Models of Distributed Systems Testing Based on Energy Consumption in Behavior

Oleksandr Martynyuk [a], Lyudmila Sugak [b], Hanna Stepova [c], Victor Antoniuk [d], Igor Mileiko [e]

[a] *Odessa National Polytechnic University, Ave. Shevchenko 1, 65044 Odessa, Ukraine*

Abstract

Deviations from the standard power consumption often indicate malfunctions of technical systems. Continuous and periodic, instant and cumulative special analysis of energy consumption in the behavior of distributed information systems (DIS), compared with the features of the procedures they execute, can be an essential part of the means of online and offline testing in operability of DIS. This paper presents a model of behavioral testing and recognition of DIS, including a formal analysis of energy consumption. The model is based on behavioral experiments of an automata class and Petri nets, representing a reference DIS with added characteristics of energy consumption. The model has the features of behavioral individual and group energy characteristics, cross-sections-states, counters, spatial and temporal structures of energy consumption. The individuality of the characteristics presupposes the possibility of identifying the states of the reference model by its energy consumption, section-states are the subsets of its attainable states/positions identified in the reference model with characteristic (identifying) energy costs, counters are moving chips, that register energy consumption, as they move, and finally, energy consumption structures supposed and achievable topological structures of the movement of counters in the graph of the behavioral model.The model is designed to determine the energy conditions, used in the construction of methods of online and offline behavioral testing of the DIS performance, extended by the analysis of energy consumption.

Keywords

Energy consumption, behavioral testing, energy characteristics and identifiers, energy consumption patterns, energy consumption check

1. Introduction

Energy consumption analysis plays a significant role in monitoring the performance of discrete systems (DS) [1-3]. Deviations upward or downward from the respectively upper or lower boundary statistical energy consumption [4-6] may indicate errors, failures, in the operation of the hardware and software of the DS [7-9], incorrect actions, including unauthorized access and viral attacks [10, 11]. It is known, that the main power consumption in the DS equipment occurs during state switching [12, 13]. Accordingly, the minimum, average and maximum reference values of energy consumption for various elementary indivisible commands and micro-operations, executed by them can be determined for hardware elements and nodes of the DS [14-16]. These values can be used, as the basis for defining energy state diagrams and their cross-sections (vectors) for firmware of DS equipment, and on their basis - more general state diagrams and their sections for programs and scenarios of DS operation. The behavioral nature of microprograms, programs and scenarios of the DS makes it possible to form quite complex spatial-temporal structures of energy state diagrams, their sections and

IntelITSIS'2021: 2nd International Workshop on Intelligent Information Technologies and Systems of Information Security, March 24–26, 2021, Khmelnytskyi, Ukraine
EMAIL: anmartynyuk@ukr.net (O. Martynyuk); lpsugak@ukr.net (L. Sugak); hanna.suhak@gmail.com (H. Stepova); antoniuk@opu.ua (V. Antoniuk); mig3@ukr.net (I. Mileiko)
ORCID: 0000-0003-2366-1920 (O. Martynyuk); 0000-0001-5624-7633 (L. Sugak); 0000-0002-7223-8822 (H. Stepova); 0000-0003-0427-9005 (V. Antoniuk); 0000-0002-8664-7037 (I. Mileiko)

traces, correlated with the corresponding behavior structures of the microprograms, programs and scenarios themselves and their derivatives. Accordingly, behavioral analysis, offline and online testing acquires additional verification capabilities, based on the analysis of events and actions of the DS together with the analysis of their energy consumption. In the case of an implicit assignment of the class of properties to be checked, as deviations from the reference values, such behavioral analysis, offline and online testing can perform research and recognition of such reference values, both of the events and actions themselves, and their energy consumption.

2. Purpose and objectives of the article research

The purpose of this work is to increase the completeness of the behavioral offline and online testing of DIS through additional monitoring and analysis of energy consumption in the process of their main and test functioning. To achieve the goal, the following tasks have been solved: firstly, studies of known approaches to the analysis of energy consumption for structural and functional models of DS; second, the development of a decomposition model for analyzing energy costs, based on an extended Petri net, which has the features of the spatial and temporal structures of localized and moving energy variables and chips, which determines the conditions for the use of energy costs in behavioral analysis, check and recognition of DIS; third, the construction of the main steps of procedures for network and behavioral energy analysis, check and recognition of DIS; fourth, testing these basic steps for the DIS components.

3. Related works

As you know, the main for the high power consumption of existing equipment are dynamic modes of operation associated with current transient switching processes between steady states, accumulation and consumption of electrical energy [5, 6]. A significant part of this energy is released in the form of heat contact (on various radiators), air-convection and in the form of infrared radiation, and its value is directly proportional to the average values of the electric currents. That is, the power consumption of the DS equipment can be analytically estimated using special formal switching models [7-9], explicitly determined using differential and integral current measurements, integral measurements of temperature and infrared radiation [5, 6].

Asynchronous, synchronous, clocked single-cycle and multi-cycle devices, both without memory and with memory, allow you to define variables, their sets, commands, micro-operations, instructions, operations and functions based on them, for which power consumption is essential, as noted above, transient processes of all switching, associated with both input variables, synchronizations, parameters, data/operands, and internal/output variables, data, results. Depending on the manufacturing technologies for the hardware of the DS, for example, FPGA [17-19], their modes of operation, for example, critical [20-22], approximation of calculations [23-25], circuit solutions, bit width, coding, values of commands and operands, it is possible to determine these switches, their approximate minimum, average maximum numbers with corresponding values of power consumption [5, 6].

Moreover, for single-cycle devices or devices with low-cycle commands, micro-operations, instructions, operations and functions, the number of switching transients associated with internal/output variables, data, results are directly proportional to the number of switching transients, associated with input variables, synchronizations, parameters, data/operands [7-9].

This circumstance makes it possible in this case to simplify the analysis of energy consumption, reducing it to an analysis of the numbers of all input switching.

For multi-cycle devices, devices with multi-cycle instructions, micro-operations, instructions, operations and functions, programs and scenarios for the operation of the DS hardware, considered indivisibly, the transient switching processes are determined to a greater extent by the sequential logic of their execution, than by the indicated input switches [17, 21].

When analyzing the energy consumption of multi-cycle devices, a time decomposition behavioral approach is advisable, based on: firstly, dividing/splitting multi-cycle instructions, micro-operations, instructions, operations and functions, programs and scripts into simpler low-cycle micro-operations instructions, instructions, operations and functions; secondly, their temporal sequential behavioral representation in the form of graph structures, in particular, basic ones – chains, trees, hammocks, feedback loops, with standard procedures for determining their energy consumption, reduced to the total (reduced) minimum, average maximum reference values of input switching for one- or few-cycle instructions, micro-operations, instructions, operations and functions, that make up these structures.

The main features of DIS, in addition to being distributed, include autonomy, asynchrony, greater separability, the predominance of interactions based on packet switching and messaging, as indivisible transactions [26, 27], the presence of memory, time delays, lower reliability of functioning [28, 29], a wide range of power consumption [30]. In the analysis of DIS energy consumption, in addition to these features, it is essential to take into account the sharply increased program-behavioral, intellectual implementation of many functions [31-33], which determines the appropriateness of the combined asynchronous-event, formal-abstract [34], intellectual [35-37], decomposition structural-spatial [38, 39] and sequential-temporal [40, 41] approaches based on the network and functional nature of DIS behavior.

Petri nets [42-44], extended with the properties of energy consumption [45], can be quite adequate formal models in monitoring the performance of DIS, in particular, with the analysis of their energy consumption, taking into account the noted features.

4. Development of a decomposition model for analyzing energy costs

The functioning of DIS components is accompanied by a certain power consumption, which depends, as noted above, on the technologies used for their implementation and operating modes. At the same time, the main emphasis in the analysis during the operation of the DIS can be made on the characteristics of energy consumption: firstly, instantaneous local energy structures (sections) for the components of the spatial architecture of DIS and various compositions of these components, essential for general or special analysis of the system's performance; secondly, dynamic energy structures (traces) for DIS components, related by direct behavioral interactions in various DIS work procedures; thirdly, the characteristic dynamic energy structures (traces) of the second type, which make it possible to identify the reference behavior of DIS components by their energy consumption.

4.1 Local energy properties and instant energy consumption state diagrams

Petri nets make it possible to efficiently represent asynchronous parallel processes of DIS functioning, including obtaining an analysis of additional temporal, probabilistic and energy characteristics, based on extensions [45].

In particular, the extended Petri net, which introduces the characteristics of energy consumption for events in positions, actions in transitions and transfers in tokens, has the form [45]:

$$S(f)^E=(P, T, X, Y, Ep, Et, F, S, M_0), \quad (1)$$

here P, T are the sets of positions, transitions, respectively, X, Y are the input and output alphabets, respectively, for events and actions, E_p, E_t are the sets of energy consumption of positions and transitions, F are the incidence relation of positions and transitions, S are the correspondences of events, actions, energy consumption to positions and transitions, M_0 - initial marking.

In this model, the energy consumption for the minimum, average and maximum reference values of the reduced numbers of one- and multi-circle switching of the events, that have occurred in the positions, and the actions, performed in the transitions, is localized in the components of the DIS and can be represented in the form of instantaneous energy structures and their compositions based on the spatial architecture DIS.

So, on the basis of the entire Petri net $S(f)^E$ or some Petri subnet for some of its special components, as their sub-models, diagrams of attainable instantaneous (at a certain moment of time t) energy states or markings (sections), marking the above events can be formed positions and transition actions, type:

$$Sp_t = \{(p_{i1},ep_{i1}),(p_{i2},ep_{i2}),\ldots,(p_{ik'},ep_{ik'}),\ldots,(p_{ik},ep_{ik})\} \cong \{(x_{i1},ep_{i1}),(x_{i2},ep_{i2}),\ldots,(x_{ik'},ep_{ik'}),\ldots,(x_{ik},ep_{ik})\}$$
$$St_t=\{(t_{j1},et_{j1}),(t_{j2},et_{j2}),\ldots,(t_{jk'},et_{jk'}),\ldots,(t_{jk},et_{jk})\} \cong \{(y_{j1},et_{j1}),(y_{j2},et_{j2}),\ldots,(y_{jk'},et_{jk'}),\ldots,(y_{jk},et_{jk})\} \quad (2)$$

For a markup diagram and some pair $(p_{ik'},ep_{ik'})$, where $k'\in\{1,2,\ldots,k\}$, $\{i_1,i_2,..,i_k\}\subseteq\{1,2,\ldots,n_P\}$ and $n_P=|P|$, the position $p_{ik'}$ is assumed to be ready, that is, the accomplished event $x_{ik'}$ of this position with the energy consumption $ep_{ik'}$, hence $(p_{ik'},ep_{ik'}) \cong (x_{ik'},ep_{ik'})$. In the case for some pair $(t_{jk'},et_{jk'})$, where $k'\in\{1,2,\ldots,k\}$, $\{j_1,j_2,..,j_k\}\subseteq\{1,2,\ldots,n_T\}$ and $n_T=|T|$, similarly, the transition $t_{jk'}$ is assumed in the completed state, that is, the perfect action $y_{jk'}$ of this transition with energy costs $et_{jk'}$, that is, $(t_{jk'},et_{jk'}) \cong (y_{jk'},et_{jk'})$.

The specified characteristics and relations of the adjacency of energy states in these diagrams represent the values of the energy consumption of DIS, as well as their dependences and transformations, selected for analysis in the Petri net model $S(f)^E$.

For DIS components, the minimum, average and maximum values of the reduced numbers of one- and multiple-cycle switches associated with input variables, synchronizations, parameters, data/operands and with internal/output variables, data, results for position events and energy state transition actions, characterize them inherent instantaneous local energy consumption.

Moreover, the energy consumption of transformations between adjacent energy states of the diagrams is not taken into account in these values of energy consumption. Energy state diagrams allow one-time, repeatable (periodically) or event-driven execution of an instant local selective analysis of energy consumption in the functioning of DIS in arbitrary, in particular, some critical states (sections) of the modeling Petri net $S(f)^E$ or its subnetwork without connection with the rest of the behavior.

For an analysis, that continues in time during unfolding for some points in time $(t_1, t_2, \ldots, t_m)$, the values of energy consumption for the corresponding selected energy states, it is also possible to form diagrams of their sequences, having the form:

$$\begin{gathered}Sp=(Sp(t_1)=\{(p_{i1}(t_1),ep_{i1}(t_1)),(p_{i2}(t_1),ep_{i2}(t_1)),\ldots,(p_{ik'}(t_1),ep_{ik'}(t_1)),\ldots,(p_{ik}(t_1),ep_{ik}(t_1))\},\\ Sp(t_2)=\{(p_{i1}(t_2),ep_{i1}(t_2)),(p_{i2}(t_2),ep_{i2}(t_2)),\ldots,(p_{ik'}(t_2),ep_{ik'}(t_2)),\ldots,(p_{ik}(t_2),ep_{ik}(t_2))\},\\ \ldots,\\ Sp(t_m)=\{(p_{i1}(t_m),ep_{i1}(t_m)),(p_{i2}(t_m),ep_{i2}(t_m)),\ldots,(p_{ik'}(t_m),ep_{ik'}(t_m)),\ldots,(p_{ik}(t_m),ep_{ik}(t_m))\})\end{gathered}$$

$$\begin{gathered}St=(St(t_1)=\{(t_{j1}(t_1),et_{j1}(t_1)),(t_{j2}(t_1),et_{j2}(t_1)),\ldots,(t_{jk'}(t_1),et_{jk'}(t_1)),\ldots,(t_{jk}(t_1),et_{jk}(t_1))\},\\ St(t_2)=\{(t_{j1}(t_2),et_{j1}(t_2)),(t_{j2}(t_2),et_{j2}(t_2)),\ldots,(t_{jk'}(t_2),et_{jk'}(t_2)),\ldots,(t_{jk}(t_2),et_{jk}(t_2))\},\\ \ldots,\\ St(t_m)=\{(t_{j1}(t_m),et_{j1}(t_m)),(t_{j2}(t_m),et_{j2}(t_m)),\ldots,(t_{jk'}(t_m),et_{jk'}(t_m)),\ldots,(t_{jk}(t_m),et_{jk}(t_m))\})\end{gathered} \quad (3)$$

Diagrams of energy states of modeling Petri nets $S(f)^E$ and sequence diagrams based on them can be used to partially check the operability and correct functioning of DIS components during selective monitoring and analysis of their energy consumption.

4.2 Moving energy properties and counters, diagrams of energy consumption behavior

Petri net extensions make it possible to determine dynamic behavioral energy structures (traces) of energy states (sections), formed by energy-loaded counter counters when modeling the Petri net $S(f)^E$ and associated with the direct functional interaction of DIS components in various DIS operation procedures.

Model (1), in addition to monitoring and analyzing the values of energy consumption in energy states, during the formation of events in positions and performing actions in transitions for a static relation of incidence F and correspondence S, allows monitoring and analysis of dynamic marking and accumulation of the reduced values of energy consumption in chips-counters, moved when modeling the Petri net $S(f)^E$, including comparing them with the minimum, average, maximum reference values.

On the basis of tokens, a set of three-level hierarchies of possible dynamic energy inputs is formed from the set $mPr=\{mpr_1, mpr_2,\ldots mpr_{n_p}\}$ for the objects of the Petri net behavior $S(f)$ - from the component Petri nets (hierarchy roots) through graph-topological energy structures (hierarchy nodes - chains, trees - branching and convergence, hammocks, cycles) to positions and transitions of the Petri net $S(f)^E$ (leaves of hierarchies - nodes of the Petri net) of the form [45]:

$$mpr_i=\{(root_i,\{(node_{i1},\{leaf_{i11},leaf_{i12},\ldots,leaf_{i1L1},\}),(node_{i2},\{leaf_{i21},leaf_{i22},\ldots,leaf_{i2L1},\}),\ldots,$$
$$(node_{iN1},\{leaf_{iN11},leaf_{iN12},\ldots, leaf_{i2N1L1},\})\})\}.$$
$$Leaf_i=\{leaf_{i11},leaf_{i12},\ldots,leaf_{i1L1},\}\cup\{leaf_{i21},leaf_{i22},\ldots,leaf_{i2L1},\}\cup\ldots\cup\{leaf_{iN11},leaf_{iN12},\ldots, leaf_{i2N1L1},\},$$
$$Node_i=\{node_{i1}\}\cup\{node_{i2}\}\cup\ldots\cup\{node_{iN1}\}.$$
$$leaf_{ij}(p)=M(p)=pr_2(S(p,x',ep'))=ep,$$
$$leaf_{ij}(t)=pr_2(S(t,y',et'))=et, \quad (4)$$

here $i\in\{1,2,\ldots,n_p\}$, for: a) the position $p\in P$, the current leaf token $leaf_{ij}(t)\in Leaf_i$; b) transition $t\in T$ of the Petri net $S(f)^E$; c) x', y' and ep', et' - respectively the events of the position, the transition action, the previous value of the energy consumption in the $leaf_{ij}(p)$ tokens that came to the position, p, $leaf_{ij}(t)$ the transition t; d) ep, et - values of energy consumption of position p, transition t.

For the initial token $leaf_{ij}(p)_0\in Leaf_{i0}$ of position $p\in P$ in the initial state of the Petri net $S(f)^E$, $leaf_{ij}(p)_0=M_0(p)$. For the node token $node_{ij}\in Node$ of the graph-topological structure, the root token $root_i$ of the hierarchy, and the entire Petri net $S(f)^E$ in its initial or current state, there is a possible accumulation of values immediately lower in the hierarchy of tokens [45]:

$$node_{ij}=leaf_{ij1}+leaf_{ij2}+\ldots+leaf_{ijLj},$$
$$root_i=node_{i1}+node_{i2}+\ldots+node_{iNi}$$
$$PNEnergy=root_1+root_2+\ldots+root_R \quad (5)$$

With an analysis continuing in time during unfolding for some points in time $(t_1, t_2, \ldots, t_m)$, the values of energy consumption for the corresponding selected graph-topological energy structures, as well as for energy state diagrams, it is possible to form sequence diagrams. that have the form:

$$node_{ij}(t_1)=leaf_{ij1}(t_1)+leaf_{ij2}(t_1)+\ldots+leaf_{ijLj}(t_1),$$
$$root_i(t_1)=node_{i1}(t_1)+node_{i2}(t_1)+\ldots+node_{iNi}(t_1),$$
$$PNEnergy(t_1)=root_1(t_1)+root_2(t_1)+\ldots+root_R(t_1),$$
$$node_{ij}(t_2)=leaf_{ij1}(t_2)+leaf_{ij2}(t_2)+\ldots+leaf_{ijLj}(t_2),$$
$$root_i(t_2)=node_{i1}(t_2)+node_{i2}(t_2)+\ldots+node_{iNi}(t_2),$$
$$PNEnergy(t_2)=root_1(t_2)+root_2(t_2)+\ldots+root_R(t_2),$$
$$\ldots,$$
$$node_{ij}(t_m)=leaf_{ij1}(t_m)+leaf_{ij2}(t_m)+\ldots+leaf_{ijLj}(t_m),$$
$$root_i(t_m)=node_{i1}(t_m)+node_{i2}(t_m)+\ldots+node_{iNi}(t_m),$$
$$PNEnergy(t_m)=root_1(t_m)+root_2(t_m)+\ldots+root_R(t_m). \quad (6)$$

The energy-loaded model defines the conditions for monitoring, analyzing and checking the correctness of energy consumption. The possibility of energy consumption in the Petri net $S(f)^E$ is determined by the possibility of executing graph-topological structures, as the possibility of executing all the events necessary for them in positions and actions in transitions. This possibility of execution is determined on the basis of reaching their three basic states of positions and transitions - waiting, readiness, execution - in accordance with the mandate (wand) functions of the tokens, moved during the operation of the Petri net $S(f)^E$. In the Petri net $S(f)^E$, some graph-topological energy structures can reach the state of partial execution or not execution at all due to the peculiarities of the current functioning or the incorrectness of the Petri net $S(f)^E$ itself. In this case, for the executable elements $Node_i'$, $Leaf_i'$, mpr_i', the inclusion $\varnothing\subseteq Node_i'\subseteq Node_i$, $\varnothing\subseteq Leaf_i'\subseteq Leaf_i$, $\varnothing\subseteq mpr_i'\subseteq mpr_i$ operates. Then it is true:

$$node_{ij}'=leaf_{ij1}'+leaf_{ij2}'+\ldots+leaf_{ijLj}'\leq node_{ij},$$
$$root_i'=node_{i1}'+node_{i2}'+\ldots+node_{iNi}'\leq root_i,$$
$$PNEnergy'=root_1'+root_2'+\ldots+root_R'\leq PNEnergy, \quad (7)$$

Thus, the possible minimum, average, maximum reference values of energy consumption of graph-topological energy structures in the achieved behavior of the Petri net $S(f)^E$ are not exceeded.

Three-level hierarchies of possible dynamic energy consumption of modeling Petri nets $S(f)^E$ and sequence diagrams, based on them, can be used to partially check the operability of DIS components during selective monitoring and analysis of their energy consumption.

4.3 Spatial and temporal energy structures and identifiers

Behavioral dynamic energy characteristics and structures are part of the behavioral fragments of the functioning of the Petri net $S(f)^E$, special behavioral characteristic types of which, for example, presented in the input-output alphabets of events and actions of the Petri nets $S(f)$, can be used to check and recognize the performance and the correct functioning of the DIS. Therefore, as part of such characteristic types of behavior - identifiers, check and recognition primitives and fragments of modeling Petri nets $S(f)^E$ [45], complementing them, reference dynamic energy characteristics and structures can be used to increase the completeness of such behavioral check and recognition.

Obviously, even with correct input-output characteristic fragments of the functioning of the DIS, the appearance of deviations in the values of the dynamic energy characteristics and structures from the reference ones, for example, below the minimum and above the maximum values, may indicate the beginning of the hardware degradation of DIS, leading to disruption of its functioning and performance.

Moreover, outside the direct correspondence to the input-output characteristic types of behavior of the modeling Petri nets $S(f)$, behavioral dynamic energy characteristics and structures considered independently, in a number of cases, can form energy characteristic projections of behavior - energy identifiers, check and recognition primitives and fragments (special energy graph-topological structures), represented in the energy alphabets Ep, Et, Em of energy consumption values for events in positions, actions in transitions, counters in tokens, respectively [45].

The energy alphabets of the Petri net $S(f)^E$ can be expanded by combining multiple energy inputs of events, actions, counters, as a result, extended energy alphabets $Ep' = N\times Ep$, $Et' = N\times Et$ and $Em' = N\times Em$. The expansion of energy-alphabets allows us to determine the sets of words starting and ending for combinations of energy consumption of events and actions, let e be the zero step, then:

$$W^{E\prime}=W^{PT\prime}\cup W^{PP\prime}\cup W^{TP\prime}\cup W^{TT\prime},\ W^{PT\prime}=(Ep'\times Em'\times Et')^*\cup\{e\},\ W^{PP\prime}=((Ep'\times Em'\times Et')^*\cup\{e\})\times Ep',$$
$$W^{TP\prime}=Et'\times((Ep'\times Em'\times Et')^*\cup\{e\}),\ W^{TT\prime}=Et'\times((Ep'\times Em'\times Et')^*\cup\{e\})\times Ep'. \quad (8)$$

Let $W^{E\prime}=\{w_1^{E\prime}, w_2^{E\prime},\ldots, w_{kw}^{E\prime}\}$ be the set of sequential multiple-linear words not marked by positions and transitions of energy behavior, which is reflected in the energy projection of the incidence relation F of the Petri net $S(f)^E$. For some $w_j'\in W'$ in the input-output alphabet $X'=N\times X\times Ep$ и $Y'=N\times Y\times Et$ [45] there exist $p_1,p_2\in P$ and $t_1,t_2\in T$, such that either $F(p_1,w_j')=p_2$, or $F(p_1,w_j')=t_2$, or $F(t_1,w_j')=p_2$, or $F(t_1,w_j')=t_2$, and $pr_3(w_j')=w_j^{E\prime}$, that is, the energy word $w_j^{E\prime}$ is the component wise third projection of the complete input-output word w_j' of the Petri net $S(f)^E$.

Let $Pr^E=\{pr_{1u}^E, pr_{2u}^E,\ldots, pr_{ku'}^E\}=\{Pr^{Ep}\cup Pr^{Et}\}$ be the set of verified energy properties of the form of third projections from, respectively, quadruples and triples of common verifiable properties Pr, obtained in [45] for the relation F and the quotient, included in F, matching S, that is:

$$Pr^E\subseteq pr_3(F:(P\times(X\times Ep)\to T)\cup(T\times(Y\times Et)\to P))\cup pr_3(S:((P\to X\times Ep)\cup(T\to Y\times Et)))=Ep\cup Et. \quad (9)$$

The first, particular approach to the definition and construction of energy-identifiers is based on the narrowing of the general identifiers of positions/transitions of the Petri net $S(f)^E$.

Let the set of candidates for energy identifiers be defined as a mapping ϖ based on projections of common identifiers of the form $ti_{jkpp}{}^{\rightarrow}=(p_{jtikp},\bigcup_{jtikip=1}^{kp}w_{jtikipp}{}^{\rightarrow}{}')$, $ti_{jkp}{}^{\rightarrow}{}_{p}=(\bigcup_{jtikip=1}^{kp}w_{jtikip}{}^{\rightarrow}{}_{p}{}',p_{jtikp})$, $ti_{jktt}{}^{\rightarrow}=(t_{jtikt},\bigcup_{jtikit=1}^{kt}w_{jtikitt}{}^{\rightarrow}{}')$, $ti_{jkt}{}^{\rightarrow}{}_{t}=(\bigcup_{jtikit=1}^{kt}w_{jtikit}{}^{\rightarrow}{}_{t}{}',t_{jtikt}) \in Ti$, obtained earlier in [45] as two positions/transitions and their characteristic behavior, that is, the mapping ϖ has view:

$$ei_{jkpp}{}^{\rightarrow}=\varpi(ti_{jkpp}{}^{\rightarrow})=(pr_1(ti_{jkpp}{}^{\rightarrow}),\ pr_3(pr_2(ti_{jkpp}{}^{\rightarrow})))=(p_{jtikp},\ pr_3(\bigcup_{jtikip=1}^{kp}w_{jtikipp}{}^{\rightarrow}{}'))=(p_{jtikp},\ \bigcup_{jtikip=1}^{kp}w_{jtikipp}{}^{\rightarrow E}{}'),$$
$$ei_{jkp}{}^{\rightarrow}{}_{p}=\varpi(ti_{jkp}{}^{\rightarrow}{}_{p})=(pr_3(pr_1(ti_{jkp}{}^{\rightarrow}{}_{p})),pr_2(ti_{jkp}{}^{\rightarrow}{}_{p}))=(pr_3(\bigcup_{jtikip=1}^{kp}w_{jtikip}{}^{\rightarrow}{}_{p}{}'),p_{jtikp},)=(\bigcup_{jtikip=1}^{kp}w_{jtikip}{}^{\rightarrow}{}_{p}{}^{E}{}'),p_{jtikp}),$$
$$ei_{jktt}{}^{\rightarrow}=\varpi(ti_{jktt}{}^{\rightarrow})=(pr_1(ti_{jktt}{}^{\rightarrow}),\ pr_3(pr_2(ti_{jktt}{}^{\rightarrow})))=(t_{jtikt},\ pr_3(\bigcup_{jtikit=1}^{kt}w_{jtikitt}{}^{\rightarrow}{}'))=(t_{jtikt},\ \bigcup_{jtikit=1}^{kt}w_{jtikitt}{}^{\rightarrow E}{}'),$$
$$ei_{jkt}{}^{\rightarrow}{}_{t}=\varpi(ti_{jkt}{}^{\rightarrow}{}_{t})=(pr_3(pr_1(ti_{jkt}{}^{\rightarrow}{}_{t})),\ pr_2(ti_{jkt}{}^{\rightarrow}{}_{t}))=(pr_3(\bigcup_{jtikit=1}^{kt}w_{jtikit}{}^{\rightarrow}{}_{t}{}'),t_{jtikt},)=(\bigcup_{jtikit=1}^{kt}w_{jtikit}{}^{\rightarrow}{}_{t}{}^{E}{}'),t_{jtikt},), \quad (10)$$

The set of energy identifiers $Ei=\{ei_{1ti},\ ei_{2ti},\dots,\ ei_{kti}\}\subseteq Ei_{jkpp}{}^{\rightarrow}\cup Ei_{jkp}{}^{\rightarrow}{}_{p}\cup Ei_{jktt}{}^{\rightarrow}\cup Ei_{jkt}{}^{\rightarrow}{}_{t}$ assumes for each eikti 'the property of injectivity, that is, $\forall ei_{kti'}\in Ei\ (|\varpi^{-1}(ei_{kti'})|=1)$.

Each energy-identifier remains uniquely incident to corresponding position p_{jtikp} and transition t_{jtikt}.

In the second approach, energy-identifiers are determined out of relation to common identifiers on the basis of their definition of the energy characteristic neighborhood of some position p_{jtikp} or transition t_{jtikt}. In this case, despite a different synthesis procedure, for example, based on the Rabin-Scott energy automata, they have the form:

$$ei_{jkpp}{}^{\rightarrow}=(p_{jtikp},\ \bigcup_{jtikip=1}^{kp}w_{jtikipp}{}^{\rightarrow E}{}'),$$
$$ei_{jkp}{}^{\rightarrow}{}_{p}=(\bigcup_{jtikip=1}^{kp}w_{jtikip}{}^{\rightarrow}{}_{p}{}^{E}{}'),p_{jtikp}),$$
$$ei_{jktt}{}^{\rightarrow}=(t_{jtikt},\ \bigcup_{jtikit=1}^{kt}w_{jtikitt}{}^{\rightarrow E}{}'),$$
$$ei_{jkt}{}^{\rightarrow}{}_{t}=(\bigcup_{jtikit=1}^{kt}w_{jtikit}{}^{\rightarrow}{}_{t}{}^{E}{}'),t_{jtikt},), \quad (11)$$

It should be noted that, in contrast to the functional, input-output representation of DIS behavior based on automata models, for example, Petri nets *S(f)*, its energy representation in most cases is not provided by an independent property similar to automata minimality. Consequently, the uniqueness of only energetic behavioral characteristic neighborhoods of positions and transitions is more difficult to achieve.

Hence, it is possible to conclude, that monitoring and analysis of dynamic energy characteristics and structures in the functioning of the DIS, even outside their direct correspondence to the input-output characteristic types of behavior of the modeling Petri nets *S(f)*, can be used for energy-consuming check and recognition of the functioning and operability of the DIS. However, it is more expedient, as noted above, to construct and use energy-identifiers, check and recognition primitives based on them in monitoring and recognizing the operability and correctness of the functioning of the DIS in connection with common, input-output identifiers, check and recognizing primitives [37, 41, 45], as their addition, essential for the efficiency of the analysis.

5. Basic steps of procedures for energy analyze, check and recognition

Monitoring and analysis of energy consumption, check and recognition of the operability and correctness of the functioning of DIS, extended by the analysis of energy consumption, as noted, can be performed one-time by the operator, periodically, for example, during routine maintenance, event-driven, when energy costs arise, that require an immediate response.

In the course of such analysis and check, it is possible to independently or comprehensively use diagrams of local energy states (sections) (Figure 1) and dynamic behavioral energy graph-topological structures, as well as sequence diagrams based on both of them, as part of more complex check and recognition of the performance and correct functioning of the modeling Petri net $S(f)^E$.

In any case, for monitoring, analysis, check and recognition of DIS, including energy, the main steps of the preprocessor and initialization stages are possible, providing for the initial *S(f)* and energy $S(f)^E$ modeling Petri nets:

1 General structural and functional analysis with the selection at the first step of the basic static graph-topological structures as part of the base/antibase, nodes, chains, trees, hammocks, feedback loops.

2 Determination of the local reduced minimum, average, maximum values of potential energy consumption for all static graph-topological structures for *S(f)* based on the features of design and technological solutions for the components of the simulated DIS with obtaining $S(f)^E$.
3 Determination of the necessary diagrams of local energy states and potential dynamic behavioral energy graph-topological structures, based on the features of DIS functioning.
4 Integration of the reduced minimum, average, maximum values of potential energy consumption for diagrams of local energy states and potential behavioral energy graph-topological structures.
5 Definition and integration of the tested functional properties and energy characteristics, as its special behavioral energy graph-topological structures.
6 Construction and integration of behavioral input-output and energy identifiers, check and recognition primitives.
7 General initialization of the placement of chips, initial values of information-parametric structures, initial values of energy consumption of diagrams of local energy states and potential behavioral energy graph-topological structures, selection of scenarios for monitoring, analysis, check and recognition of DIS and their initialization.

The stages of monitoring, analysis, recognition, testing of DIS, including complex input-output and energy, include the possible basic steps (Figure 2) for the energy modeling Petri net $S(f)^E$:

1 Selection and registration of real input-output and energy behavior in the alphabets of $S(f)^E$, performed in a passive mode for monitoring, analysis, recognition, operational check of the functioning of real DIS.
2 Selection and modeling of the reference input-output and energy behavior, based on the reference $S(f)^E$, performed for monitoring, recognition analysis, operational check for DIS functioning.
3 Comparative analysis of the selected real and reference input-output and energy behavior in the alphabets $S(f)^E$, performed for monitoring, analysis, recognition, operational check of the functioning of a real DIS.
4 Complex recognition of input-output and energy identifiers of reference positions/transitions $S(f)^E$, congruence of identically marked reference positions/transitions, determination of the registered behavior, based on the congruence of neighborhoods of behavior of the congruences positions/transitions.
5 Analysis of the completeness of check and recognition of input-output and energy behavior for the functioning of DIS, as a cover of the check primitives of check and recognition of $S(f)^E$.
6 Recognition and generation of confirmed connecting paths into $S(f)^E$ to uncovered check and recognition primitives, respectively, in the input-output, energy behavior of DIS.
7 Generation of tests for input-output and energy behavior $S(f)^E$, performed in the active mode according to special procedures, based on deterministic, pseudo-random, evolutionary optimization search and coverage for testing of the functioning of DIS.

Initial, intermediate and final execution conditions, precedence and compatibility relations, internal and external parallelism, distribution and separability, exception and error handling for the presented models and steps are determined, when constructing specific procedures and methods for monitoring, analysis, recognition, online and offline testing, data structures (Figure 3) depending on the features of the functioning of the selected class of real DIS, which is the subject of a subsequent study.

6. Estimates and experiments for the basic steps of the procedures

Representations of Petri nets *S(f)* by list structures for $|P|=n_p$, $|T|=n_t$, $|M|=n_m$, $n=n_p+n_t+2n_m$ (here $2n_m$ are two fields with a label index of an energy-loaded type and their number), $|Ev|=n_e$, $|X|=n_x$, $|Ac|=n_a$, $|Y|=n_y$, where $X\subseteq Ev$ and $Y\subseteq Ac$ for the transition requires no more than n_t memory cells containing no more than $2n_p+1_t+1_a+2_{Addr}$ ($2n_p+4$ conditional fields), for a position - no more than np conditional memory cells containing no more than $2n_t+1_p+1_e+2_{Addr}$ ($2n_t+4$ conditional fields). Here i_p is the field with the position index, i_t is the field with the transition index, $2i_m$ are the fields with the energy load and the number of label instances, i_e is the field with the event index, i_a is the field with the action index, $2i_{Addr}$ are the fields with the address of the next and previous cells list. The upper bound for the number of conditional fields is defined as:

$$c_{S(f)}=n_t(2n_p+1_t+1_a+2_m+2_{Addr})+n_p(2n_t+1_p+1_e+2_m+2_{Addr})=$$
$$=4n_pn_t+(2_m+2_{Addr})(n_t+n_p)+1_tn_t+1_an_t+1_pn_p+1_en_p\cong 4n_pn_t+6(n_t+n_p)\,. \quad (12)$$

The formulas for the limiting case for the number of conditional cells and the maximum length when searching in the graph for attainable states of the Petri net *S(f)* have the form:

$$c_{AS(f)}= d_{AS(f)}= 2*6n_tn_p(n_t)+2*6n_pn_t(n_p)=12n_pn_t(n_p+n_t),$$
$$c_{AS(f)multi}= d_{AS(f)multi}= 6n_t(n_p2^{nt+1}-1)+6n_p(n_t2^{np+1}-1)=3n_t(n_p2^{nt+2}-2)+3n_p(n_t2^{np+2}-2). \quad (13)$$

The representation of the Rabin-Scott automaton $RS_{S(f)}$ differs in the order and multiplicity of positions, transitions, labels, events and actionsб associated with them. In the general case, the upper bound on the number of fields in conventional cells is not more than:

$$C_{RSS(f)p\text{-}n}=min((2n_p+1)*min((2^{np}-1), n_t),(2n_p+1)*(2^{nt}-1))=(2n_p+1)*min(min((2^{np}-1), n_t),(2^{nt}-1)). \quad (14)$$

The total estimate of the fields does not exceed:

$$C_{RSS(f)pet\text{-}n}= (2n_p+1)*min(min((2^{np}-1), n_t),(2^{nt}-1))+\Sigma_{i=0}^{np}(3n_p*n_t^i). \quad (15)$$

The estimates represent the upper limits of the applicability of the abstract model, and their reduction is expedient due to the decomposition of the Petri net model *S(f)*.

Algorithmic, informational and software implementation of basic steps of check and recognition of correctness of DIS functioning, taking into account energy consumption, was carried out for three-level DIS as a part (server-administrative), zonal (regional, broker) and point (local, controller, sensor-actuator) levels.

For levels it is assumed ascending collection, operative processing and generalization of information, descending management.

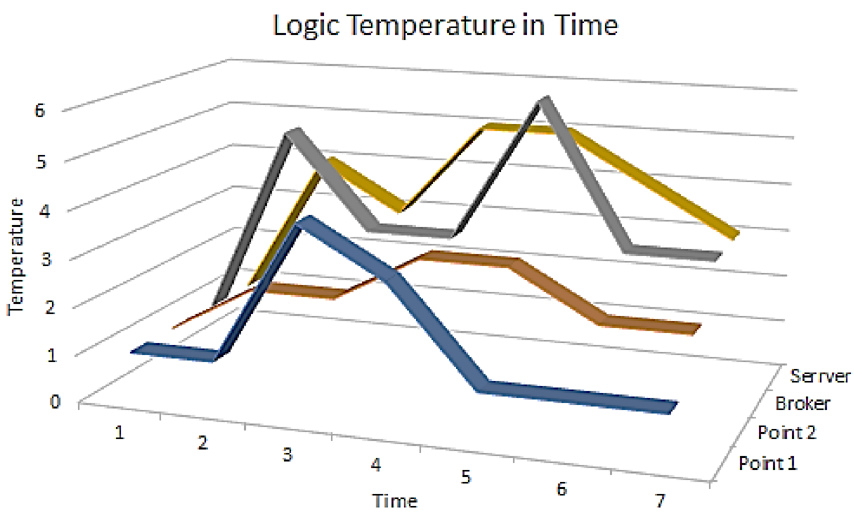

Figure 1: Time temperature monitoring of DIS

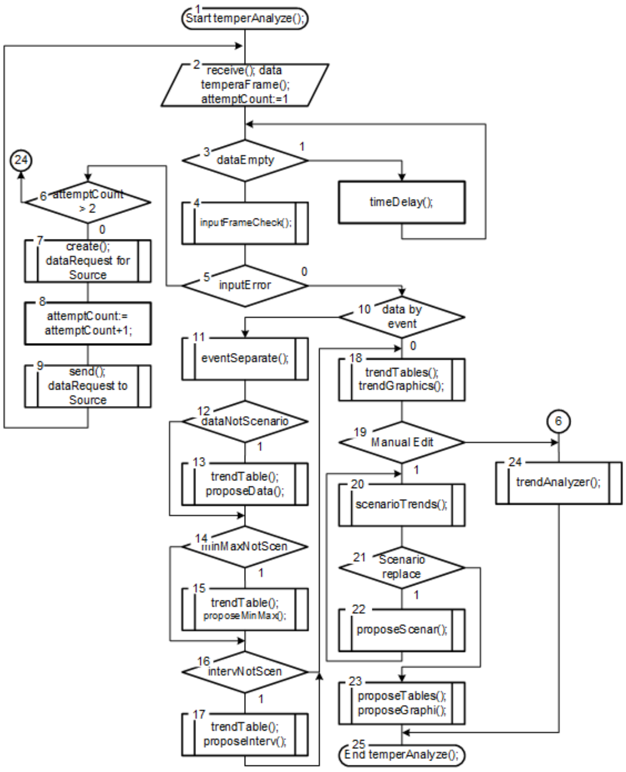

Figure 2: Basic algorithm of energy check

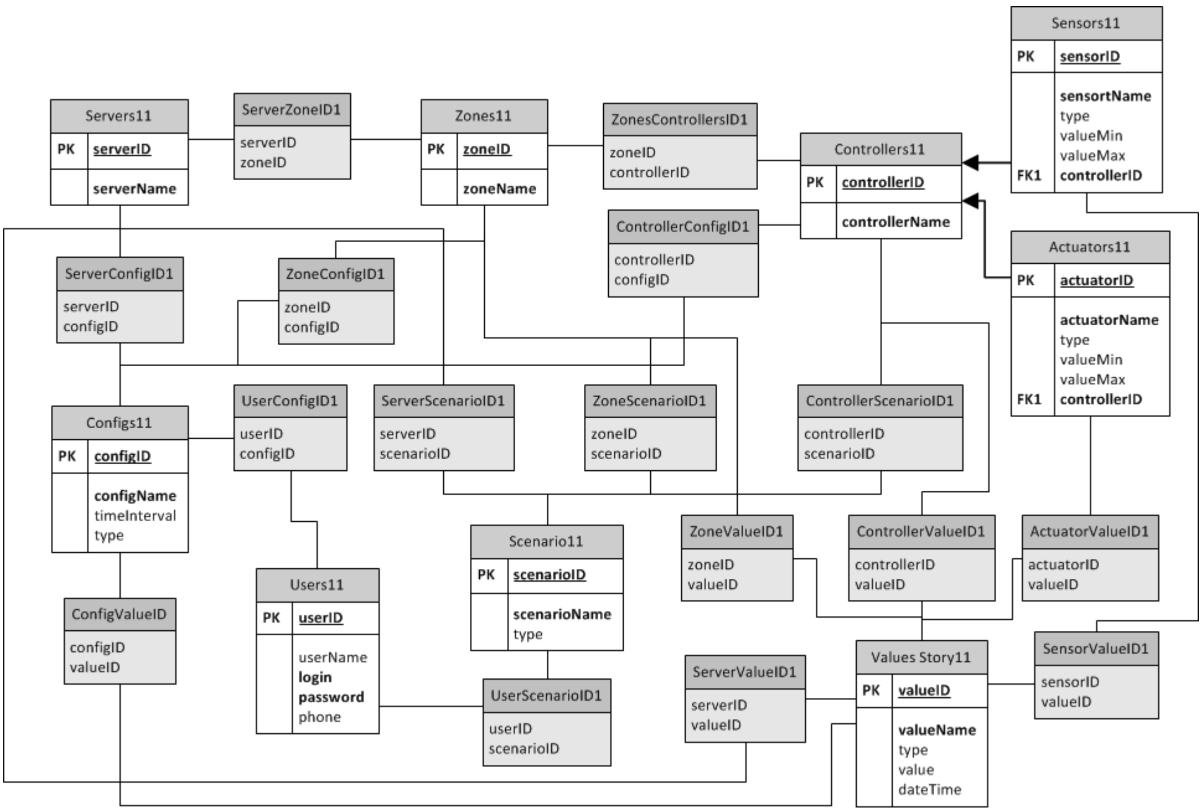

Figure 3: Normalized model of Data Base for energy check

Approbation of models and basic steps of their analysis and check of correctness of functioning, taking into account energy consumption for real components of DIS of a class of system "smart house", showed possibility and expediency of their application. Real estimates of the complexity of the check analysis, due to functional features and partial certainty of behavior, were lower on 80-87% than formal upper estimates and amounted to 10^2-10^5 conditional vertices with 10-2-10 seconds of timer, which allows you to perform real-time testing of components of DIS.

7. Conclusion

In the present work, we propose the development of a model of Petri nets in the behavioral check of DIS, supplemented by the verification of energy consumption. The energetic characteristics added to positions, transitions and tokens allowed for a comprehensive analysis of the energy-loaded behavior of the model at three levels of the reference model. The first level is represented by states / positions with characteristic energy costs. The second level is represented by possible behavioral topological structures with characteristic power consumption. The third level is topological structures achievable when moving chips with characteristic, including identifying, power consumption. Check model makes it possible to determine and save the energy consumption of DIS components, based on a formal analysis of the behavior of model elements - vertices, graph-topological elements and subnets of the Petri net. This in turn allows you to determine the steps of the basic behavioral check procedures for DIS components, extended by checking their energy consumption.

8. References

[1] O. Ivanchenko, V. Kharchenko, B. Moroz, L. Kabak, S. Konovalenko, Risk assessment of critical energy infrastructure considering physical and cyber assets: Methodology and models, in: proc. 2018 IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS), 2018.

[2] C. Piguet, Low-power CMOS circuits: technology, logic design and CAD tools, CRC/Taylor & Francis, 2005. 440 p.

[3] S. Tyurin, A. Kamenskih, Green Logic: Models, Methods, Algorithms, Green IT Engineering: Concepts, Models, Complex Systems Architectures. Studies in Systems, Decision and Control, V. Kharchenko, Y. Kondratenko, J. Kacprzyk (eds) vol. 74, Springer, 2017. pp. 69 – 86.

[4] S. Murugesan, G. Gangadharan, Harnessing Green IT. Principles and Practices, UK: Wiley and Sons Ltd. 2012.

[5] Intel Quartus Prime Standard Edition User Guide: Power Analysis and Optimization. 2018. https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/ug/ug-qpspower.pdf, last accessed 2019/03/20

[6] Analog Devices Power Monitors, last accessed 2019/03/20. (2019) https://www.analog.com/en/parametricsearch/11099,

[7] P. Ammann, J. Offutt, Introduction to Software Testing. Cambridge University Press. 2016. p. 26.

[8] O. Drozd, K. Zashcholkin, R. Shaporin, J. Drozd, Y. Sulima, Development of ICT Models in Area of Safety Education, in: Proceedings of 18th IEEE East-West Design & Test Symposium, Varna, Bulgaria, 2020, pp. 115 –119. doi: 10.1109/EWDTS50664.2020.9224861

[9] O. Drozd, K. Zashcholkin, O. Martynyuk, O. Ivanova, J. Drozd, Development of Checkability in FPGA Components of Safety-Related Systems, in: proc. CEUR Workshop Proceedings, vol. 2762, 2020. pp. 30 – 42. Online http://ceur-ws.org/Vol-2762/paper1.pdf

[10] O. Savenko, S. Lysenko, A. Nicheporuk, B. Savenko, Approach for the Unknown Metamorphic Virus Detection, in: proc. The 8-th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Bucharest (Romania), September 21–23, 2017. Bucharest, 2017. pp. 71 – 76.

[11] S. Lysenko, O. Savenko, K. Bobrovnikova, A. Kryshchuk, B. Savenko, Information technology for botnets detection based on their behaviour in the corporate area network, Communications in Computer and Information Science. 718 (2017) 166 -181.

[12] A.V. Palagin, V.N. Opanasenko, S.L. Kryvyi, Resource and Energy Optimization Oriented Development of FPGA-Based Adaptive Logical Networks for Classification Problem, Green IT

Engineering: Components, Networks and Systems Implementation, V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), Vol. 105, Berlin, Heidelberg: Springer International Publishing, 2017. pp. 195 – 218.

[13] IEC 61508-1:2010. Functional Safety of Electrical/Electronic/Programmable Electronic Safety Related Systems – Part 1: General requirements, Geneva: International Electrotechnical Commission. 2010.

[14] Mohamad Najem, Pascal Benoit, Florent Bruguier, Gilles Sassatelli, Lionel Torres, Method for Dynamic Power Monitoring on FPGAs, in: proc. Conference Digest of 24th International Conference on Field Programmable Logic and Applications, Munich, Germany, September 1–5. 2014.

[15] C. Metra, L. Schiano, M. Favalli, B. Ricco, Self-checking scheme for the on-line testing of power supply noise, in: Proceedings of Design, Automation and Test in Europe Conf. Paris (France). 2002. pp. 832 – 836.

[16] V. Székely, Cs. Márta, M. Rencz, Zs. Benedek, B. Courtois, Design for Thermal Testability (DFTT) and a CMOS Realization, in: proc. 1st Therminic Workshop, Grenoble, France. 1995.

[17] O. Drozd, V. Romankevich, M. Kuznietsov, M. Drozd, O. Martynyuk, Using natural version redundancy of FPGA projects in area of critical applications,. in: proc. 11th IEEE International Conference DESSERT, Ukraine. 2020. pp. 58-63. doi: 10.1109/DESSERT50317.2020.9125050

[18] O. Drozd, I. Perebeinos, O. Martynyuk, K. Zashcholkin, O. Ivanova, M. Drozd, Hidden fault analysis of FPGA projects for critical applications, in: Proceedings of IEEE International Conference TCSET. Paper 142, Lviv-Slavsko, Ukraine, 2020. doi: 10.1109/TCSET49122.2020.235591

[19] O. Drozd, M. Kuznietsov, O. Martynyuk, M. Drozd. A method of the hidden faults elimination in FPGA projects for the critical applications, in: proc. 2018 IEEE 9th International Conference on Dependable Systems, Services in and Technologies (DESSERT'2018), Kyiv, Ukraine, 2018. pp. 231 – 234, doi: 10.1109/DESSERT.2018.8409131

[20] Ten Chee-Wooi, Govindarasu Manimaran, Liu Chen-Ching, Cybersecurity for Critical Infrastructures: Attack and Defense Modeling, IEEE Transactions on Systems, Man, and Cybernetics 40 4 (2010) 853 – 865.

[21] V. A. Romankevich, Self-testing of multiprocessor systems with regular diagnostic connections, Automation and Remote Control. vol. 78, issue 2, 2017. pp. 289 – 299.

[22] D. Basile, F. Di Giandomenico, S. Gnesi, A Refinement Approach to Analyse Critical Cyber-Physical Systems. SEFM Workshops, 2017. pp. 267 – 283.

[23] Trong Duc Nguyen, Approximate query processing in a data warehouse using random sampling, Graduate Theses and Dissertations. 2020. 18195. https://lib.dr.iastate.edu/etd/18195

[24] A. Drozd, M. Lobachev, J. Drozd, The problem of on-line testing methods in approximate data processing, in: Proceedings of 12th IEEE International On-Line Testing Symposium, Como, Italy, 2006. pp. 251 – 256, doi: 10.1109/IOLTS.2006.61

[25] A. Drozd, S. Antoshchuk, New on-line testing methods for approximate data processing in the computing circuits, in: proc. 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Prague, Czech Republic, 2011. pp. 291 – 294. doi:10.1109/IDAACS.2011.6072759

[26] G. Coulouris, J. Dollimore, T. Kindberg, G. Blair, Distributed Systems: Concepts and Design, 5th ed. Boston: Addison-Wesley, 2011. 1067 p.

[27] A.A. Bakker, M. Steen, A.S. Tanenbaum, Wide-Area Distribution Network for Free Software, ACM Trans. on Internet Technology 6 (2006) 259-281.

[28] V. Hahanov, E. Litvinova, S. Chumachenko, A. Hahanova Cyber Physical Computing. In: Cyber Physical Computing for IoT-driven Services. Springer, Cham. 2018. doi: https://doi.org/10.1007/978-3-319-54825-8_1

[29] V. Hahanov, A. Zhalilo, W. Gharibi, E. Litvinova, Cloud-driven traffic control: Formal modeling and technical realization, in: proc. The 2015 4th Mediterranean Conference on Embedded Computing, MECO 2015 - Including ECyPS 2015, BioEMIS 2015, BioICT 2015, MECO-Student Challenge 2015. pp. 21 – 24. doi: 10.1109/MECO.2015.7181896

[30] Y. Kondratenko, V. Korobko, O. Korobko, Distributed Computer System for Monitoring and Control of Thermoacoustic Processes, in: proc. The 2013 IEEE 7th Int. Conf. on Intelligent Data

Acquisition and Advanced Computing Systems (IDAACS). Berlin, vol. 2, Sept. 12 – 14, 2013, pp. 249-253.
[31] D. Maevsky, V. Kharchenko, M. Kolisnyk and E. Maevskaya, Software reliability models and assessment techniques review: Classification issues, in: proc. 2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Bucharest, 2017, pp. 894 – 899, doi: 10.1109/IDAACS.2017.8095216.
[32] T. Hovorushchenko, O. Pavlova, Method of Activity of Ontology-Based Intelligent Agent for Evaluating the Initial Stages of the Software Lifecycle. Advances in Intelligent Systems and Computing, 836 (2019) 169-178. doi: 10.1007/978-3-319-97885-7_17
[33] T. Hovorushchenko, O. Pavlova, M. Bodnar, Development of an Intelligent Agent for Analysis of Nonfunctional Characteristics in Specifications of Software Requirements, Eastern-European Journal of Enterprise Technologies, 1 2(97) (2019) 6-17. doi: 10.15587/1729-4061.2019.154074
[34] V.B. Kudryavtsev, I.S. Grunskii, V.A. Kozlovskii, Analysis and synthesis of abstract automata, Journal of Mathematical Sciences, September, 169 4 (2010) 481-532.
[35] D.E. Ivanov, T.A. Vasyaeva, Genetic Algorithm for Estimating Peak Heat Dissipation for Large Integrated Circuits, Bulletin of East-Ukrainian National University named after V. Dal, 13 (167) (2011) 277 – 283.
[36] A. Sugak, O. Martynyuk, A. Drozd, Models of the Mutation and Immunity in Test Behavioral Evolution, in: proc. 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Warsaw, Poland, 2015, pp. 790 – 795. doi: 10.1109/IDAACS.2015.7341411
[37] Y.A. Skobtsov, V.Y. Skobtsov, A.A. Shalyto, Test generation and simulation for crosstalk faults, Radio Electronics, Computer Science, Control, 4 (2015) 72–78. doi: 10.15588/1607-3274-2015-4-11
[38] L. Gomes, J.M. Fernandes, Behavioral Modeling for Embedded Systems and Technologies: Applications for Design and Implementation, 2010 doi: 10.4018/978-1-60566-750-8
[39] Combining model learning and model checking to analyze TCP implementations. In Computer Aided Verification, in: Proceedings of 28th International Conference, CAV 2016, Toronto, ON, Canada, 2016, vol. 9780,pp. 454–471.
[40] A. Classen, M. Cordy, P.-Y. Schobbens, P. Heymans, A. Legay, J.-F. Raskin, Featured Transition Systems: Foundations for Verifying Variability-Intensive Systems and Their Application to LTL Model Checking, TSE, 39(8) (2013) 1069 – 1089.
[41] A. Classen, M. Cordy, P.Y. Schobbens, P. Heymans, A. Legay, J.F. Raskin, Featured Transition Systems: Foundations for Verifying Variability-Intensive Systems and Their Application to LTL Model Checking, TSE, 39(8) (2013) 1069 – 1089.
[42] J. Desel, J.F. Esparza, Choice Petri Nets, Cambridge University Press, Cambridge, 1995, URL: https://www7.in.tum.de/~esparza/fcbook-middle.pdf
[43] J. Kleijn, A.Yakovlev, (Eds). Petri nets and Other Models of Concurrency, ICATPN, Lecture Notes in Computer Science,_Vol. 4546, Springer-Verlag, 2007.
[44] I. Poliakov, A. Mokhov, A. Rafiev, D. Sokolov, A. Yakovlev, Automated Verification of Asynchronous Circuits Using Circuit Petri Nets, in: proc._The 14th IEEE International Symposium on Asynchronous Circuits and Systems,_Newcastle upon Tyne, UK, 2008. pp. 161-170. doi**:** 10.1109/ASYNC.2008.18
[45] R. Marinescu, E.P. Enoiu, C. Seceleanu Statistical analysis of resource usage of embedded systems modeled in east-adl. In Annual Symposium on VLSI, IEEE, 2015. pp. 380–385.