that may cause crashes in digital devices.

Key words: K-value neural network . adaptive resonance theory . classification of K-value logic gates . signals and devices . failures in digital devices

THE PROBLEM STATEMENT AND ANALYSIS OF LITERATURE

At present, analyzing the capacity of designed devices without the application of different design system is impossible. Well-known systems are the Boole functional design of OrCAD [1] and PCAD [2] and systems of analog design Micro-Cap [3]. However, most of the binary design systems allow obtaining the complete picture of complicated modern fast-acting microcircuits with high integration degree switching character. In addition, analog systems do not allow executing the analysis of difficult devices. In this connection, in [4], the use of a computer-aided design based on the K-value differential calculus, which allows a more complete analysis, in comparison with the binary design, to present the processes of Boole signal switching and simulate difficult computing devices is suggested [5, 6]. However, the practical use of this design system requires information on the risk of failure in switching processes and the correct switching from one stable state to another state.

Development of the programmatic realization of the neural network, which was used for determining the types of switching signals in logical elements on the basis of the 13-digit alphabet of Fantozi [8], was considered in the article [7]. A double-layer neural network on the basis of K-value neurons, which was taught on the basis of the Khebba rule for K-value neurons, was used. However, in practice, data, which are used for training or self-training networks, are often incomplete. For example, if we consider an image, which refers to the new class and has to be recognized, then the ordinary double-layer neural network would fail to do this task [9-11]. The neural network Khebba cannot identify new images and also does not have the ability to continue the training process because the new image training results in distortion of the already memorized information. Hence, for correct training, all memorized information must be used. Thus, double-layer or multilayered networks used in teaching the algorithm Khebba or the method of backpropagation error do not have the stability property, that is, the property to save the known information while memorizing new information.

Networks of adaptive resonance theory (ART) were developed to solve this problem, that is, a neural network memorizes new classes of images without distortion or loss of already storable information [12]. In this case, the use of correct switching and switching neural networks ART, which contain information on the risk of failure during transition of logical signals from one stable state to another state, is suggested.

The input image of the signal network of ART-1 tries to associate it with some class from a number already present. In particular, one type of switching of output logical element from one stable state to another state can be used. If the class of switching has to be determined, then the classification of the signal is closed. Thus, a signal can be referred either to the class of correct switching or to the class of signals, which is the risk of failure. If the corresponding class was not determined, then a new class is created. Afterward, the created signal is used as a prototype (typical representative) for a new class. Thus, the known classes do not change.

Corresponding Author: Leonov S. Yu, National Technical University "Kharkov Polytechnic Institute", Kharkov, Ukraine

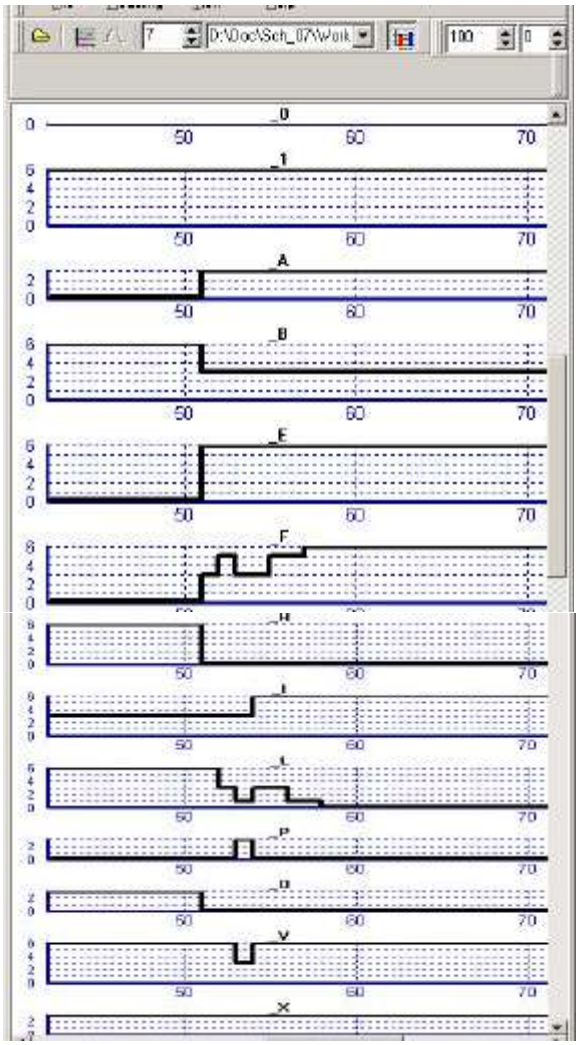

Fig. 1: Thirteen types of signals in K-value view by using the seven-digit alphabet. The design system was based on the K-value differential calculus

BASIC PART

The goal of this article is the development of a K-value neural network on the basis of ART for the classification of signals, which have been placed during the design of digital computing devices in the design system based on the K-value differential calculus.

Let us consider the K-value ART neural network which classifies 13 types of signals in K-value view by using the seven-digit alphabet. The input data is obtained from the design system based on the K-value differential calculus. Figure 1 shows the 13 typical signals obtained from this system.

Based on Fig. 1, when using the seven-digit alphabet, the following 13 types of signals are used [5]: the top signal "_0" corresponds to the static binary signal at the zero level, signal "_1" corresponds to the value "K-1" and is related to the value of the level of the static binary "unit," signal "_A" corresponds to the transition from a zero state to an indeterminate state "(K-1)/2," signal "_B" corresponds to the transition from "(K-1)" to the indeterminate state "(K-1)/2," "_E" corresponds to the transition from a zero state to "(K-1)," "_F" corresponds to the dynamic risk failure during the transition from a zero state to "(K-1)," "_H" corresponds to the normal transition process from "(K-1)" to the zero state, "_I" corresponds to the transition from an indeterminate state "(K-1)/2" to "(K-1)," "_L" corresponds to the dynamic risk failure during the transition from "(K-1)" to the zero state, "_P" corresponds to the static risk failure in the zero state, "_O" corresponds to the transition from an indeterminate state "(K-1)/2" to the zero state, "_V" corresponds to the static risk failure in (K-1), signal "_X" corresponds to the value of the indeterminate state "(K-1)/2" at the binary presentation of signals.

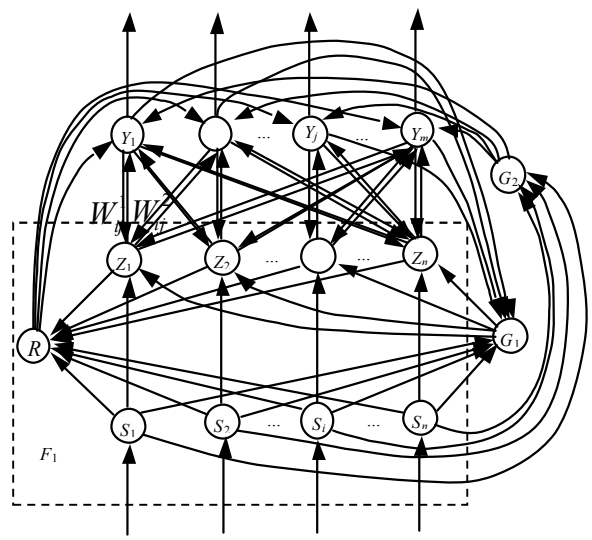

Fig. 2: Architecture of the ART-1K neural network

These signals must be stored in the memory of the K-value ART-1K neural network. The architecture of this network is shown in Fig. 2.

The architecture of the network consists of three groups of neurons: field F_1 input processing neurons, which, in turn, consists of two layers S-elements and Z-elements; layer of recognition Y-neurons; and control neurons R, G_1 and G_2 (Fig. 2).

Field F_1 input processing neurons consists of two layers, namely, input layer S-elements and interface layer Z-elements. The input layer obtains the image and passes the obtained information to the neurons of interface layer Z-elements and control neurons R, G_1 and G_2. Every element Z_i (i = 1, …, n) interface layer is

related to every element Y_j (j = 1, …, m) recognizing layer Y by two types of the weighted connections. Signals from an interface layer are passed on to layer Y by the top to bottom connections with weights W_{ij}^1 and from a recognizing layer to an interface layer by connections with weights W_{ij}^2 (j = 1, …, m; i = 1,…, n). Considering the large number of connections, only one pair of connections with weights W_{ij}^1 and W_{ij}^2, between the interface and recognizing layers of elements is shown in Fig. 2.

Layer Y is the layer of competitive or competing neurons. At any moment, every element Y_j (j=1, …, m) recognizing layer is in one of the following three states:

- Active (output signal U_{outY_j} of neuron Y_j is equal to d: $U_{outY_j} = d$, where d = 6 for ART-1K at K = 7 value of the source alphabet);
- Inactive ($U_{outY_j} = 0$, but a neuron can participate in a competition);
- Inhibited ($U_{outY_j} = -1$ and a neuron is not allowed to participate in a competition).

After producing the input image, only one recognizing neuron is active and all other Y-elements have zero or subzero output signals. In the training mode, the selected recognizing neuron is allowed to teach by using only the input image in the case when its weighting connection vector from layer Y to Z is similar to the entrance vector. This solution is accepted with the help of the R-neuron and the special parameter, which is called the parameter of likeness and signals, which are transferred from the input and interface element layers. Further exclusion of the selected recognizing Y-element training or inhibiting with the neuron from the set of competing neurons is performed using the accidental elements. Inhibiting is performed in the case when the same input image is repeatedly entered, if new candidates to training by input image are selected from the Y-layer.

Most connections, which are shown in Fig. 2, are excitant: from an input layer S-elements to the neurons R, G_1 and G_2 in the Z-layer and from neurons G_1 and G_2 to the neurons of layers Z and Y, correspondingly. Inhibiting signals pass the large numbers of connections from interface elements to R-neuron, from Y-neurons to element G_1 and from R-neuron to the winner neuron in a recognizing layer only. All connections of the ART-1K network pass K-value signals from an alphabet M = {0, 1, 2, …, K-1}.

Every element in an interface or Y-layer of the ART-1K network has three sources of input signals. Any interface element Z_i (i = 1, …, n) can obtain signals from an element S_i of the input layer and from Y-layer elements and neuron G_1. Analogously, element Y_j (j = 1, …, m) can obtain signals from interface elements and neurons R and G_2. For switching of interface or recognizing layers in neurons to the active state, the existence of two sources of input excitant signals is needed. As each of the examined neurons has three possible sources of signals, the condition of excitation of these neurons was called "rule two from three."

The neurons R, G_1 and G_2 and input layer S have zero output signals in the initial state. When K-value component input images are entered into the S-elements, a part of the elements, which is in the nonzero state (U_{BbIX}>0), is switched to the excited state. Excitant signals from the outputs of these neurons switch neurons G_1 and G_2 to state "6" and also insert the input of the corresponding neurons to the interface layer. Neurons of interface layer, which obtained the signals from the neurons of the input layer and element G_1, by rule two from three pass the active voice and send the excitant signals to connections with scales W_{ij}^1 (i = 1, …, n; j = 1, …, m) upon entrance of neurons to the Y_j recognizing layer. The neurons of the recognizing layer pass the active voice by rule two from three when obtaining the excitant signals not only from the elements of the interface layer but also from element G_2. Output signals of active Y-neurons are expressed as follows:

$$U_{outY_j} = U_{inpY_j} = \sum_{i=1}^{n} W_{ij}^1 \, U_{outZ_i}, \quad j = 1, \ldots, m$$

and meet the following condition:

$$0 < U_{outY_j} \le 6$$

Then, a lateral process of selection of the unique element J with the largest output signal takes place in Y-layer neurons. All Y-layer neurons, except the winner Y_J, are switched to the inactive state "0" ($U_{BbIX\,Y}$ = 0). In addition, the winning neuron is switched to the state with unit (K = 6) output signal. The winning Y-neuron signal inhibits the control neuron G_1 and also inserts the interface layer neuron inputs by connections with weights W_{ij}^2. Given that the elements of the interface layer follow the rule two from three, when the excitant signal from neuron G_1 is absent, the interface elements, which obtain the signals of the input layer element and winning neuron Y_j recognizing layer, are in the active state only. The inhibiting signals of interface layer active elements insert the input of the

R-element, which also obtains excitant signals from the neurons of the input layer. Depending on the ratio between excitant and inhibiting signal values, the output signal of control element R is determined.

When the R-element output signal is equal to zero, resonance occurs in the neural network, as well as the training of the connection weights of the winning Y-neuron. When the output state is equal to unit, the winning Y-neuron is inhibited ($U_{BbIXY_j} = -1$) and, in practice, loses the possibility to take a participant in the competition when the input image is entered. Then, in the Y-layer, the new winning neuron is chosen. If the input image is not similar enough to one of the memorized images, all of the Y-neurons used will be inhibited and a winner will be selected from the unused neurons, which memorizes the new image in its weights.

On the basis of the rapid teaching method of the ART-1K neural network, a training algorithm, which can use the following denotations, is accepted:

m: The maximum number of recognizing elements in the Y-layer or the maximum number of recognizable image data;

n: A number of components in an entrance vector or image;

S^k: n-measured K-value input vector, k = 1, …, q;

q: Number of input vectors;

$U_{outZ} = \left(U_{outZ_1}, \ldots, U_{outZ_n}\right)$: n-measured K-value vector of the output signals of the interface element layer;

||X||: Norm of vector X;

p: Parameter of likeness between an entrance vector and vector kept in the scale of connections of the winning neuron Y-layer; the range of legitimate values of the parameter is $0 < p < 1$;

W_{ij}^1: Weight of connection from an element Z_i (i = 1, …, n) to the element Y_j (j = 1, …, m); the range of legitimate initial values is $0 < W_{ij}^1 \le \frac{1}{6(L-1+n)}$; the recommended initial value is $W_{ij}^1 = \frac{1}{6(1+n)}$; all calculations are executed taking into account the use of the K-value signals;

L: Constant, excelling unit; the recommended value is L = 2;

W_{ij}^2: Weight of connection from the element Y_j to the element Z_i; the recommended initial value is $W_{ij}^2 = 6$, which corresponds to the K-value "unit."

TRAINING ALGORITHM

Step 1: The first step is initializing parameters L and p and weights W_{ij}^1 of connections from the interface layer to the winning layer and W_{ij}^2 (i = 1, …, n; j = 1, …, m) of connections from the winning layer to the interface layer.

Step 2: The terms of stop are analyzed and while they are not executed, Steps 3 to 14 of algorithm will be realized.

Step 3: For every teaching entrance vector S^k (k = 1, …, q), Steps 4 to 13 are executed.

Step 4: Set zero signals to all recognizing elements of the winning Y-layer, as follows:

$$U_{outY_j} = 0, \quad j = 1, \ldots, m$$

By using the entrance vector S^k, S-elements of the input layer are activated, expressed as follows:

$$U_{outS_i} = S_i^k, \quad i = 1, \ldots, n$$

Step 5: The norm of the output signal vector neurons of the entrance layer is calculated as follows:

$$\left\|U_{outS}\right\| = \left\|S^k\right\| = \sum_{i=1}^{n} S_i^k$$

Step 6: For the elements of the interface layer, the entrance vector and output signals are formulated as follows:

$$U_{inpZ_i} = U_{outS_i}, \quad i = 1, \ldots, n$$

$$U_{outZ_i} = U_{inpZ_i}, \quad i = 1, \ldots, n$$

Step 7: Settle the accounts for the output signal of every unbraked Y-neuron:

if $U_{outYj} \neq -1$, then

$$U_{outY_j} = \sum_{i=1}^{n} W_{ij}^1 U_{outZ_i}, \quad j = 1, \ldots, m$$

Step 8: While the Y-neuron of the output layer is not found, the weight vector, which is in accordance with the specified value of the parameter of likeness p, corresponds to the entrance vector S^k; Steps 9 to 12 are executed.

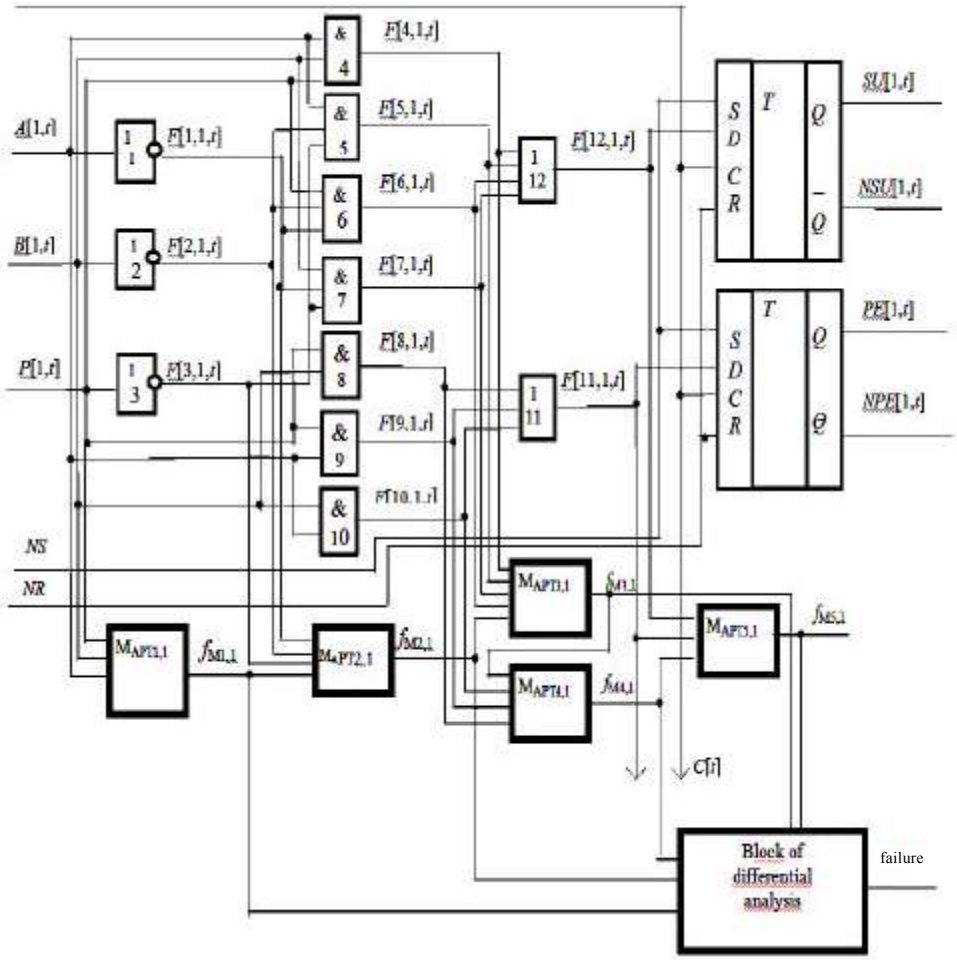

Fig. 3: Chart of one digit of the multidigit adder

Step 9: In the Y-layer, neuron Y_J is determined, which meets the following condition:

$$U_{outY_J} \geq U_{outY_j}, \quad j = 1, \ldots, m$$

If such elements are few, we obtain an element with smaller index. If $U_{outY_j} = -1$, then all elements are put on the brakes and the entrance image cannot be classified or memorized.

Step 10: Calculating the output signals of the Z-elements of the interface layer using the following K-value Boolean operation:

$$U_{outZ_i} = U_{outS_i} \langle \cap \rangle_K W_{Ji}^2, \quad i = 1, \ldots, n$$

Step 11: The norm of the output signal vector of the interface layer is calculated as follows:

$$\left\| U_{outZ} \right\| = \sum_{i=1}^{n} U_{outZ_i}$$

Step 12: Check the condition of the teaching possibility of the selected neuron Y_J.

If $\left\| U_{outZ} \right\| / \left\| S^k \right\| < p$, then the condition is not executed and element Y_J is braked, that is, $U_{outY_J} = -1$. Afterward, Step 8 of the algorithm is performed.

If $\left\| U_{outZ} \right\| / \left\| S^k \right\| \geq p$, then the condition of the teaching possibility of neuron Y_J is executed and progresses to the next step of the algorithm.

Step 13: Adapt the weight of connections of element Y_J, taking into account the use of the K-value signals, expressed as follows:

$$W_{ij}^1 = \frac{L U_{outZ_i}}{6\left(L - 1 + \left\| U_{outZ} \right\|\right)}, \quad i = 1, \ldots, n$$

$$W_{ij}^2 = U_{outZ_i}, \quad i = 1, \ldots, n$$

Step 14: The conditions of stop are checked.

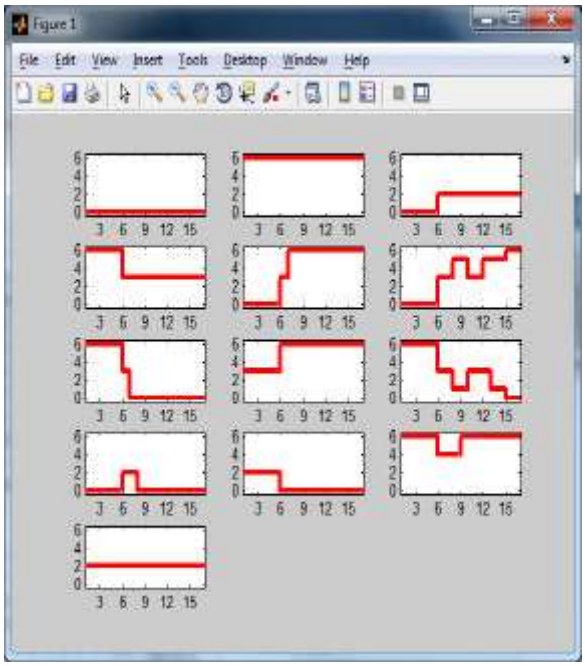

Fig. 4: K-value signals as input on the ART-1K neural network

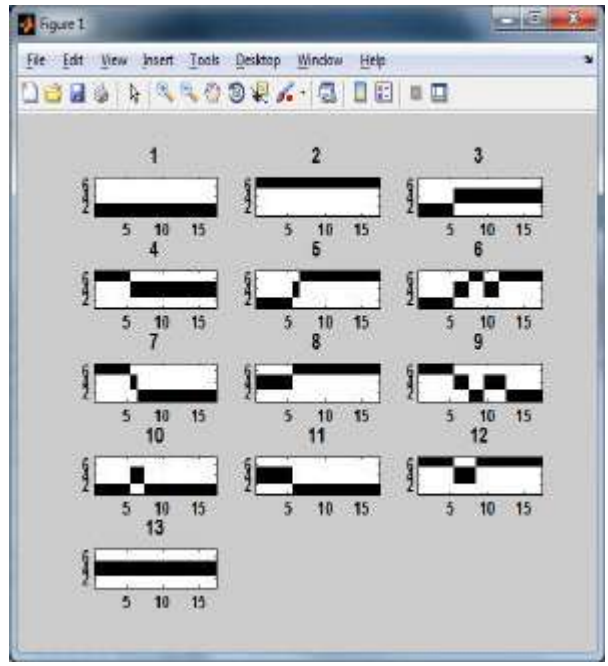

Fig. 5: Fields of K-value signals as input on the ART-1K neural network in training mode

The conditions of stop can be the absence of changes in the scales net W_{ip}^{1}, W_{ij}^{2} during an epoch and the achievement of the specified number of epochs.

Step 15: Stop.

Work on the resulting algorithm is shown in the following example.

We will consider the work on the K-value ART-1K neural network for diagnostic signals of the electronic accumulator. The chart of one digit of such adder has a different kind (Fig. 3).

On the entrance of such adder, we enter the K-value signals of the addends A[1, t] and B[1, t] and the signal from a previous digit P[1, t]. In addition, on the entrance, controlling signals of reset output triggers in a zero NR and signals of their setting in unit NS can be observed. The outputs of such device are the K-value signal of the result of the sum for the first digit SU[1, t] and signal of the next digit PE[1, t]. All entrance and internal signals that enter through the entrance blocks of the automated diagnostics are executed on the developed ART-1K neural network.

These signals can have a form of switching processes, as shown in Fig. 1, where 13 types of signals from the alphabet of Fantozi are shown, each of which can be on every entrance of the multidigit adder as new information and corresponds to the K-value signal design systems on the basis of the K-value differential calculus. This K-value signal (K = 7) is examined within the framework of Windows with a duration of 17 cycles (Fig. 4). The input of the K-value ART-1K neuronal network is transformed into a vector with a 17×7 = 119 elements dimension.

Figure 4 shows a K-value signal as input on the block automated identification of signals.

In the real devices, the type of switching signals can be close to the expected result, but not fully concur with such change. In this connection, teaching of the ART-1K neural network executed on signals, where each of the signals are in the field, has three ranges in accordance with a seven-digit alphabet: [0-1], [2-4], [5, 6], as shown in Fig. 5.

After training of the neural network, testing was conducted on signals from the alphabet of Fantozi, each of which is included in the resulting range of deviations.

As a result of the experiments, the developed neural network correctly classifies 95% of signals produced.

CONCLUSIONS

Thus, based on the ART and discrete ART-1 neural network, a K-value neural network for the classification signals, which can appear during the simulation of the digital system based on the K-value differential calculus, has been developed.

The method of the ART-1K neural network, which is used to test computer technique operability, has been developed.

Testing of the developed neural network has been performed. The operability and possibility of applying the neural network in the simulation system based on the K-value differential calculus have been shown.

REFERENCES

1. Razevig, V.D., 2000. System of design digital devices OrCAD, M.: Solon, pp: 160.
2. Razevig, V.D., 1992. Application of the programs P-CAD and PSpice for a scheme design on PC. M.: Radio and svjaz.
3. Razevig, V.D., 1997. System for a scheme simulation MICRO-CAP 5. M.: Solon, pp: 152.
4. Gladkikh, T.V. and Leonov S.Yu., 2005. System of K-value simulation for research switching processes in digital devices. Herald of the National Technical University "KhPI". Subject issue: Information Science and Modelling. Kharkov: NTU "KhPI". 56: 175-187.
5. Korsunov, N.I., V.D. Dmitrienko and Leonov S.Yu., 1994. Numerical methods for solving K-value differential equitions. Engineering Simulation, 12 (1): 29-38.
6. Korsunov, N.I., V.D. Dmitrienko, S.Yu. Leonov and T.V. Gladkikh, 1998. Use of the technique of derivatives of K-valued functions for simulation of computing units. Engineering Simulation, 15 (2): 127-135.
7. Dmitrienko V.D., S.Yu. Leonov and T.V. Gladkikh, 2011. Use of neuron network on basis of K-value neurons for recognition of risks of failures. Herald of the National Technical University "KhPI". Subject issue: Information Science and Modelling.-Kharkov: NTU "KhPI", 36: 52-60.
8. Aktouf, C., 2002. A complete strategy for testing an on-chip multiprocessor architecture, IEEE Design & Test of Computers. 1: 18-28.
9. Rutkovsky, L., 2002. Methods and technologies of artificial intelligence, M.: Gorjachaja linija-Telecom, pp: 520.
10. Krug, P.G., 2002. Neuron networks and neurocomputers. M.: Publishing MEI, pp: 176.
11. Grossberg, S., 1987. Competitive learning: From interactive activation to adaptive resonance. Cognitive Science, 11: 13-25.
12. Noskov, V.I., V.D. Dmitrienko, N.I. Zapolovsky and S.Yu. Leonov, 2003. Design and optimization of control system and control of locomotives, Kharkov: KhFI "Transport of Ukraine", pp: 248.