An Approach to Restore the Proper Functioning of Embedded Systems Due to Cyber Threats

Serhii Toliupa [1], Serhii Shtanenko [2], Oleksiy Silko [2], Pavlo Khusainov [2] and Andrii Kulko [1]

[1] *Taras Shevchenko National University of Kyiv, Volodymyrs'ka str. 64/13, Kyiv, 01601, Ukraine*
[2] *Military Institute of Telecommunication and Information Technologies named after the Heroes of Kruty, Moskovska str. 45/1, Kyiv, 01011, Ukraine*

Abstract

The paper proposes an approach to restore the proper functioning of specialized microprocessor-based control systems at the element base level. This approach includes two stages. The first stage involves assessing the technical state of the microprocessor system by applying existing control methods in order to identify faults (cyber incidents and cyber attacks), as well as localizing faults (response to cyber incidents and cyber attacks) by applying methods of testing and functioning diagnostics of digital devices. At the second stage, the proper functioning of the microprocessor control system is restored by reconfiguring its internal structure at the level of logical elements. The implementation of the proposed approach will increase fault tolerance (cyber resilience) of the embedded system – the ability to maintain operability after failure of one or more of its components due to cyber threats.

Keywords[1]

Microprocessor system, programmable logic device, digital device diagnostics embedded system, digital device monitoring and diagnostics, reconfiguration, cyber resilience, cyber threats, proper function recovery, neural network.

1. Introduction

Modern control systems for various purposes (automated control systems (ACS), automated process control systems (APCS), automated organizational control systems (AOCS), etc.) contain computer hardware – processors, memory units, software, various types of converters, sensors, gauges, actuators [1]. These devices are often implemented by using a modern base – microprocessor sets and special devices on large/very large integrated circuits (LSI/VLSI), which are essentially embedded systems.

An embedded system is a specialized microprocessor-based monitoring and control system which design concept lies in its functioning by being embedded directly in the device it controls [2].

Embedded systems are now widely used in a variety of industries such as: machine building and machine-tool building, aviation, car industry, nuclear power industry, banking, and the military-industrial complex [3].

It should be noted that the first embedded systems were developed as specialized digital devices based on the integrated circuits of either small or medium integration. However, with the rise of microcontroller and microprocessor technology, and later integrated circuits with programmable structure, the concept of embedded system has been greatly transformed. Thus, while the first embedded systems represented a specialized structure with a central processor, separate integrated circuits for peripheral equipment controllers and digital memories, today's embedded systems are based on System-on-Chip (SoC) technology [4].

Information Technology and Implementation (IT&I-2023), November 20 - 21, 2023, Kyiv, Ukraine
EMAIL: tolupa@i.ua (A. 1); sh_sergei@ukr.net (A. 2); oleksiy.silko@viti.edu.ua (A. 3); indesys@ukr.net (A. 4); kulko.andrii@gmail.com (A. 5);
ORCID: 0000-0002-1919-9174 (A. 1); 0000-0001-9776-4653(A. 2); 0000-0001-6605-6945 (A. 3); 0000-0002-0675-0369 (A. 4); 0009-0006-1185-0774 (A. 5);

CEUR Workshop Proceedings (CEUR-WS.org)

System-on-Chip refers to a computing system implemented in an integrated design that includes a high-performance processor or several processors, mathematical processor for data processing and digital signal processing, additional memory modules, controllers, etc. Such organization of computing system is widespread due to its versatility, low power consumption as well as its possibility of reconfiguration of its algorithmic structure. It should be noted that systems-on-chip are now replacing bulky computing structures implemented with a set of integrated circuits by modern microcontrollers (PIC, AVR, MSP430, STM32, Cortex-M, TSP32 etc.), programmable logic device (PLD – CPLD, FPGA, FLEX) and Raspberry Pi type of single board computers [5].

Besides it should be taken into account that the creation of modern embedded systems, using System-on-Chip technology, is based on application of high-tech CAD systems of functional digital devices, which requires from its developers deep knowledge not only of digital circuitry and architecture of computing systems, but also knowledge of synthesis methods of special devices with microprogram control, knowledge of hardware description languages and program code development, and also methods of controllable synthesis.

2. Analysis of Recent Studies and Publications

Nowadays the design and functioning of embedded systems has been the subject of a large number of scientific papers. For example, in the research [6], the problem of improving the quality of microprocessors used in access control and management systems is considered. The requirements are allocated, and also the variant of structure of commands which are necessary for qualitative construction of microprocessors working on the basis of system of residual classes for the access control and management is offered. In [7] a review of embedded microprocessor systems design tools implemented on the basis of FPGA is given and software debugging tools for microprocessor systems based on Pico Blaze, Micro Blaze and Power PC cores are reviewed. In [8] questions of organization of hardware of embedded microprocessor systems are considered, and also synthesis of elements of embedded systems on programmable logic on the basis of model of programmable automata. Paper [9] reviews the issues of regularising embedded microprocessor systems as well as the synthesis of hardware component of embedded systems by means of variable logic with the program-controlled automaton model being used. The paper [10] presents a set of practical strategies for determining the first steps when deploying Model-Based Design and code generation in production development processes. The paper [11] examines the need to ensure prompt response to cyber incidents within a limited time frame and determines the improvement of the information decision-making model.

However, the analysis shows that nowadays the issues of assessing the technical state of embedded systems in terms of their proper functioning in the case of cyber threats, as well as immediate automatic recovery of the system by the results of self-diagnosis are not fully elaborated. In addition, it should be taken into account that the existing methods of control and diagnostics, as a rule, are developed for a particular type of integrated circuits, that is not always acceptable for use with respect to a particular type of circuits. Thus, the purpose of this article is to develop an approach to restore the proper functioning of a specialized microprocessor-based control system due to cyber threats at the level of the programmable element base according to the results of self-diagnostics.

3. Control and Diagnosis of Microprocessor Systems

The control of microprocessor (computer) systems is understood as the process of obtaining information to determine the technical condition of a computer system by applying hardware, software and combined methods and means of control and to establish its compliance with the requirements.

To assess the effectiveness of the control methods, the quality factor of the supervised computing system, defined as the probability of producing an error-free result of the information conversion, can be used

$$K(t) = 1 - P_{err},$$

where P_{err} is the probability of missing errors by the control system when issuing the result of information conversion in the computer system. At the same time, the main types of control can be classified: on the basic of the means used; on the basic of the nature of control; on the basic of the way of organization; on the basic of the object of control.

However, it should be kept in mind that these types of control are generally used in general-purpose computing systems. At the same time, in specialized computer systems executing a limited number of functional programs the control of program execution correctness called program-logic control is widely used: the control of program execution duration, the method of control functions and smoothness control [12].

The above-mentioned types of program execution control are mainly performed by software tools. They make it possible to detect errors in the operation of computer systems with a delay that is commensurate with the execution time of the program or subprogram. Software controls together with hardware controls help to detect errors that were not detected by the hardware controls. If the intended use of the computer system does not require the rapid detection of errors, software controls are sufficient.

Furthermore, diagnosis refers to the procedure of localizing the fault of an object, i.e. identifying which part of the object being diagnosed is faulty. Diagnostics involves locating the fault of an object at a lower hierarchical level than monitoring. In some cases, monitoring of computer systems is seen as a special case of diagnosis. By continuing down the hierarchical structure of the diagnosing object, it is possible to reach any desired level of the hierarchy, to individual contact connections, radio components or even parts of their construction. The measure of penetration through the object hierarchy is the depth of diagnosis. At the same time, the depth of diagnosis is decided on the basis of the organization of the recovery process. In order to restore the system quickly, it is advisable to limit the identification of the failed device first. This task is solved in most cases by means of hardware control, without involvement of software testing methods. In this case the control procedure can be considered as the procedure of diagnostics at the lowest depth. Let's consider the most common methods of diagnosing computer systems.

Today, a distinction is made between test diagnostics and functional diagnostics according to the nature of interaction between the object and the diagnostic tool [13]. In test diagnosis, specially prepared test influences are applied to the object and the object responses to these influences are compared with the reference responses. This type of diagnostics is used when it is necessary to check the serviceability of functioning or detect a fault (defect) affecting the performance of the tested object. In test diagnosis, special algorithms consisting of elementary control steps are implemented. The final diagnosis is made based on the results of the elementary control of the computer system. In this case, heuristic approaches, diagnostic models of analytical descriptions or graph-analytical representations of the main properties of the object and diagnostic algorithms developed on their basis as a set of sequential operations are used. It should be noted that test diagnosis methods contain very cumbersome and expensive preparatory operations to develop deterministic tests and reference reactions. Three types of testing are distinguished:

statistical, where the change of test sets on the output and the removal of responses is much lower than the frequency when the computing system is operating under real conditions;

dynamic, where test sets are given and output responses are analyzed at the limiting frequencies of the computing system;

parametric, when the parameters of the computing system are checked, both static – voltage, current, resistance, gain, and dynamic – changes in voltage, current, conductivity, gain, time delays, etc. The main methods of test diagnostics include: method of diagnosis at the level of logical circuits; method of diagnosis at the level of pluggable units; method of microdiagnosis; method of reference conditions; method of command core.

Functional diagnosis, in its turn, means processing information that characterizes the quality of functioning of the diagnosed object, when the parameters of performance of the computer system are determined for the performance of basic functions. Functional diagnostics can be performed either continuously or periodically or episodically.

It should be noted that the first microprocessor systems used functional control methods, i.e. control that lies in checking the performance of basic operational functions by the control object – microprocessor system or its part. During the further development of microprocessor systems, it turned out that the functional control is hampered by the "dimensionality barrier", because the number of functions carried out by the controlled object is too high to check them all. Therefore, the principle was proposed that it is not the functions of the microprocessor system that should be checked, but its elements (processor, memory, peripheral devices, etc.).

Currently, two main approaches are known in functional diagnosis – deterministic and probabilistic (stochastic). The first one uses deterministic model of the diagnosed system, the essence of which is generation (reading) of static and dynamic tests prepared manually or automatically, as well as the analysis of output and reference responses prepared in advance by special means. The second one is probabilistic (stochastic), which implies feeding of noise-like (random and pseudo-random) influences generated by inbuilt generators to an input of a computing system and analysis of output reactions. At the same time, the main methods of functional diagnostics include: diagnosis by means of the circuits of embedded control; diagnosis by means of the self-diagnostic dubbling; diagnosis by means of conditions registering.

Separate mention should be made of compact testing (signature analysis), which refers to both probabilistic control methods and test diagnosis methods. The essence of this method is to compare test results with a benchmark (compressed long bit sequence with high accuracy into short codes – signatures) [14]. This is done with the help of signature registers implementing the polynomial of bit sequence convolution with high accuracy. The resulting signatures are compared with the reference ones recorded in the signature dictionary implemented as a fault finding tree. And also competing with the signature analysis method is the spectrogram method that makes it possible to use distributions of relative frequencies of appearance of separate combinations formed by output symbols at consecutive points in time as diagnostic features. These distributions are called spectrograms of the computing device being diagnosed. The spectrogram can be obtained analytically or by simulating the operation of the device on a fixed input sequence.

Thus, the considered methods of control and diagnostics of technical means are the general methodological and technical basis for forecasting and diagnosing of failures, and this, in its turn, allows providing the required reliability and efficiency indicators of computer systems at the least expenses of forces and means.

4. A Neural Network Approach to Recognizing the Technical State of Microprocessor Systems.

Recently, according to [15], a fundamentally new approach to building recognition systems for the technical state of complex technical systems that function under conditions of incomplete, unclear and contradictory information has been gaining popularity, and this approach consists in the use of intelligent systems. In contrast to expert systems, which use the experience (intelligence) of specialists (experts), intelligent systems have the ability to learn and self-learn (use their own knowledge and experience).

So, the category of intelligent systems includes neural network systems that simulate the activity of the neural structures of the human brain by their structure and principles of functioning. These models present information through networks of interconnected nodes and are "self-processing" in the sense that they function without any external program, their nodes and connections are active processing elements. In addition, neural network systems represent global system behavior, which is due to the simultaneous local interactions that occur in parallel between multiple elements of the network. As a result of simultaneous local interactions between nodes in the system, signs of intelligence spontaneously emerge, which is one of the basic principles underlying the construction of neural network systems.

Given the above, according to [16], the solution of the problem of determining the technical condition of the microprocessor system can be represented as a search problem, where the desired solution is the goal of fault finding, and the set of possible ways to achieve the goal is a space of states, a set of branches of the decision tree. In the process of searching for solution in a decision tree, a certain number of vertices must be expanded and a certain amount of operations must be performed. The number of nodes to be disclosed in a search depends significantly on the method that determines the sequence in which they are disclosed. For the small spatial states the brute- force method is the simplest and the most reliable. However, for the large spatial states the brute- force method is unacceptable due to the increasing number of vertices in the decision tree. The reality of 'combinatorial explosion' arises, as there are no options to limit the diagnostic information. Applying simplification as a method of choosing a solution is not possible. However, simplification is known to be a tool of the human brain that quickly selects a subset suitable only for a particular situation from a huge variety of facts. The challenge, however, is to incorporate a simplification mechanism similar to that of the human brain. This is the task of artificial intelligence systems and needs to be solved on a neural network decision tree.

Decision search methods in artificial intelligence systems can be based on heuristic information, experience, common sense and intuition of the decision maker. In doing so, the discovery of the vertices of the decision search tree seeks to order the search process in such a way that it spreads in the most promising directions.

In addition, non-monotonic reasoning, i.e. common sense reasoning, is used in most cases to determine technical condition. Such reasoning is based on hypotheses where there is no information about their inconsistency. These hypotheses change when additional information is obtained, i.e. return procedures are possible in the decision tree. If the wrong search direction appears, a return to the state in which the wrong hypothesis was chosen takes place.

Given the uncertainty, a Bayesian approach is possible to calculate the probability of some hypothesis. The decision probability $P(S_i/S_2)$ is determined by the priori decision probabilities $P(S_2)$ $P(S_i)$ and the posterior probability $P(S_2/S_i)$:

$$P(S_i/S_2) = \frac{(S_2/S_i)P(S_i)}{P(S_2)},$$

provided there is no accompanying heuristic. Bayesian-based approaches are based on the assumption that for any solution there is a (albeit very small) a priori probability that it is true.

Considering that each type diagnostic object has its own image (portrait), the cluster space generates a neural ensemble with a statistical description of the cluster through a probable portrait (possible matrix) [17, 18]. Thus, the number of layers in the structure will be determined by the number of clusters.

The number of neural-like elements (neurons) in the layer is determined by the volume of the statistical sample (the number of features). At the same time, large statistical samples increase the dimension of the clusters represented by a portrait (set of parameters), while small ones do not allow unambiguously linking symptoms with a diagnosis. A portrait of a cluster will be optimal to allow obtaining the necessary amount of information. In this case, the size of the statistical sample will be determined by the number of parameters characterizing this type diagnostic object. Thus, the number of neural-like elements in the neural ensemble will be determined by the cluster portrait and the number diagnostic object parameters. A collection of neural ensembles (layers) is a neural network. Such neural networks are a simplified Markov model.

The set of clusters that need to be recognized and the influences acting on the information system (IS) can be represented in the form of dynamic discrete systems (DDS). Dynamic discrete systems can only be represented by stochastic ones in the form of logical, algebraic, or operation-oriented models. Mathematical apparatus for describing stochastic models focused on the functioning of DDS, selected on the basis of Markov fields. Such a mathematical representation allows one to describe the functioning of the DDS as an element of the external environment and to optimally map it to the structure of the deterministic part of the neural network.

The number of DDS states is determined by the accuracy of the piecewise constant approximation of the continuous phase trajectory of the dynamic portrait. Improving the accuracy of the piecewise constant approximation of the phase trajectory of a continuous system requires the introduction of a cluster space A of high dimensionality, which complicates the analytical description. The way out is the possibility of increasing (gluing) states, that is, the transition from the space of configurations $\Omega = \mathrm{A}^T$ to the space of states $\Omega_B = B^T$. New macrostates can be obtained by combining former states as follows:

$$B_k = \sum_{j \in \{K\}} A\,;\ k = 1, \ldots, p\,;\ j = 1, \ldots, r\,;\ p < r\,,$$

where $\{K\}$ – set of system states indices $\{A_j\}$ united in B_k .

The external environment for a neural network system (NNS) can be represented as a set of DDS recognition with associated discrete states.

The generalized model of a problem-oriented NNS has a structure that includes (Fig. 1): a sensory matrix that perceives the information Markov field in the form of a set of observations; a set of neural ensembles (classifiers), determined by the number of clusters M; a neural field that takes into account a priori information in the form of probabilities of hypotheses P; a neural field that takes into account the element values of the payment matrix C; majority network, which makes the decision G on recognition; subsystem (subnet) of training. Having a limited number of feature measurements obtained from IS, it is necessary to develop a procedure for processing parameters that allows to automatically obtain information about the state of the system.

Signs are perceived by the sensory matrix in the form of a set of observations: $X = (X_1, X_2, ..., X_i, ..., X_m)$, $i = 1, 2, ..., n$, $X_i = (X_{1i}, X_{2i}, ..., X_{ij}, ..., X_{pi})$.

In a separate sensory channel, the reduction of the sample space X occurs, as a result of which a sequence of discrete variables U_k, $k = 1, 2, ..., n-1$, take values $Z_1, Z_2, ..., Z_r$.

It is necessary to synthesize the structure of the Neuro-like classifier, which implements the decisive function $\gamma(U)$ on the reduced sample space U.

Sequence of discrete variables U_k, $k = 1, 2, ..., n-1$, which take values z_a, $a = 1, 2, ..., r$, can be approximated by vectors Ξ, $\Phi(0)_\mu$ and Ξ, $\Phi(k)_\mu$.

Using vector notation, we can write:

$$ln\, l_\mu = \left(\Xi, \Phi(0)_\mu\right) + \left(\Xi, \Phi(k)_\mu\right) = |\Xi|, \left|\Phi(0)_\mu\right| \cos\left(\Xi\,\hat{}\,\Phi(0)_\mu\right) + |\Xi|, \left|\Phi(k)_\mu\right| \cos\left(\Xi\,\hat{}\,\Phi(k)_\mu\right),$$

where $|\Xi|, \left|\Phi(0)_\mu\right|, \left|\Phi(k)_\mu\right|$ – vector modules Ξ, $\Phi(0)_\mu$, $\Phi(k)_\mu$; $\Xi\,\hat{}\,\Phi(0)_\mu$, $\Xi\,\hat{}\,\Phi(k)_\mu$ – angles between this vectors.

The above expression completely determines the optimal structure of the classifier for fixed j and i. It allows you to interpret the functioning of the synthesized structure.

Thus, the same excitation vector arrives at the input of each ensemble. Ensembles differ in the effectiveness of their connections. If the vector lengths for all ensembles are the same, then the magnitude of the excitation of the ensemble at constant Ξ will depend only on the angles between $\Xi\,\hat{}\,\Phi(0)_\mu$ and $\Xi\,\hat{}\,\Phi(k)_\mu$. This means that the most excitation is the ensemble whose vectors $\Phi(0)_\mu$ and $\Phi(k)_\mu$, are collinear to vector Ξ. The decision is made according to the number of the most excited ensemble.

The structure of the simplest neural-like system is a set of $M+1$ ensembles of neural networks of the first layer. The ensemble consists of n neurons, the level of excitation of which is defined as:

$$Y_\mu(k) = \sum_{a=1}^{r} \Xi_a(k)\, \Phi_a(k)_\mu \,.$$

Each neuron carries out the process coding, which is determined by the so-called method of labeled lines, in which a certain value of the process is provided in accordance with certain (labeled) lines $Z_1, Z_2, ..., Z_a, ..., Z_k$ and, therefore, a certain value of the process parameter is answered by one very excited synoptic connection $\Xi_a(k) = 1$.

In contrast to a typical neuron, whose synoptic connections are equivalent, in a neuron that encodes using the labeled lines method, synoptic connections have priority. The higher-numbered synoptic input corresponds to the higher value of the informative process parameter. Another difference is that at the k-th moment of time only one synoptic connection is excited and, thus, the task of introducing and controlling the threshold Ξ, is greatly simplified using the weight function *w.*

In fact:

$$Y_\mu(k) = \sum_{\alpha=1}^{r} \Xi_\alpha(k)\, \Phi_\alpha(k)_\mu - \Xi_\alpha(k)_\mu = \sum_{\alpha=1}^{r} \Xi_\alpha(k)\, \Phi_\alpha(k)_\mu \,. \qquad (1)$$

For $\Xi_{\alpha}(k)_{\mu} = 0$ the structure of the system, that recognizes, is quasilinear, and for $\Xi_{\alpha}(k)_{\mu} > 0$ it has nonlinear boundary properties.

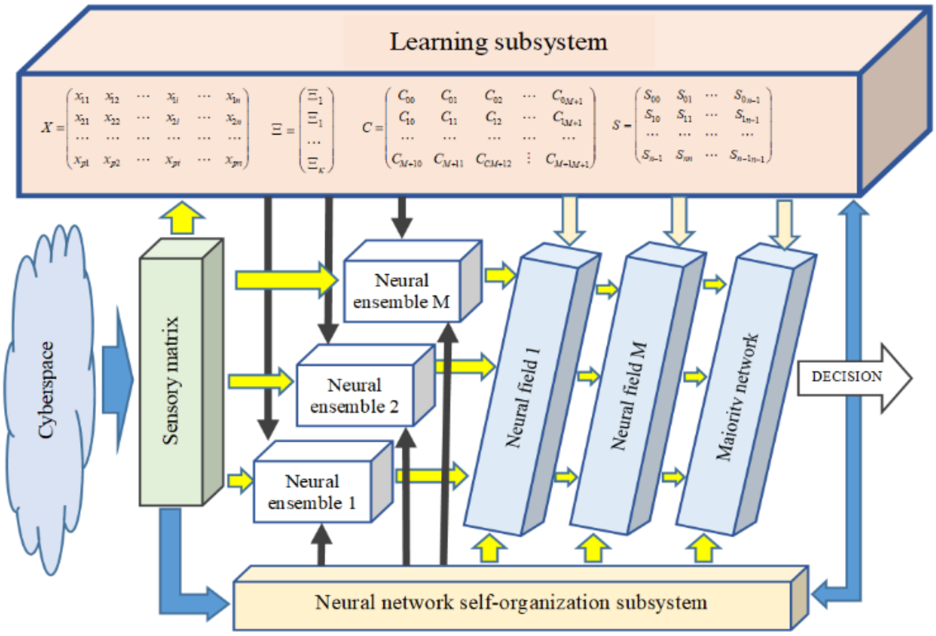

Figure 1: Generalized model of problem-oriented neural network for intrusion recognition

The second layer of neurons in the ensemble implements the operation

$$\ln l_{\mu}(k) = \sum_{k=0}^{n} Y_k(k)\,. \tag{2}$$

It is connected to the first layer by projection links, which establish a one-to-one correspondence between neurons of different fields, that is, they transfer state changes from one field to another.

The input information for the third layer of neurons is the vector $\ln L(U) = \left(\ln l_0(U), \ln l_1(U), \ldots, \ln l_m(U)\right)$. It acts as a majority logical device.

Expressions (1) and (2) completely define the structure of the system, which recognizes the state of the network. Let us introduce the matrix of connections of the k-th group of sensors with the k-th neuron, composing it from the indicators of excitation:

$$\Xi = \begin{pmatrix} \Xi_1 \\ \Xi_2 \\ \cdots \\ \Xi_{\kappa} \end{pmatrix}. \tag{3}$$

We also introduce the concept of the coefficient of interneuronal communication S in an ensemble (in a layer),

$$S_{kl} = \begin{cases} 1, \text{ if presents connection between the } k\text{-th group of neurons and } l\text{-th neuron;} \\ 0, \text{ otherwise.} \end{cases}$$

We form a matrix of interneuronal connections, composing it from the coefficients of interneuronal connections:

$$S = \left\{ \begin{matrix} S_{00} & S_{01} & \cdots & S_{0\,n-1} \\ S_{10} & S_{11} & \cdots & S_{1\,n-1} \\ \cdots & \cdots & \cdots & \cdots \\ S_{n-1} & S_{nn} & \cdots & S_{n-1\,n-1} \end{matrix} \right\}, \tag{4}$$

Matrices Ξ_k and S completely determine the structure of connections in the ensemble. When $Y = 0$ the matrix S is rearranged into a diagonal matrix with dimension $n \times n$.

The synthesized structures assume a fixed sample size, that is, the recognizing system observes the entire phase trajectory diagnostic object at once.

The information field is perceived by the sensor matrix in the form of a set of observations:

$$X = \left\{ \begin{matrix} x_{11} & x_{12} & \cdots & x_{1i} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2i} & \cdots & x_{2n} \\ \cdots & \cdots & \cdots & \cdots & \vdots & \cdots \\ x_{p1} & x_{p2} & \cdots & x_{pi} & \cdots & x_{pn} \end{matrix} \right\} \tag{5}$$

Each column $x_i = (x_{i1}, x_{j2}, \ldots, x_{pi})$, $i = 1, 2, \ldots, n$ and row $x_j = (x_{j1}, x_{i2}, \ldots, x_{ji})$, $j = 1, 2, \ldots, p$ of the matrix X is respectively n and p – dimensional vector processes.

Possible $M + 1$ hypothesis $N_0, H_1, ..., H_\mu, H_M$ on the ownership of the information field μ-th class, that is observed. The prior probabilities of hypotheses are known $P = P\{H_\mu\}, \mu = 0, 1, \ldots, M$. Also the payment matrix is known:

$$C = \left\{ \begin{matrix} C_{00} & C_{01} & C_{02} & \cdots & C_{0\,M+1} \\ C_{10} & C_{11} & C_{12} & \cdots & C_{1\,M+1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ C_{M+10} & C_{M+11} & C_{M+12} & \cdots & C_{M+1\,M+1} \end{matrix} \right\} \tag{6}$$

element C_{jm} is a solution for γ_μ, when the true hypothesis was H_j, $j = \mu = 0, 1, ..., M$. Decision space $G = (\gamma_0, \gamma_2, \ldots, \gamma_M)$ is made of $M + 1$ is made of γ_μ – decision to accept a hypothesis H_μ. The task of the recognition system is to accept one of the hypotheses and reject others based on the results of observation. The average risk in making a decision is determined as follows:

$$R = \sum_{j=0}^{m} \sum_{\mu=1}^{m} C_{j\mu} P_j \int_{G_\mu} w(x_1, x_2, ..., x_m) H_0 \Big/ X. \tag{7}$$

The minimum value of the average risk is achieved if in the area G_μ decision-making γ_μ recognition systems will assign points X in the sample space that satisfy the system of inequalities:

$$\sum_{j=1}^{m} (C_{ij} - C_{jm}) \frac{P_i w(x_1, x_2, ..., x_i, ..., x_m)/H_i}{P_i w(x_1, x_2, ..., x_i, ..., x_m)/H_0} \geq C_{0\mu} - C_{0j}. \tag{8}$$

If we introduce the vector of likelihood ratios $l(x) = \lfloor l_0(x), l_1(x), l_\mu(x), \ldots, l_m(x) \rfloor$,

where, $l_\mu = \dfrac{w(x_1, x_2, ..., x_i, ..., x_m)/H_i}{w(x_1, x_2, ..., x_i, ..., x_m)/H_0}$ then the system of inequalities (8) can be represented as:

$$\sum_{j=1}^{M} (C_{ij} - C_{jm}) \frac{P_i}{P_i} l_i(x) = C_{0\mu} - C_{0j}. \tag{9}$$

Vector $l(x)$ carries all the information about the hypotheses being tested and to make a decision about the observation results, it is enough to calculate the components M-the dimensional vector of the relationship of truthfulness. The problem of calculating $l(x)$ and making a decision in the structure of a neurorecognizable system (NRS) is solved by the classifier.

When solving any problems associated with recognizing technical condition, it is necessary to first assess the degree of compliance of the adopted parameters (portraits) with the reference ones, that is, to determine the decision criterion. The quantitative measure of conformity has to be chosen in different ways, in accordance with the nature of the research being carried out.

Erroneous decision in the operation of the network is expressed in the fact that a portrait of one object of diagnosis is practically removed and will be assigned to another cluster. If the error is a random event, then the correctness of the decision is naturally characterized by the probability of no error, that is, the probability of correct classification. If the error probability is denoted by, then the probability of the correct classification: since error and correct classification form a complete group of events. However, it should be noted that microprocessor technology is dynamic in operation, i.e. it is a complex diagnostic system and its technical state undergoes changes over time. These changes must be identified in order to prevent failure to fully perform its functions. This requires organization of monitoring and diagnostics, i.e. systematic recognition of the current state of the microprocessor technology, which can change under the influence of controlled and uncontrolled causes. Given that a change in the value of any parameter can be caused by a number of reasons, this makes it almost impossible to use any clear model that adequately describes all the diagnostic properties of the object as a whole. The disadvantages of the conventional graph representation is the impossibility of exhaustively describing by such a model the entire variety of diagnoses belonging to different classes of faults. The complexity of solving the diagnostic problem is further exacerbated by the probabilistic nature of the occurrence of faults, with statistical information often missing. Conventional recognition methods based on the use of a priori statistical data are therefore not applicable.

As a result of recent research in the field of the technical diagnostics, many authors [19, 20] are inclined to apply a fuzzy model of the control object, which allows building a description of the relationships between different combinations of symptoms and diagnoses based on a probabilistic relationship. The diagnostic model is represented by a neural network, rather than a graph, which allows a relatively small number of basic relationships between individual symptom and diagnoses to be described, in principle, making all other relationship for symptom combinations derived along the way computable. This makes it possible to streamline the diagnostic recognition system by avoiding low- informative rules that establish computational relationship, i.e., avoiding inefficient sprawl of the model. Next, consider the state recognition model, which can be represented as a set of subsystems (Fig. 2).

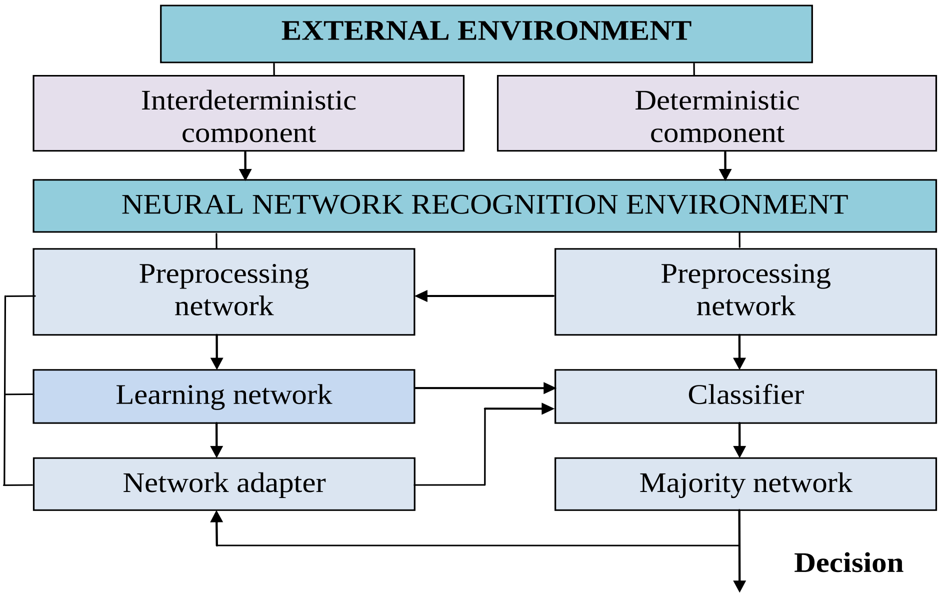

Figure 2: Generalized model of technical state recognition

The solution of the recognition problem, in general terms, makes it possible to determine ways and methods of solving the problem of increasing the efficiency of the recognizing diagnostic system. Let us consider the problem in more detail.

Let there is a set of diagnosis objects, $W = \{\omega_i\}\ i = 1, \dots, \rho$, on which there exists a partition into a finite number of subsets called technical diagnoses (classes), $L = \{A_\mu\}\ \mu = 1, \dots, M$. The set

$L=\bigcup\limits_{\mu=1}^{M} A_\mu$, referred to as the space of technical states, in the general case, is not fully defined, only some a priori information $J(M)$ about it is given, i.e. the number of diagnoses M is unknown. Diagnostic objects W_i are represented by a set of values of some measurements that constitute the diagnostic portrait of the recognition object. The set of N values of the features that determine the dictionary of diagnostic features, $X=\{x_j\}\ j=1,...,N$, according to which the recognition is actually performed, can be found by means of transformation (preprocessing) $J(\omega_i)$, i.e. $x_j=J(\omega_i)\,\omega_i$. Measurements (observations) of diagnostic objects W_i involve significant destabilizing factors (cyber threats), therefore, the signs of recognized diagnostic objects and their portraits will be probabilistic, which can be accounted for by the probability density function (PDF) $f(x)$ – a mixture of distributions of signs in all classes:

$$f(x)=\sum_{\mu=1}^{M} f(A_\mu)(x/A_\mu),$$

where $f(A_\mu)$ is the PDF of the occurrence of the μ class; $f(x/A_\mu)$ is the PDF of the conditional probabilities of the features x_i when the μ class occurs.

Note that spatio-temporal changes in the parameters of diagnostic objects ω_i require taking into account the dynamics of change $f(x)$ as a temporal process, which reflects the dynamism of the diagnostic environment. In the process of training, a set of random mappings of objects $P(x)$ of the external environment are transformed into the so-called probabilistic diagnostic portrait of the object $P(x)$

$$P(x)=Y_{lear}(x)x,$$

where $Y_{lear}(x)$ is the operator of the learning subsystem in recognizing system defining its purpose and function.

The probabilistic diagnostic profile acts as a generalized probabilistic reference that is formed during the training process and used in solving the technical condition recognition task.

Thus, the task of diagnostic recognition is to decide for a given diagnostic object W_i, the alphabet of technical diagnoses, $L=\{A_\mu\}\ \mu=1,...,M$ (or a priori information $J(M)$) and the dictionary of diagnostic attributes, $X=\{x_j\}\ j=1,...,N$, on the basis of the obtained description X_j and its probabilistic portrait $P(x)$, to decide whether the diagnostic object ω_i belongs to one of the diagnoses, that is A_μ

$$H(W_i/A_\mu)\to W_i\in A_\mu,\ \mu=1,...,M.$$

The possibility of dynamic changes in the structure (composition) of the external environment, i.e. $\omega_i(t)\in W(t)\to \mathrm{var}$, leads to inconsistencies in the values of ω_i and A_μ, which shows the inconsistency of information in recognition.

The basic assumption is that the recognizing system and the external environment are considered as a single anthropogenic system. This is the reason for the adequacy of the properties of the recognizing system, reflecting the reciprocal relationships of the elements of its structure.

As it is known, from the point of view of system analysis, the effectiveness of the recognizing system depends on the parameters (diagnostic attributes) of the external environment, $X=\{x_j\}$ $j=1,...,N$ and the parameters of the structure of the system itself, $S=\{S_k\}\ k=1,...,d$ with the parameters of the structure of the recognizing system characterizing both the elements of the structure and the links between them.

Consequently, the efficiency of a recognizing system is generally evaluated by its functioning:

$$E=E\left[\{x_j\}\ j=1,...,N;\ \{S_k\},\ k=1,...,d\right],$$

and the solution of the efficiency problem is reduced to finding its extremum under the constraints of the costs associated with obtaining the alphabet of diagnoses L, measuring and processing the dictionary of diagnostic features X of the mathematical functioning and hardware implementation of the recognizing system (r_0), that is:

$$E_{\max} = \max_{\substack{x,\,s\\ j,\,k}} E\left[\{x_j\}\, j=1,\dots,N;\ \{S_k\},\ k=1,\dots,d\right]$$

when providing $C_{r_0} \le C_{r_{add}}$.

In this case, each element of the diagnostic recognition system can be represented by some multipole with known input-output relationship, so let us represent a generalized model of the recognition system as a set of subsystems, the functioning of each of which corresponds to a deterministic or stochastic operator (Fig. 3).

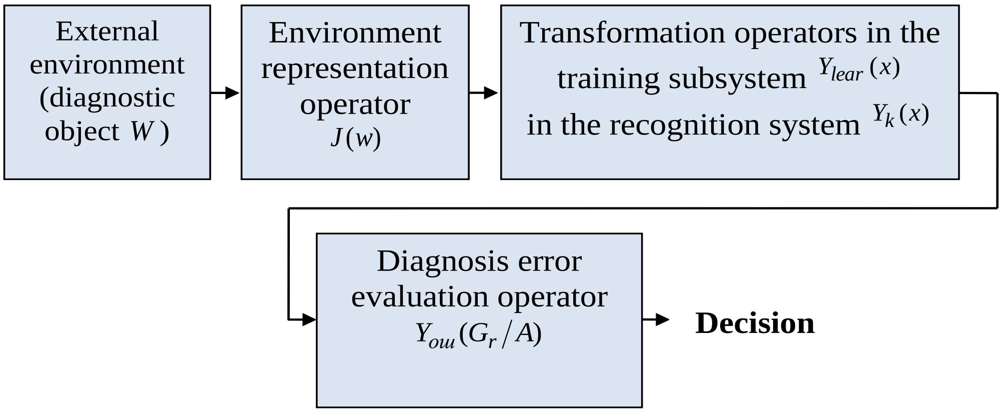

Figure 3: Generalized model of recognition of technical condition in the form of a set of subsystems

The solution of the problem is implemented on the principles of the system approach, experimental-theoretical research methods and is based on the application of neural network theory, static decision theory, random Markov field theory, and cluster analysis theory.

Thus, the presented neural network approach for the technical state recognition of microprocessor-based systems can act as a competitor for embedded control and diagnostics systems of complex technical systems, while creating an optimal space of technical states of the external environment, necessary for rapid technical state determination in real time.

5. Method for Reconfiguring Microprocessor Systems With Self-Diagnostics.

The microprocessor system as an object of diagnostics is a complex functional structure, which contains a large number of electronic elements and many branching connections. Besides, a great variety of microprocessors differ from each other by the set and the order of execution of commands, time organization of work, have different productivity, clock frequency, bit depth, cache-memory volume and micro- and macro-architecture. Proceeding from this, according to [21] decomposition approach is used when organizing control and also test and functional diagnostics of microprocessor systems, at which separate functional devices act as an object of control and diagnostics: arithmetic-logic device, processor, operating-memory device, permanent-memory device, input-output devices. It should be taken into account the difficulties arising in the control and diagnosis of microprocessor systems, which are associated with a high degree of integration of LSI/VLSI, ramified relationship between the elements, the lack of complete information about the internal structure of the microprocessor system, and the lack of hardware embedded control of the processor. And given the fact that one of the most promising trends in the design of microprocessor systems is technology System-on-Chip, which uses as an element base microcontrollers, integrated circuits with programmable structure and single-board computers such as Raspberry Pi, the problem of control and diagnosis acquires a completely new nature.

Thus, in the paper [13, 22] to assess the technical condition of microprocessor systems implemented on integrated circuits with programmable structure, it is proposed to use self-diagnostic tools of digital devices, while implementing the principle of interaction (testing) of microprocessors with each other by introducing a service processor into a multiprocessor system. The main function of such a processor is to monitor and diagnose the multiprocessor system, as well as rapid automatic recovery by reconfiguring the system. Realization of this principle and introduction of means of self-diagnostics will endow the microprocessor system with adaptation property, i.e. with ability to change parameters, structure, controlling actions in order to achieve optimum system functioning under initial uncertainty and changing conditions of work.

In addition, it is assumed that adaptation of the microprocessor system to the changing operating conditions will take place through reconfiguration of its internal structure, at the level of the logical elements of the programmable integrated circuit.

The method of reconfiguring the structure of digital devices (components of a microprocessor system) by changing the internal links between logical elements when a corresponding signal from self-diagnostic tools appears is presented below. The reconfiguration method is based on the consideration of digital devices as dynamic control systems, subject to external perturbations. To compensate the action of external perturbations on the proper functioning of such systems, the provision of prescriptive theory, which considers issues of purposeful control of objects of different nature that are in a state of "conflict" with other objects, is used [23].

The essence of this method is to find such a redundancy of the structure B_i, which, when connected to the input of the module A (with the faulty node A_j disconnected), leads to restoration of proper functioning of the module A (Fig. 4).

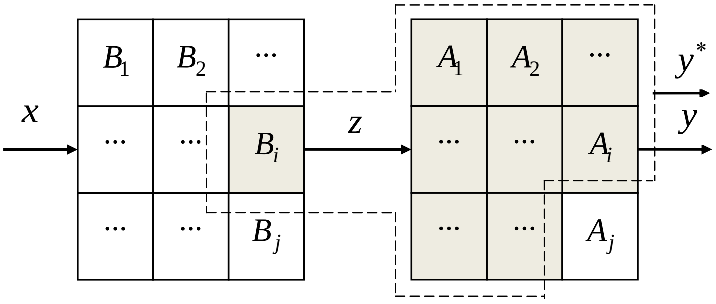

Figure 4: Structure of the restored module A

The module under consideration, A, is a set of logical elements a_i, implementing the function $Y_{n-1} = \varphi_0(X_n, Y_n)$, where $X(x_1, x_2, ..., x_n)$ is the input word, $Y(y_1, y_2, ..., y_n)$ is the output word, n is the time clock.

It is required to synthesize some system $S \supseteq A$, also realizing a given function $\varphi_0(X_n, Y_n)$, under the condition of failure of any of the subsystems A_j of the system $A(a_i \in A)$ of a given partition complexity C_j. The system A can be represented as a matrix M_A of any of its subsystems A_j. Removing any of its subsystems from the matrix M_A results in a distortion matrix $M\{A_{0j}\}$, $A_{0j} = (A_0 \setminus A_i)$, resulting in a set of new functions $M\{\varphi_{0j}(X_n, Y_n)\}$.

In order to restore the proper functioning of the system (realization of the function Y) it is necessary to form a restoring matrix $\{M_{B_j}\}$. In doing so, each subsystem B_j of the matrix $\{M_{B_j}\}$ is connected to the input of the subsystem A_{0j} of the distortion matrix $M\{\varphi_{0j}(X_n, Y_n)\}$. In general case for all B_i there is usually an intersection of structures

$$\bigcap_i B_i = B_1 B_2 \ldots B_j \ ,$$

possessing functional properties common to all $B_1, B_2 ..., B_j$ or most of them. But there may also be individual structures which do not contain any overlaps. In this case, for each fault and disconnectable subsystem A_j the formation of a redundant structure B_i based on the generalized module $\bigcap\limits_i B_i$ is formed by appropriate connection of input $\{X\}$ and output $\{Z\}$ signals of this module (Fig. 5).

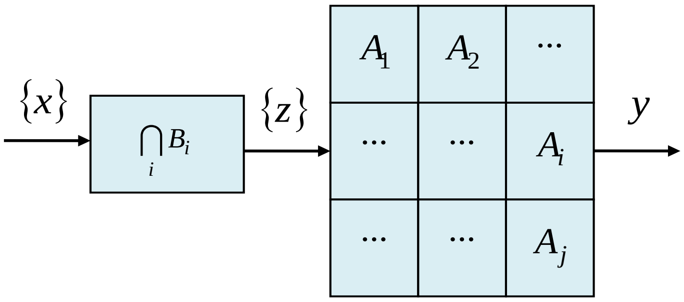

Figure 5: Mechanism of restoring the module

The content of the functionally reliable synthesis of the considered system A_0 is the definition of the rule Ψ of the description of the subsystem B_j of the matrix $M\{B_j\}$ at the removal of any subsystem A_j of the matrix M_A. Note that the rule Ψ should induce as restoring subsystems B_j by A_{0j}: $B_j = \Psi A_{0j}$, as well as the restoring matrix $M\{B_j\}$ distortion submatrix: $M\{B_j\} = \Psi\{M_{Aj}\}$.

Thus, the considered method of reconfiguration of digital devices allows to determine the structure of redundant subsystems B_j depending on disconnected faulty subsystems A_j of the given device A_0. In this case the calculated structure B_j and the remaining serviceable part of the device $A_{0j} = A_0 \setminus A_j$ implements the given function Y_{n+1}.

Consider the problem of formalizing the induction rule Ψ. The subsystem A_{0j} of the distortion matrix $M\{A_{0j}\}$ is a part of the system A_0 and is described by a function, like $B_j = \Psi A_{0j}$. To transform the distortion function φ_{0j} into the function φ_0, a new subsystem B_j is required , the composition of which with the subsystem A_{0j} with respect to θ_j forms the system $A_0 = \{B_j \Theta_j A_{0j}\}$.

The relation Θ_j specifies the cohesion operator of the systems A_{0j} and B_j or equivalence relation between the subset of the outputs $Z \subseteq X_{0j}$ of the subsystem B_j and the subset X_{0j} of the subsystem A_{0j}. By definition Θ_j specifies the functional relationship between indexes i of outputs of subsystem B_j and indexes k of inputs of subsystem A_{0j}: $i = \Theta_j(\alpha, \beta, k)$,

where α is the parameter specifying the number of inputs A_{0j} used by the subsystem B_j; β is the parameter specifying the relationship between inputs and outputs depending on α.

According to the above, the output $Z = B_j X_n$ of the restoring subsystem B_j implements the function $Z = f_j(X_n, Y_n)$.

To determine B_j or f_j according to $M\{B_j\} = \Psi\{M_{Aj}\}$ we get the expression: $\varphi_{0j}(X, Y, Z) = \varphi_0(X_n, Y_n)$, which functionally represents the right-hand side of the ratio $A_0 = \{B_j \Theta_j A_{0j}\}$.

Then the equation $\varphi_{0j}(\Theta, X, Y, Z) = \varphi_0(X_n, Y_n)$ will define Z, that is will describe the subsystem $Z = \Psi(X_n, Y_n, \Theta, j)$.

Thus, the relation $Z = \Psi(X_n, Y_n, \Theta, j)$ function Ψ sets the rule for inducing the restoring subsystems B_j for all $j \in J$.

It should be noted that when reconfiguring digital devices, a control (diagnostic) device is mandatory. Considering the fact that its structure and the functional tasks it performs are rather complex, it is advisable to develop a self-diagnostic device [24]. The principle of construction of such devices can be based on the method of reconfiguration of redundant digital devices. In this case the digital device A is also divided into nodes $A_1, A_2, ..., A_j$ and depending on this division the structure of the redundant device B, consisting of circuits $B_1, B_2, ..., B_j$. The self-diagnostic and self-reconfiguring device is shown in Fig. 6. It contains a reconfigurator R, which provides the appropriate reconfiguration of the devices A and B, and two control registers recording the results of calculations P_1 and P_2.

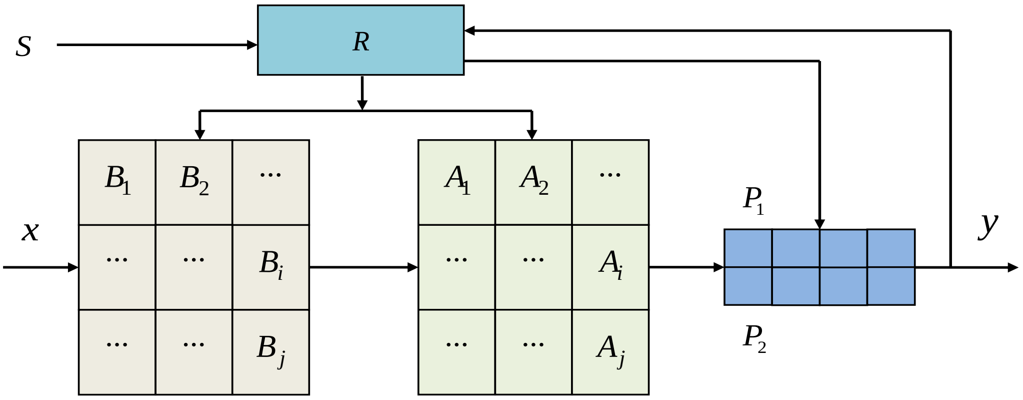

Figure 6: Structure of the redundant digital device with self-diagnosis

The principle of operation is that a digital device A, consisting of nodes $A_1, A_2, ..., A_j$ can be divided into three enlarged blocks A_I, A_{II}, A_{III} (Fig. 7, a). Then, in the absence of malfunctions, the entire device A operates. After a certain calculation step, a self-diagnostic signal S is sent to the reconfigurator R. In this case a test program is entered into the device and an intermediate calculation result (first step) is written into the first register P_1. Before the second calculation step, the reconfigurator R disconnects a part of the device, e.g. A_1 and connects the device B , e.g. B_I to it respectively. The same test program as in the first calculation step is applied to the input of the resulting device $B_I - A_{II} - A_{III}$ and the result of the calculation is written to the second register P_2 (second step). The reconfiguration of this type of device is shown in Fig. 7, b.

If the contents of registers P_1 and P_2 are the same, the device A will continue operating. If the results of calculations in P_1 and P_2 are different, it means that one of the units of the device A is defective. In this case one of the test diagnostic methods can be used as a signal S. Thus, the considered method of self-diagnostic reconfiguration makes it possible not only to assess the technical condition of a digital device, but also to restore its proper functioning due to cyber threats by rebuilding the internal structure.

6. Conclusion

An approach to restore of proper functioning of digital devices, as components of the specialized microprocessor-based control system, implemented on "System-on-a-Chip" technology is proposed. This approach is based on methods of control and diagnosis of computing systems and on method of reconfiguration of digital redundant structures with self-diagnostic means at the level of Boolean equations. The implementation of this approach in the design of embedded systems at the level of programmable logic will make it possible to increase the reliability (cyber resistance) not only of the specialized microprocessor control system, but also of the entire control system of complex objects and technological processes as a whole.

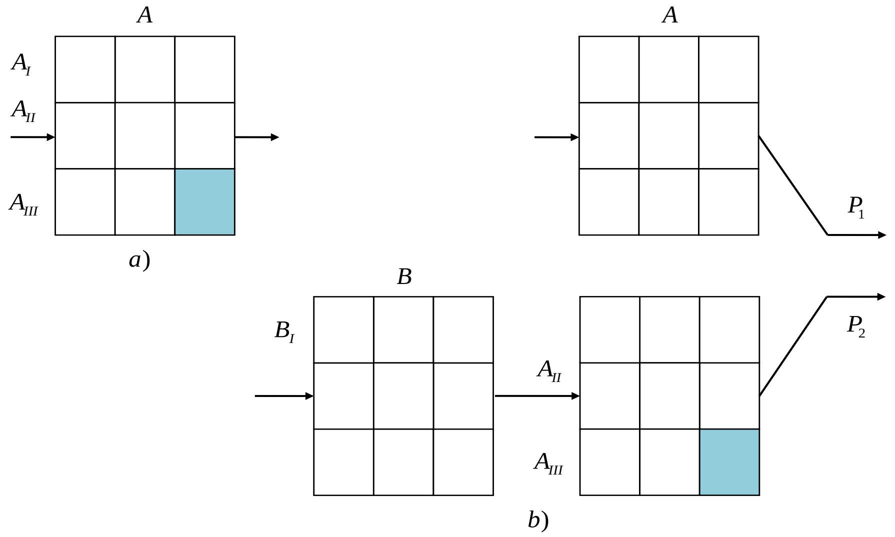

Figure 7: The principle of reconfiguration of the digital device with self-diagnosis: *a* – enlarged digital device A; *b* – digital device *A* with the redundant structure B

The focus for the future work is on the design of an adaptive microprocessor-based control system with integrated intelligent condition detection and a system for rapid, automatic restoration of correct operation, capable of counteracting adverse influences, both intentional and unintentional.

7. References

[1] Kodkin, V.; Baldenkov, A.; Anikin, A. A Method for Assessing the Stability of Digital Automatic Control Systems (ACS) with Discrete Elements. Hypothesis and Simulation Results. Energies 2021, 14, 6561. https://doi.org/10.3390/en14206561.

[2] A. Crespo, P. Albertos, J. Simó, Embedded control systems: from design to implementation, Ifac Proceedings Volumes, Volume 40, Issue 1, 2007, Pages 25-32, ISSN: 1474-6670, ISBN: 9783902661210, https://doi.org/10.3182/20070213-3-CU-2913.00006.

[3] H. De Man, System-on-Chip design: impact on education and research, in IEEE Design & Test of Computers, vol. 16, no. 3, pp. 11-19, July – Sept. 1999. https://doi.org/10.1109/54.785820.

[4] Chen, Yen-Kuang & Kung, Sun-Yuan. (2005). Trends and challenges with System-on-Chip technology for multimedia system design. 2005. 4 pp.. 10.1109/EITC.2005.1544365.

[5] S. Shtanenko, Y. Samokhvalov, S. Toliupa, O. Silko, Increasing Survivability of Technological Systems Based on the Technology of Programmable Logic Device, Conference: Information Technology and Implementation (IT&I-2021), December 01-03, 2021, Kyiv, Ukraine At: CEUR Workshop Proceedings (Vol-3132), 2022, pp. 237-245.

[6] Khan FH, Pasha MA, Masud S. Advancements in Microprocessor Architecture for Ubiquitous AI-An Overview on History, Evolution, and Upcoming Challenges in AI Implementation. Micromachines (Basel). 2021 Jun 6;12(6):665. https://doi.org/10.3390/mi12060665. PMID: 34204065; PMCID: PMC8227299.

[7] Furber S. Microprocessors: the engines of the digital age. Proc Math Phys Eng Sci. 2017 Mar; 473(2199):20160893. https://doi.org/10.1098/rspa.2016.0893. Epub 2017 Mar 15. Erratum in: Proc Math Phys Eng Sci. 2017 May; 473(2201):20170304. PMID: 28413353; PMCID: PMC5378251.

[8] Ma L, Xia F, Peng Z. Integrated Design and Implementation of Embedded Control Systems with Scilab. Sensors (Basel). 2008 Sep 5; 8(9):5501-5515. https://doi.org/10.3390/s8095501. PMID: 27873827; PMCID: PMC3705517.

[9] Wehrmeister, Marco & Becker, Leandro & Pereira, Carlos. (2005). An approach for designing real-time embedded systems from RT-UML specifications. 16. 1060-1060.

[10] Dillaber, E., Kendrick, L., Jin, W., and Reddy, V., Pragmatic Strategies for Adopting Model-Based Design for Embedded Applications, SAE Technical Paper 2010-01-0935, 2010, https://doi.org/10.4271/2010-01-0935.

[11] P. Khusainov, S. Toliupa and S. Shtanenko, "A Secure State of the OSI-Based Distributed System in the Context of Evaluating Cyber Security," 2022 IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), Kharkiv, Ukraine, 2022, pp. 33-37, doi: 10.1109/PICST57299.2022.10238670.

[12] Hristov, Hristo & Hristova, Mariya. (2018). Computer Control Systems With Critical Safety Applications: Problems And Some Solutions. JITA – Journal of Information Technology and Applications (Banja Luka) – APEIRON. 14. 10.7251/JIT1702061H.

[13] Shtanenko, S., Samokhvalov, Y., Toliupa, S., Silko, O. (2023). The Approach to Assessment of Technical Condition of Microprocessor Systems that Are Implemented on Integrated Circuits with a Programmable Structure. In: Klymash, M., Luntovskyy, A., Beshley, M., Melnyk, I., Schill, A. (eds) Emerging Networking in the Digital Transformation Age. TCSET 2022. Lecture Notes in Electrical Engineering, vol 965. Springer, Cham. https://doi.org/10.1007/978-3-031-24963-1_28.

[14] Argüelles, J., Bracho, S. Signature analysis for fault detection of mixed-signal ICs based on dynamic power-supply current. J Electron Test 9, 89-107 (1996). https://doi.org/10.1007/BF00137567.

[15] Romanuke, V. Optimal construction of the pattern matrix for probabilistic neural networks in technical diagnostics based on expert estimations / Romanuke Vadim // Information, Computing and Intelligent systems. 2021. No. 2. Pp. 19-25. Bibliogr.: 13 ref. https://doi.org/10.20535/2708-4930.2.2021.244186.

[16] Qi B, Zhang L, Liang J and Tong J (2022) Combinatorial Techniques for Fault Diagnosis in Nuclear Power Plants Based on Bayesian Neural Network and Simplified Bayesian Network-Artificial Neural Network. Front. Energy Res. 10:920194. https://doi.org/10.3389/fenrg.2022.920194.

[17] S. Toliupa, I. Tereikovskiy, I. Dychka, L. Tereikovska and A. Trush, "The Method of Using Production Rules in Neural Network Recognition of Emotions by Facial Geometry," 2019 3rd International Conference on Advanced Information and Communications Technologies (AICT), Lviv, Ukraine, 2019, pp. 323-327, doi: 10.1109/AIACT.2019.8847847.

[18] Kutyrev, A., Kiktev, N., Kalivoshko, O., Rakhmedov, R. (2022). Recognition and Classification Apple Fruits Based on a Convolutional Neural Network Model. Conference "Information Technology and Implementation" (IT&I-2022), Kyiv, Ukraine, November 30 – December 02, 2022. CEUR Workshop Proceedings, 3347, 90-101.

[19] Abdellatif Seghiour, Hamou Ait Abbas, Aissa Chouder, Abdlhamid Rabhi, Deep learning method based on autoencoder neural network applied to faults detection and diagnosis of photovoltaic system, Simulation Modelling Practice and Theory, Volume 123, 2023, 102704, ISSN 1569-190X, https://doi.org/10.1016/j.simpat.2022.102704.

[20] Samokhvalov, Y. (2021). Risk Assessment of Innovative Projects Based on Fuzzy Modeling. In: Babichev, S., Lytvynenko, V., Wójcik, W., Vyshemyrskaya, S. (eds) Lecture Notes in Computational Intelligence and Decision Making. ISDMCI 2020. Advances in Intelligent Systems and Computing, vol 1246. Springer, Cham.

[21] Liqiong Xu, Shuming Zhou, Jiafei Liu, Shanshan Yin, Reliability measure of multiprocessor system based on enhanced hypercubes, Discrete Applied Mathematics, Volume 289, 2021, Pages 125-138, ISSN 0166-218X, https://doi.org/10.1016/j.dam.2020.09.012.

[22] Oliveira, Domingos F. & Gomes, João & Pereira, Ricardo & Brito, Miguel & Machado, Ricardo-J. (2022). Development of a Self-Diagnostic System Integrated into a Cyber-Physical System. Computers. 11. 131. 10.3390/computers11090131.

[23] V.Ye. Obukhov and V.V. Pavlov, Synthesis of Redundant Discrete Systems with Structure Reconfiguration, Naukova Dumka, Kiev (1979), p. 156.

[24] S. Shtanenko, Y. Samokhvalov, O. Silko. A Methodical Approach to Designing Secure Adaptive Embedded Systems Implemented Via Reconfiguring their Structures,"Conference: Information Technology and Implementation", Workshop Proceedings (IT&I-WS 2022), Kyiv, Ukraine, November 30 – December 02, 2022. At: CEUR Workshop Proceedings (Vol-3384), 2022, pp. 97-109.