Date of publication SEP-30, 2024, date of current version JUN-29, 2024.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.23.3.XXXX

The Method of Diversity-Based Ensuring the Reliability of the Router in the IIoT System

MARYNA KOLISNYK[1,2], IRYNA PISKACHOVA[3], VYACHESLAV KHARCHENKO[2]

[1]Institute of telecommunications of Vienna University of Technology "TU Wien", Vienna, Austria
[2]Department of computer systems, networks and cybersecurity of National aerospace university "KhAI" Kharkiv, Ukraine.
[3]Department of automation and computer-integrated technologies of State Biotechnological University, Cyberport Educational and Research Institute, Kharkiv, Ukraine

Corresponding author: Maryna Kolisnyk (e-mail: m.kolisnyk@csn.khai.edu).

ABSTRACT The architecture of the Industrial Internet of Things (IIoT) system assumes the presence of a router as the main subsystem, which will allow connecting the subsystems into a single network and connecting them to the equipment of the Internet service provider. Failure of the router can lead to downtime of industrial equipment and the whole workshop at the industrial enterprise, and to the failure of the IIoT system as a whole. Control processing units of industrial routers (RCPU) are complex microprocessor systems (MS). Violation of their operability can be caused by both hardware (HW) and software (SW) failures. Therefore, to increase the reliability of RCPU it is reasonable to use such methods of reliability assurance as HW redundancy (duplication of processors with comparison of results), as well as the use of modern means of control and diagnostics. However, the number of HW and SW failures in RCPUs, is still quite high. In this connection, the solution of issues related to the development of methods for ensuring high reliability of RCPU is of great urgency. This paper proposes a method of ensuring the reliability of RCPU using SW diversity on the basis of priority series. The method let to choose the variant of redundancy of HW and the number of versions of SW as diversity in RCPU.

KEYWORDS IIoT; Reliability; N-version programming; Router; CPU.

I. INTRODUCTION

A. MOTIVATION AND GOAL OF THE PAPER

FOR modern telecommunication systems, such as Industrial Internet of Things (IIoT) systems [1, 2], one of the mandatory requirements is to ensure high reliability. One of the most important subsystems of the IIoT is the industrial router, on the reliable functioning of which depends the reliable functioning of the entire IIoT system. Router includes hardware (HW) and software (SW). The most important device in the structure of the router is the control processing unit (RCPU). This device has one processor with options of core redundancy: either two cores (dual) or four (quad) processor cores. Most often in RCPU a dual core with homogeneous architecture is used (two information processing processes are executed in parallel).

SW (functional and system) is stored in such types of memory as ROM, RAM, NVRAM, Flash-memory (like SSD, SD-card). In a complex HW-SW system router it is possible to use redundancy to ensure the reliability of both HW and SW.

In the latest versions of industrial router provides several versions of configuration files that can be stored in NVRAM, and then rewritten in RAM, as well as for storing SW versions can be used cloud, which is accessed by router in case of need. Also, the system SW (router's operating system (iOS)) can have multiple versions that are stored in flash memory, and can also be saved to a bootable flash drive, or to an SSD drive. SW version updates can be downloaded from cloud.

SW diversity and HW redundancy are well-known methods of reliability assurance. But the question arises: which structure of RCPU is more reliable from the point of view of HW and SW redundancy? How many HW channels of RCPU are sufficient to provide the required reliability, and how many necessary and sufficient SW versions should be used in RCPU? In router there is such practice of using RCPU: two identical chips with the same SW are installed, in case of SW failure either the same redundant version of SW is reloaded or, if the operation of this version cannot provide reliable operation of R, another version of SW is loaded. However, such a scheme leads to time wasted in reprogramming the RCPU. Hence, the problem of improving the reliability of RCPU and the whole router without unnecessary time expenditure is relevant.

This paper proposes a method for selecting the redundancy structures of the HW and SW of RCPU on the basis of priority series developed by the authors to ensure the required reliability of RCPU.

B. WORK RELATED ANALYSIS

The problems of SW reliability have been addressed by the scientific schools of A. Avizienis [3], B. Littlewood [4, 5], L. Strigini [4-6], P. Popov [5, 7], A. Romanovsky [7], V. Kharchenko [1, 8], W. Kuo [9], B. Volochiy [10], V. Krasnobayev [11].

The study of SW reliability and diversity, as well as HW redundancy in their works was considered by scientists Babeshko E. [8], Kolisnyk M. [12], Piskachova I. [12], who proposed to use SW and HW redundancy for digital telecommunication systems. The authors of the scientific paper [13] proposed a model of SW reliability with control of criticality of its errors. Some research papers such as [14, 15] proposed the creation of reliable FPGA-based routers, and the application of SW diversity for SW logic controllers [16]. Preserving the stability of the old SW version without giving up features and fixes added to the new version [17].

The majority voting algorithm was modified using an online stability check with a proposed N-version (multiversion) controller [18].

Model and method diversity metrics are defined and reliability improvement with N-version machine learning architecture is analytically demonstrated [19].

An N-version anomaly-based fault detection (NvABFD) technique is proposed and utilized to improve the fault tolerance of fog-based IoT systems [20].

The task of investigating the reliability of router control device using SW diversity and HW redundancy is new, and requires the development of a method to ensure the reliability of RCPU using multiple SW versions. One of the most convenient ways to choose the redundancy structure based on the given requirements is the creation of priority series.

Purpose of the paper: development of a method to ensure the reliability of RCPU on the basis of priority series which will be developed and ordered considering a set of fault-tolerant structures with version redundancy.

II. ANALYSIS OF TWO-VERSION THE ROUTER'S ARCHITECTURE

The architecture of the modern IIoT RCPU includes 2 processors with multicore (dual or quad cores) [16].

The diagram (Fig. 1) shows the types of memory used by the router's CPU. In the latest router models, it is possible to use several iOS versions. Extended flash memory capacity of SSD up to 100 GB and EMMS 16 GB, possibility of using different storage media (16 GB SD card, up to 32 GB flash drive) allow storing images of the current OS and different OS versions (different CPU processors can work with different OS versions) on the SSD disc and on the bootable flash drive. The configuration file is stored in NVRAM and can be overwritten in RAM, changed, and overwritten again both in NVRAM and on the flash drive and on the SD card. Log files from the RAM buffer are overwritten to eMMS. ARP tables, routing tables are written for storage on the SSD disc. Thus, SW multiversion can be realized in modern routers with sufficient memory capacity.

Complex real-time MS, such as RCPU, have high reliability requirements. Their high reliability can be ensured by means of HW redundancy and SW redundancy [1-6].

SW redundancy can be implemented in the following ways [1, 3-8]: independent development of SW versions; introduction and maintenance of two or more variants (versions) of programs that perform the same functions, etc.). Such programming is called diversity programming [4-7]. Diversity can be introduced at the stages of [5]-[7]: specification development; design (algorithms, data structures, programs, etc.); coding; testing and verification, etc. [1-4].

Also, the diversity of RCPU SW can be useful in patching and in prevention of cyber-attacks on vulnerabilities [1]. One version of iOS can have vulnerabilities, which can be used by attackers, but another version of iOS doesn't have the same vulnerabilities. So, diversity of iOS will help to prevent successful cyber-attacks on vulnerabilities.

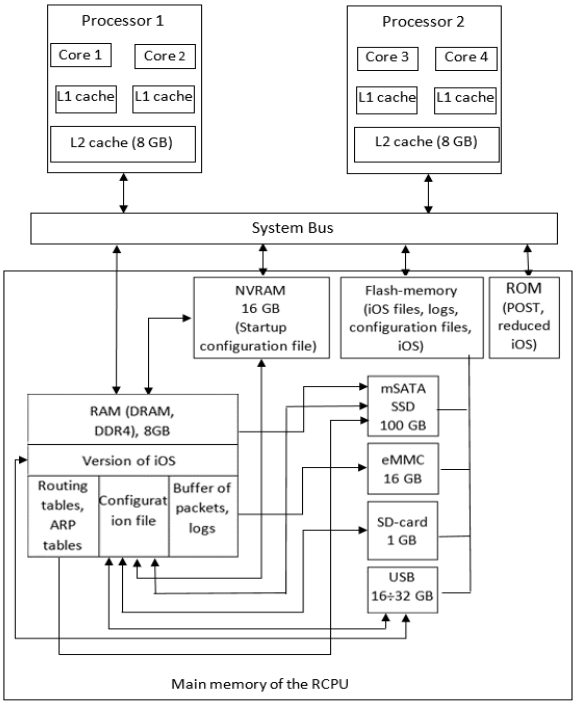


Figure 1. The main locations of SW in N-version programming in different types of memories.

Let us conduct a system analysis of the router architecture, which will allow us to determine how feasible and reasonable is the implementation of SW diversity and HW redundancy in the router.

The analysis has shown that there are the following types of memory in R, the disruption of which can lead to the failure of RCPU and the entire IIoT system and, under certain conditions, to catastrophic consequences [21-23]: RAM (Random Access Memory) – the current configuration and routing tables of the device are stored here. This type of memory loses its contents when the device is restarted.

The most commonly used RAM in router is dynamic random-access memory (DRAM). All data that is stored in the router's DRAM will be lost when it is switched off.

This memory is divided into two parts:

Main Processor Area. This area stores information about router's operational configuration (commands that are currently executing), routing tables, and ARP (Address Resolution Protocol) tables.

An area of shared I/O memory that acts as temporary memory that stores queued (buffer) data packets. This area

stores data packets that pass-through router. If the processes in router are fast enough compared to a fast processor, data packets will pass through it. If router is overloaded with data packets, it needs to buffer them in this memory area. This memory can be physically removed from router and increased in size by inserting a larger DRAM card or adding additional DRAM cards if router has additional DRAM slots. Whenever the router is rebooted, all data stored in RAM is deleted. NVRAM can be used to permanently store router setup data.

Flash memory is an EPROM (erasable programmable persistent memory device). The information stored in R's SIM card is not lost when router is switched off and is stored by iOS of the router. Flash memory is used to store iOS SW images. It can also be used to store other files such as configuration backup files, log files, telephony settings, HTML files, configuration backups, and more. When iOS of router is rebooted, it is copied from flash memory to RAM. It retains its contents even after the device restarts. In modern models of routers, its volume reaches 16 GB, and can be used to back up system SW. In addition to flash memory chips on the router's motherboard, bootable USB memory sticks can be used to recreate the router iOS image during a system upgrade or to reboot the router in case of a boot failure. During the flash-USB boot process, router reboots using the version available on the USB memory stick.

Another way to restore the SW is to reprogram the SIM card and load a different version of iOS on it which will support different features of the router. This memory can be easily removed from the router and expanded to a larger size by inserting a larger flash card into the router or adding additional SIM cards if the router has multiple flash card slots.

ROM (permanent memory) – stores the boot program, which is used to initialize the boot process. This is where RCPU reads the microcode to start the boot process and router's basic checks. Four basic functions are performed here:

POST – Power-on self-test. The RCPU performs POST to make sure all components are working, this is a variant of the router's self-test of the RCPU and interfaces. This is where the processor, the amount of DRAM and flash memory installed, and all interfaces are tested.

Bootstrap — the boot program initializes the RCPU and starts the router boot process by detecting and loading the iOS.

RxBoot is a mini-iOS that is used when the iOS R is corrupted or the flash card is empty and an iOS version needs to be installed. It has limited functionality, so there is an option to download a new version of iOS to the flash card.

ROM Monitor is a mode that is used to diagnose or boot the iOS image through a console session.

The RCPU bootstrap procedure starts with ROM memory testing. The ROM consists of SW instructions such as POST (power-on self-test) and the Bootstrap program. If the RCPU, RAM, and interfaces are not working properly, POST sends an error message. The bootstrap application is then used to configure the RCPU bootstrap functions. The boot application is responsible for detecting and booting the router iOS. All this information is saved/stored in ROM and the data is retained even if R is powered off or rebooted.

NVRAM (non-volatile RAM) – used to store R's startup configuration file. Startup configuration files are copies of the router's configuration (settings) file and are saved after its rebooted. In NVRAM, if the router is rebooted or shut down, the data is not lost and can be easily restored. The configuration is also stored in DRAM, it is the operational configuration that is lost when the router is switched off.

The router architecture allows the introduction of SW diversity and for this purpose NVRAM, Flash types of memory can be used. Also, inside the RCPU cores there is redundancy, and most often two redundant cores are used in order to parallelize information processing.

The paper investigates a method of improving the reliability of RCPU using HW and SW diversity. Modelling allows determing how much it is necessary to introduce both HW and SW diversity, and what kind of RCPU organization structure (how many HW channels and how many SW versions are needed) will increase its reliability.

III. THE MAIN METRICS OF SW RELIABILITY IN N-VERSION PROGRAMMING

Dual programming is of great interest. The fact that the results differ indicates an error. To detect a discrepancy in diversity structures and localize the failed channel in two-channel structures, it is necessary to use highly reliable control and diagnostic tools.

When dual programming is used to increase the level of SW correctness, the same algorithm can be implemented using two different programming languages [3-12]. The language for the main version of the SW is determined by the specifications and requirements of the system in which it is used. The language for the backup version is chosen based on considerations of convenience and debugging capabilities.

There is another way to create diversity SW, when one program is developed, but debugged by two programmers independently of each other. In this case, the total number of errors remaining in the program can be estimated by using the number of coincident errors found by the two programmers.

The correct operation of SW depends on random changes in external influences, on the parts of the program involved in the work and their states, so the quality of the program's functioning is characterized by the laws of probability. Therefore, to assess SW reliability, it is necessary to select reliability indicators that take into account the random nature of the influencing factors and characterize the program's debugging.

The following requirements are put forward for SW reliability indicators [3, 4, 7, 12, 24]: completeness of accounting for the system's features and conditions of use; simplicity of physical interpretation and use for calculations; high sensitivity to the estimated parameter; convenience of checking the actual values of reliability indicators during operation.

The main SW reliability indicators are: is the number N of errors remaining in the program after debugging and correction; probability of failure-free operation of the SW P(t) in the interval from 0 to t (before the first failure and error correction) from t_i to t_i+1 (ti is the time of start of operation after SW error correction to t_i+1 is the time of SW recovery.

If a non-renewable object operates once during a given period of time, then it is advisable to choose the probability of failure-free operation (PF) P(t) or the probability of failure Q(t) as an indicator of reliability, related by the dependence [13].

Another common quantitative measure of reliability is the failure rate λ, which is the conditional probability that an error will appear in the interval from t to t+Δt, provided that no errors have been detected by the time t.

At the initial moment, the SW contains the number of errors N_0 – and from the beginning of its use, a failure stream with a certain failure rate λ_0 is detected, which, given a constant number of errors, is a constant value ($\lambda_0(t) = \lambda_0 = const$). With each new SW recovery, the number of errors changes, and the failure rate λ_i changes. Separating the number of errors that appeared at certain time intervals during SW debugging, it is possible to determine approximate values of the intensity of the SW error flow (failure rate) at these points. The number of errors contained in a program cannot be determined precisely. The dependence N(t) cannot be used as the basis for a SW reliability model.

We use the Wei Kuo [9] and Jelinsky-Moranda [25] models for our research. The development of such a model begins with specifying the behavior of the function $\lambda(t)$.

During SW debugging and testing, secondary errors are first detected, that is, the results of the manifestation of initial defects, which should be qualified as primary errors.

The intensity of the manifestation of secondary SW errors depends on the total number of errors in the SW or on the probability of an error in the command. Time intervals between SW distortions are assumed to be statistically independent.

The assumptions for the developed method are as follows [12]: monitoring and diagnostic tools are close to perfect and allow to determine the technical condition of the RCPU SW with a high degree of reliability; the number of SW errors and primary defects in router SW are constant on the interval from the start of operation to the first correction of old errors, the number of primary defects in RCPU SW is constant and the introduction of new ones is likely, and, therefore, it is fair to assume that the SW failure rate follows an exponential law (i.e., the failure rate before the first SW failure is constant) with the failure rate λ_p [8, 11]; the failure rate of HW follows an exponential law with a failure rate of λ_a [8, 12]. The failures of the SW and the HW of each channel are independent; studies are conducted until the first failure without taking into account recovery.

There is a relationship between the probability of failure and the failure rate $\lambda(t)$, which, when $\lambda(t) = \lambda = const$, is characterized by the following relation $P(t) = e^{-\lambda t}$.

Finding the probability of fault-free operation begins [11] with dividing the entire set of RCPU states into two subsets Z and M.

Z – the complete set of all operable states of RCPU;

Z_i – serviceable state, $Z_i \in Z$;

M – the complete set of all inoperable states of RCPU;

M_j – failure state, $M_j \in M$.

The optimal redundancy (redundancy introduction) is to achieve the maximum reliability index, provided that some restrictions are not exceeded due to the requirements of the technical task (RoT).

IV. RELIABILITY MODELS OF RCPU

A. RELIABILITY RCPU FLOWCHARTS

For each of the variants of duplicated systems (with single-version SW and HW and multi-version SW and HW), reliability structural diagrams are built (Figs. 2 – 5) depending on the number of versions that affect the system's performance. This is the basis for developing models for assessing the reliability of RCPU.

According to the Reliability Block Diagrams (RBD), presented in Figs. 2 – 5, where:

X – input;

Y – output;

CDT – Control and diagnostic tools;

$p_p(t)$ – PF until the first failure of the first version of the SW;

$p_{p1}(t)$ – PF until the first failure of the second version of the SW;

$p_a(t)$, $p_{a1}(t)$ – PF for the different versions of the HW (one verson and two versions).

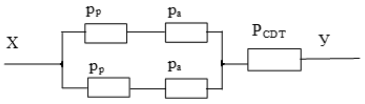


Figure 2. RBD of two-channel RCPU with two equally reliable SW versions (their PF $p_p(t)$) and equally reliable channels HW (whose PF $p_a(t)$) (St1)

Using the logical-probabilistic method [12], the possible states of dual-channel RCPU are investigated (assuming the condition of system operability when at least one of the two channels is in working condition) and analytical dependencies are obtained to determine the PF of single- and two-version dual-channel RCPU before the first failure.

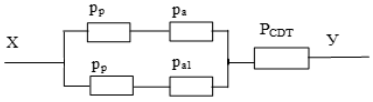


Figure 3. RBD of two-channel RCPU with two equally reliable SW versions (their PF $p_p(t)$) and unequally reliable HW channels (their PF $p_a(t)$, $p_{a1}(t)$, respectively) (St2).

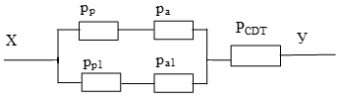


Figure 4. RBD of two-channel RCPU with two not equally reliable SW versions and HW channels (St3).

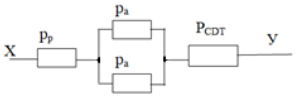


Figure 5. RBD of two-channel RCPU with one SW version (their PF p(t)) and equally reliable channels HW (whose PF $p_p(t)$) and equally reliable HW channels (whose PF $p_a(t)$) (St4).

According to the structural reliability schemes of RCPU, using the logical-probabilistic method of reliability assessment, formulas for determining their PF were obtained:

$$P1(t) = (2 \cdot p_a(t) \cdot p_p(t) - p_a(t)^2 \cdot p_p(t)^2) \cdot P_{CDT},$$

$$P2(t) = (-(p_a(t)^2 \cdot p_p(t) \cdot p_{p1}(t)) + p_a(t) \cdot p_{p1}(t) + p_a(t) \cdot p_p(t)) \cdot P_{CDT}$$

$$P3(t) = \left(-\left(p_a \cdot p_{a1} \cdot p_p \cdot p_{p1}\right) + p_{a1} \cdot p_{p1} + p_a \cdot p_p\right) \cdot P_{CDT},$$

$$P4(t) = (2 - p_a(t)) \cdot p_a(t) \cdot p_p(t) \cdot P_{CDT},$$

$$P(t) = p_a(t) \cdot p_p(t) \cdot P_{CDT},$$

where P(t) – PF of the RCPU reliability structure with the one HW channel and the one SW version (St0); P1(t) – PF of the

RCPU reliability structure with the two equal HW channels and two SW versions (St1); P2(t) – PF of the RCPU reliability structure with the two equal HW channels and two not equal SW versions (St2); P3(t) – PF of the RCPU reliability structure with the two not equal HW channels and two not equal SW versions (St3); P4(t) – PF of the RCPU reliability structure with the two equal HW channels and the one SW version (St4); PCDT(t) – PF of Control and Diagnostic Tools.

In the case of the exponential law of time distribution of the HW and the SW, we can write down the analytical dependencies for determining the PF structures as follows:

$$P(t) = e^{-\lambda_a \cdot t} \cdot e^{-\lambda_p \cdot t} \cdot e^{-\lambda_{CDT} \cdot t},$$

$$P1(t) = 2 \cdot e^{-\lambda_a \cdot t} \cdot e^{-\lambda_p \cdot t} \cdot \left(2 - e^{-\lambda_a \cdot t} \cdot e^{-\lambda_p \cdot t}\right) \cdot e^{-\lambda_{CDT} \cdot t},$$

$$P2(t) = \left(-e^{-2 \cdot \lambda_a \cdot t} \cdot e^{-\lambda_p \cdot t} \cdot e^{-\lambda_{p1} \cdot t} + e^{-\lambda_a \cdot t} \cdot e^{-\lambda_{p1} \cdot t} + e^{-\lambda_a \cdot t} \cdot e^{-\lambda_p \cdot t}\right) \cdot e^{-\lambda_{CDT} \cdot t},$$

$$P3(t) = \left(-e^{-\lambda_a \cdot t} \cdot e^{-\lambda_{a1} \cdot t} \cdot e^{-\lambda_p \cdot t} \cdot e^{-\lambda_{p1} \cdot t} + e^{-\lambda_{a1} \cdot t} \cdot e^{-\lambda_{p1} \cdot t} + e^{-\lambda_a \cdot t} \cdot e^{-\lambda_p \cdot t}\right) \cdot e^{-\lambda_{CDT}(t) \cdot t},$$

$$P4(t) = e^{-\lambda_a \cdot t} \cdot e^{-\lambda_p \cdot t} \cdot \left(2 - e^{-\lambda_a \cdot t}\right) \cdot e^{-\lambda_{CDT} \cdot t},$$

where λ_p is the SW failure rate of the first (main) channel; λ_{p1} is the failure rate of the second (backup) channel; λ_a is the failure rate of the first (main) channel's automatic equipment; λ_{a1} is the failure rate of the second (backup) channel; λ_{CDT} is the CDT failure rate.

B. SIMULATION RESULTS

Let us simulate the obtained mathematical models. The obtained graphical dependencies are shown in Fig. 6 – 14.

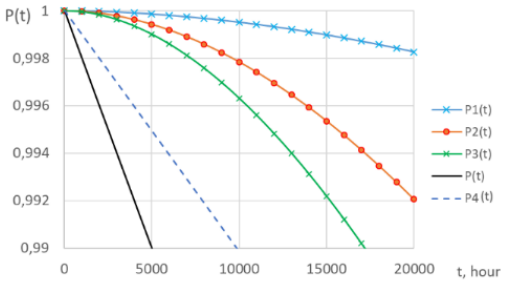


Figure 6. Graphical dependencies of PF structures RCPU St0, St1, St2, St3, St4 on time until the failure λ_a=10^{-6} 1/h, λ_p=10^{-6} 1/h, λ_{a1}=10^{-5} 1/h, λ_{p1}=10^{-5} 1/h, λ_{CDT}=10^{-8} 1/h.

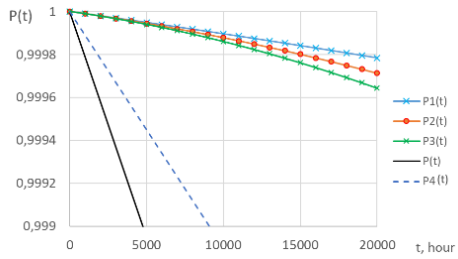


Figure 7. Graphical dependencies of PF structures RCPU St0, St1, St2, St3, St4 on time until the failure λ_a = 10^{-7} 1/h, λ_p=10^{-7} 1/h, λ_{a1}=10^{-6} 1/h, λ_{p1} = 10^{-6} 1/h, λ_{CDT}=10^{-8} 1/h.

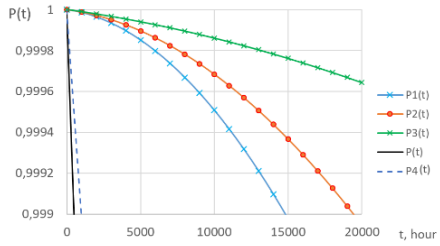


Figure 8. Graphical dependencies of PF structures RCPU St0, St1, St2, St3, St4 on time until the failure λ_a= 10^{-6} 1/h, λ_p=10^{-6} 1/h, λ_{a1}=10^{-7} 1/h, λ_{p1}=10^{-7} 1/h, λ_{CDT}=10^{-8} 1/h.

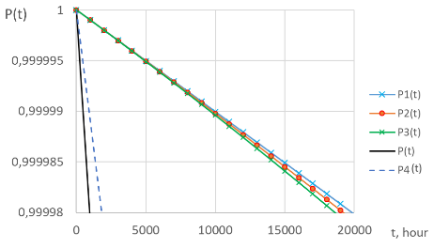


Figure 9. Graphical dependencies of PF structures RCPU St0, St1, St2, St3, St4 on time until the failure λ_a = 10^{-8} 1/h, λ_p=10^{-8} 1/h, λ_{a1}=10^{-7} 1/h, λ_{p1}=10^{-7} 1/h, λ_{CDT} =10^{-9} 1/h.

The interest is the study of how impacts on PF of RCPU structures PF of the first (main channel) SW version (Fig. 10-14).

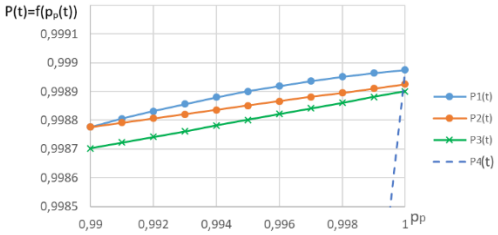


Figure 10. Graphical dependencies of the PF of RCPU structures St1, St2, St3, St4 on the PF of the SW of the first (main) channel if p_a(t)=0,995; p_{a1}(t)=0,99; p_{p1}(t)=0,99; P_{CDT}(t) = 0,999.

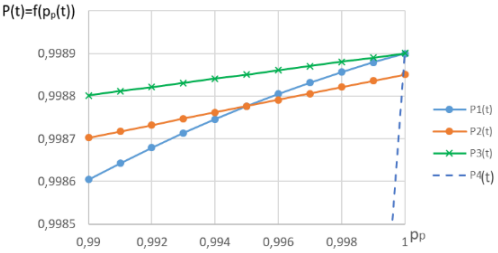


Figure 11. Graphical dependencies of the PF of RCPU structures St1, St2, St3, St4 on the PF of the SW of the first (main) channel if p_a(t)=0,99; p_{a1}(t)=0,995; p_{p1}(t)=0,995; P_{CDT}(t)=0,9999.

For a more overview view, how the reliability of a single-version two-channel RCPU structure with equally reliable channels changes with changing values of the main channel's PF compared to two-version PF and equally reliable and unequally reliable HW is presented in Fig. 12.

The smaller the main channel's PF of the main channel, the less reliable RCPU becomes and does not allow to achieve the values required by the RoT ($P_r \geq 0,99999$).

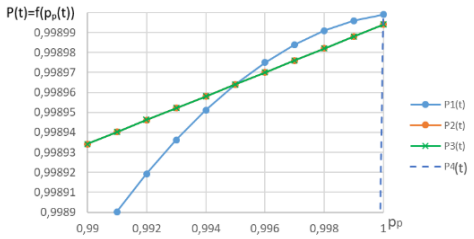


Figure 12. Graphical dependencies of the PF of RCPU structures St1, St2, St3, St4 on the PF of the SW of the first (main) channel if $p_a(t) = 0,999$; $p_{a1}(t) = 0,999$; $p_{p1}(t)=0,995$; $P_{CDT}(t) = 0,999$.

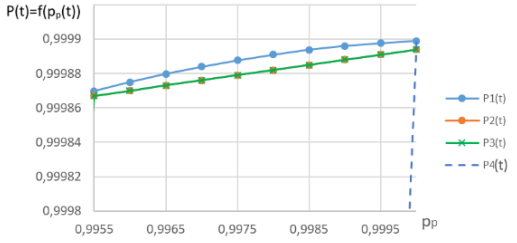


Figure 13. Graphical dependencies of the PF of RCPU structures St1, St2, St3, St4 on the PF of the SW of the first (main) channel if $p_a(t) = 0,999$; $p_{a1}(t) = 0,999$; $p_{p1}(t)=0,995$; $P_{CDT}(t) = 0,9999$.

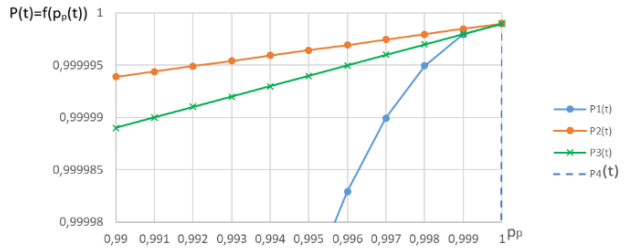


Figure 14. Graphs of the dependence (a, b) of the PF of RCPU structures St1, St2, St3, St4 on the PF of the SW of the first (main) channel if $p_a(t)=0,99999$; $p_{a1}(t)=0,9995$; $p_{p1}(t)=0,9995$; $P_{CDT}(t) = 0,999999$.

The difference between those shown in Fig. 6-14 graphical dependencies is that the simulation of the models is carried out with different values of reliability indicators of the subsystems of the RCPU. Analysis of the graphical dependence 6 – the highest value of probability of failure-free operation is given by the structure St1, and when the value of the operating time of the RCPU changes from 0 to 20000 hours, its PF falls from the value of 1 to 0.999. St2 shows a slightly worse value of PF: P(t) of 1 to 0.992 for 20000 hours of operation. The worst value of PF we have at realization of the structure St0, which decrease within 1...0.99 for 5000 hours of operation of RCPU. Analysis of the graphical dependence 7 – the highest value of PF is given by the structure St1, and when changing the value of time of operation of the RCPU from 0 to 20000 hours, its PF falls from the value of 1 to 0.9998.

St2 shows a slightly worse value of PF: P(t) = 1...0.9997 for 20000 hours of operation. The worst value of PF we have at realization of structure St0, which falls within the limits of 1...0.999 for 5000 hours of operation of RCPU. The value of PF at realization of structure St4, falls within the limits of 1...0.999 for 9000 hours of operation of RCPU.

The analysis of the graphical dependence Fig. 8 – the highest value of PF is given by the structure St3, and at change of value of time of operation of the RCPU from 0 to 20000 hours its PF falls from value 1 to 0.99965. St2 shows a slightly worse value of PF: P(t) = 1...0.9999 for 18500 hours of operation. The worst value of PF we have at realization of structure St0, which falls within 1...0.999 for 14800 hours of operation of the RCPU.

Having carried out a similar analysis of modelling results for all graphical dependencies obtained with different initial data, we make tables (Table 1, Table 2) with priority series. These priority series allow us to choose the most reliable structure of the RCPU for the given initial data.

Table 1. Priority series of RCPU structures when changing the values of time between failures

Initial data, 1/h	Interval of time, h	Priority of structures				
		1	2	3	4	5
$\lambda_a=10^{-6}$, $\lambda_p=10^{-6}$, $\lambda_{a1}=10^{-5}$, $\lambda_{p1}=10^{-5}$, $\lambda_{CDT}=10^{-8}$	$0\leq t\leq 20000$	St1	St2	St3	St4	St0
$\lambda_a=10^{-7}$, $\lambda_p=10^{-7}$, $\lambda_{a1}=10^{-7}$, $\lambda_{p1}=10^{-6}$, $\lambda_{CDT}=10^{-8}$	$0\leq t\leq 20000$	St1	St2	St3	St4	St0
$\lambda_a=10^{-6}$, $\lambda_p=10^{-6}$, $\lambda_{a1}=10^{-7}$, $\lambda_{p1}=10^{-7}$, $\lambda_{CDT}=10^{-8}$	$0\leq t\leq 20000$	St3	St2	St1	St4	S0
$\lambda_a =10^{-8}$, $\lambda_p=10^{-8}$, $\lambda_{a1}=10^{-7}$ $\lambda_{p1}=10^{-7}$, $\lambda_{CDT}=10^{-9}$	$0\leq t\leq 20000$	St1	St2	St3	St4	St0

Table 2. Priority series of RCPU structures when changing the SW PF values of the first HW channel

Initial data, PF values	Intervals of values	Priority of structures			
$p_a(t) = 0,995$; $p_{a1}(t) = 0,99$; $p_{p1}(t) = 0,99$; $P_{CDT}(t) = 0,999$	$0,99<p_p(t)\leq 1$	St1	St2	St3	St4
	$p_p(t) = 0,99$	St1, St2	St3	St4	
$p_a(t) = 0,99$; $p_{a1}(t) = 0,995$; $p_{p1}(t) = 0,995$; $P_{CDT}(t) = 0,999$	$0,99\leq pp(t)<0,995$	St3	St2	St1	St4
	$0,995<p_p(t)\leq 1$	St3	St1	St2	St4
	$p_p(t) = 0,995$	St3	St2, St1	St4	
$p_a(t) = 0,999$; $p_{a1}(t) = 0,999$; $p_{p1}(t) = 0,995$; $P_{CDT}(t) = 0,999$	$0,99\leq p_p(t)<0,995$	St2, St3	St1	St4	
	$0,995< p_p(t) \leq 1$	St1	St2, St3	St4	
	$p_p(t) = 0,995$	St1, St2, St3	St4		
$p_a(t) = 0,999$; $p_{a1}(t) = 0,999$; $p_{p1}(t) = 0,995$; $P_{CDT}(t) = 0,9999$	$0,9955\leq p_p(t) \leq 1$	St1	St2, St3	St4	
$p_a(t) = 0,99999$; $p_a(t) = 0,9995$; $p_{p1}(t) = 0,9995$; $P_{CDT}(t)=0,999999$	$0,999\leq p_p(t) \leq 1$	St2	St1, St3	St4	

The analysis of the modelling results and graphical dependencies showed that the PF of the main HW channel $p_p(t)$ strongly influences the PF of the RCPU. With high reliability

of SW single channel ($p_p(t) \geq 0,9999$), in some cases, the application of unequal reliability of channels and PF is inappropriate.

Fig. 14 shows that high PF CDT (($P_{CDT}(t) = 0,999999$) and $p_a(t)=0,99999$)) can achieve the required values ($P_r(t) \geq 0,99999$) with reliability $p_p(t) \geq 0,997$.

V. CHOICE OF THE RCPU STRUCTURE USING PRIORITY SERIES

The method of improving the reliability of RCPU on the basis of choice of the RCPU structure using priority series that includes the following steps:

1. Analyzing the requirements of the RoT for the reliability of RCPU of a particular router model.
2. System analysis of RCPU architecture to understand whether it is possible to introduce diversity of SW and redundancy of HW in this device. This requires analyzing the types of memory where the SW versions can be stored, the memory capacity, the ability to access the cloud to load an equally reliable or unequally reliable with the first (main) version of the SW.
3. Selection of RCPU reliability indicators. The following indicators were selected: probability of failure-free operation, failure rate, mean time between failures.
4. Modelling and evaluation of the main reliability indicators. This method proposes the use of the logical-probabilistic method of estimating the PF, on the basis of which modelling was carried out.
5. Obtain the mathematical models for estimation of PF for several types of RCPU structures: single-channel single-version, two-channel double-version with equally reliable versions of HW and SW, two-channel double-version with equally reliable versions of HW and unequally reliable versions of SW, two-channel double-version with unequally reliable versions of HW and SW, two-channel HW single-version SW structure.
6. Simulation of the obtained models and construction of graphical dependencies for each type of structures.
7. Creation of priority series of RCPU structures for different initial data on the basis of studies of graphical dependencies.
8. Selection of the RCPU structure with the highest PF value, which is suitable for implementation in a particular router model, using priority series tables.

The algorithm of selection of RCPU reservoir structures is shown in Fig. 15.

Let's assess the accuracy of hypothesis of exponential distribution law for the HW and SW of RCPU. Difference between characteristics values is: $\Delta = Z - R$ (Fig.16).

To determine the reliability of the results obtained, the Kolmogorov-Smirnov criterion is used [26], which states: that whatever the distribution function of a continuous random variable, with an unlimited increase in the number of independent experiments (N), the probability of inequality $D\sqrt{N} \geq \alpha$ tends to the limit

$$P = 1 - \sum_{n=0}^{N}(-1)^n \cdot e^{-2 \cdot n^2 \cdot \alpha^2}.$$

A comparative analysis of the results of the simulation of a simulation model of a two-channel single-version architecture with an exponential distribution of the flow of failures in time and analytical calculations (Fig. 16) showed that, according to the Kolmogorov-Smirnov criterion, the accuracy of the calculations (the maximum value of the spread of values between dependencies) is $D = 3.2 \times 10^{-3}$; $\alpha = 0.186$ per 10,000 hours of device operation.

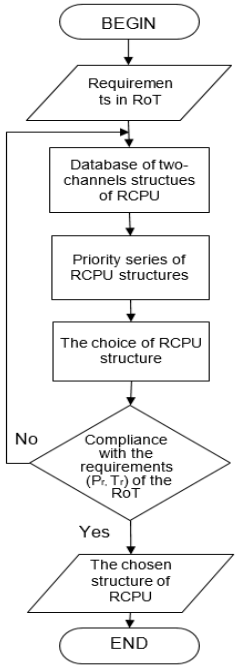


Figure 15. An algorithm for selecting of RCPU structures based on the formation of priority series, where Pr is the PF required for RCPU according to the system specification; Tr is the required uptime according to the system specification (RoT).

Based on the table of probability values P(α), it was determined that with the obtained value $1 - P(\alpha) \approx 0.99$. The likelihood that, due to random reasons, the maximum discrepancy between the characteristics is large for the selected parameters, therefore, the research results for the St4 architecture (dual-channel single-version) are with high accuracy.

Thus, the hypothesis about the exponential law of distribution of the flow of failures over time in HW and SW is plausible.

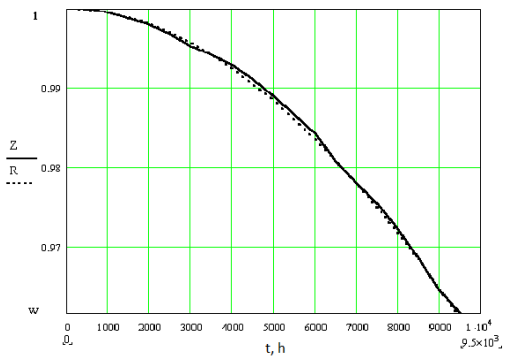


Figure 16. Graph of the dependence of the PF reserved HW and two-versions RCPU on time, obtained using imitation simulation (Z) and analytical (R) modeling under the condition of exponential distribution of the failure flow of HW and SW and St4 architecture.

VI. DISCUSSIONS

The analysis of graphical dependencies and priority series has shown that it is inexpedient to introduce unequal reliability of HW (i.e. HW from different manufacturers) into RCPU.

However, the application of redundancy of HW channels and diversity of SW leads to a significant increase in PF.

However, in all RCPU structures, the PF can reach the required values only at high reliability of control and diagnostics means, their PF should be not less than 0.9999. Therefore, methods of reliability improvement should be applied, including SW diversity and duplication of HW channels.

VII. CONCLUSIONS

The main contribution of the research is a set of reliability models for single-type and different-type of RCPU diversity structures and a methodology for their selection, taking into account the requirements for RCPU reliability, priority series for selecting the reliability structure most suitable for a given router, obtained during the study of these reliability models. The results are based on the system analysis of HW-SW structures, requirements to the router reliability indicators and reliability block diagrams, taking into account the possibility and necessity of SW diversity and application of HW redundancy.

Practical application of the proposed method of increasing RCPU reliability is possible when designing new or renovating obsolete routers, as well as modern models of routers it is possible to use the second version of SW at the expense of additional types of memory of sufficient size. Priority series allow to determine the feasibility of dual-channel HW redundancy and SW diversity under different operating conditions and RCPU reliability requirements.

Further studies of reliability models considering attacks on RCPUs are currently underway. These studies should help in selecting the RCPU structure most resistant to attacks.

References

[1] M. Kolisnyk, V. Kharchenko, I. Piskachova, *"Investigation of the Smart Business Center for IoT Systems Availability Considering Attacks on the Router,"* Dependable IoT for Human and Industry. New York, 2022. pp. 169–195. https://doi.org/10.1201/9781003337843-11.

[2] O. Duda et al., "Data Processing in IoT for Smart City Systems," *2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, Metz, France, 2019, pp. 96-99, https://doi.org/10.1109/IDAACS.2019.8924262.

[3] A. Avizienis, J.-C. Laprie, "Reliability computing: From concepts to design diversity," *Proceeding of the IEEE*, vol. 74. issue 5, pp. 629-638, 1986. https://doi.org/10.1109/PROC.1986.13527.

[4] B. Littlewood, L. Strigini, "Software reliability and reliability: a roadmap," *Proceedings of the ACM Conference on the Future of Software Engineering (ICSE'00)*, May 2000, pp. 175–188. https://doi.org/10.1145/336512.336551.

[5] B. Littlewood, P. T. Popov, L. Strigini, N. Shryane, "Modelling the effects of combining diverse Software fault detection techniques," *IEEE Transactions on Software Engineering*, vol. 26. issue 12, pp. 1157–1167, 2000. https://doi.org/10.1109/32.888629.

[6] L. Strigini, "Fault tolerance and resilience: Meanings, measures and assessment," In: K. Wolter, A. Avritzer, M. Vieira, A. Van Moorsel, (Eds), *Resilience Assessment and Evaluation of Computing Systems*, Springer, Berlin, Heidelberg, November 2012, pp. 3–24. https://doi.org/10.1007/978-3-642-29032-9_1.

[7] P. Popov, L. Strigini, A. Romanovsky, "Diversity for off-the-shelf components," *Proceeding of the International Conference on Dependable Systems & Networks (FTCS-30, DCCA-8) Fast Abstracts*, New York, USA, 2000, pp. B60-B61. https://doi.org/10.1109/TDSC.2007.70208.

[8] V. Kharchenko, Y. Ponochovnyi, I. Babeshko, E. Ruchkov, A. Panarin, "Safety assessment of maintained control systems with cascade two-version 2oo3/1oo2 structures considering version faults," In: Zamojski W., Mazurkiewicz J., Sugier J., Walkowiak T., Kacprzyk J. (eds) *Dependable Computer Systems and Networks. DepCoS-RELCOMEX Lecture Notes in Networks and Systems*, 2023, vol. 737. Springer, Cham. https://doi.org/10.1007/978-3-031-37720-4_11.

[9] W. Kuo, M. J. Zuo, *Optimal reliability modeling: principles and applications*, John Wiley & Sons, Book, 2003, 543 p.

[10] L. Ozirkovskyy, B. Volochij et al., "Functional safety analysis of safety-critical system using state transition diagram," *Radioelectronic and Computer Systems*, vol. 2, pp. 145-158, 2022. https://doi.org/10.32620/reks.2022.2.12.

[11] A. Yanko, V. Krasnobayev, A. Martynenko, "Influence of the number system in residual classes on the fault tolerance of the computer system," *Radioelectronic and Computer Systems*, vol. 3, pp. 159-172, 2023. https://doi.org/10.32620/reks.2023.3.13.

[12] M. O. Kolisnyk, I. V. Piskachova, *Reliability of Software Features of Microprocessor Devices for Controlling Telecommunication Systems*, Kharkiv: UkrDAZT, 2012, 167 p.

[13] C. Guo, S. Zhou, J. Li, F. Chen, D. Li and X. Huang, "A Novel software reliability growth model of safety-critical software considering fault severity classification," *Proceeding of the 2019 4th International Conference on System Reliability and Safety (ICSRS)*, Rome, Italy, 2019, pp. 25-29, https://doi.org/10.32620/reks.2023.3.13.

[14] M. Frihi, M. Boutalbi, S. Toumi, C. Tanougast and M. Heil, "Optimized and dependable router suitable for dynamic networks on chip," *Proceeding of the 2014 International Conference on Control, Decision and Information Technologies (CoDIT)*, Metz, France, 2014, pp. 783-788, https://doi.org/10.1109/CoDIT.2014.6996997.

[15] R. Kammerer, R. Obermaisser, B. Frömel, "A router for the containment of timing and value failures in CAN," *EURASIP Journal on Embedded Systems*, article 4, 2012. https://doi.org/10.1186/1687-3963-2012-4.

[16] H. Ghareb, Software redundancy in Siemens PLC – Hardware versus Software [Online]. Available at: https://instrumentationtools.com/SW-redundancy-in-siemens-plc/.

[17] *Cisco Catalyst IR8300 Rugged Series Router Data Sheet*. March 23, 2022. [Online]. Available at: https://www.cisco.com/c/en/us/products/collateral/routers/catalyst-ir8300-rugged-series-router/nb-06-cat-ir8340-rugged-ser-rout-ds-cte-en.html.

[18] *Juniper networks. Routers.* [Online]. Available at: https://www.juniper.net/us/en/products/routers/mx-series/mx304-universal-routing-platform.html.

[19] *Fortinet. Router. Fortigate 200f-series datasheet.* 2023. 10 p. [Online]. Available at: https://www.fortinet.com/content/dam/fortinet/assets/data-sheets/fortigate-200f-series.pdf.

[20] D. Liew, D. Schemmel, C. Cadar, A. F. Donaldson, R. Zahl and K. Wehrle, "Floating-point symbolic execution: A case study in N-version programming," *Proceeding of the 2017 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE)*, Urbana, IL, USA, 2017, pp. 601-612, https://doi.org/10.1186/1687-3963-2012-4.

[21] C. Cadar and P. Hosek, "Multi-version software updates," *Proceeding of the 2012 4th International Workshop on Hot Topics in Software Upgrades (HotSWUp)*, Zurich, Switzerland, 2012, pp. 36-40, https://doi.org/10.1109/HotSWUp.2012.6226615.

[22] N. Subasi, U. Guner, I. Ustoglu, "N-version programming approach with implicit safety guarantee for complex dynamic system stabilization applications," *Measurement and Control*, vol. 54, issues 3-4, pp. 269-278, 2021. https://doi.org/10.1177/0020294019887473.

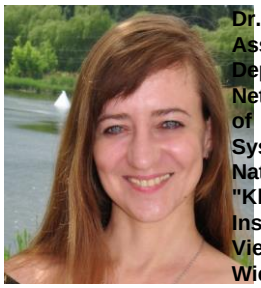
[23] F. Machida, "N-version machine learning models for safety critical systems," *Proceedings of the 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, Portland, OR, USA, 2019, pp. 48-51, https://doi.org/10.1109/DSN-W.2019.00017.

[24] V. Girdhar and E. Al-Masri, "N-version programming for enhancing fault tolerance in fog-based IoT systems," *Proceedings of the 2020 6th International Conference on Science in Information Technology (ICSITech)*, Palu, Indonesia, 2020, pp. 109-114, https://doi.org/10.1109/ICSITech49800.2020.9392033.

[25] Z. Jelinski and P. B. Moranda, "Software Reliability Research", *Freiberger, W. Ed., Statistical Computer Performance Evaluation,*

Academic Press, New York, 1972, pp. 465-484. https://doi.org/10.1016/B978-0-12-266950-7.50028-1.

[26] Kolmogorov-Smirnov Goodness-of-Fit Test. [Online]. Available at: https://www.itl.nist.gov/div898/handbook/eda/section3/eda35g.htm.



Dr. MARYNA KOLISNYK, PhD, an Associate Professor of the Department of Computer Systems, Networks and Cybersecurity, Faculty of Radio Electronics, Computer Systems and Infocommunications, National Aerospace University "KhAI", Guest Professor of the Institute of Telecommunications, Vienna University of Technology (TU Wien). Field of scientific interests: Dependability of software and hardware of microprocessor control systems; cybersecurity of internet of things (IoT) and industrial IoT (IIoT) systems, Predictive analytics, Machine Vision, Extended Reality.

Dr. IRYNA PISKACHOVA, PhD, an Associate Professor of the Department of Automation and Computer-Integrated Technologies, Cyberport Institute of the State Biotechnology University. Field of scientific interests: Reliability of software and hardware of micro-processor control systems.

Prof. VYACHESLAV KHARCHENKO, a Doctor of Technical Sciences, Professor, Head of the Department of Computer Systems, Networks and Cybersecurity, Faculty of Radio Electronics, Computer Systems and Infocommunications, National Aero-space University "KhAI". Field of scientific interests: Software quality and reliability; Functional and cyber security of information management systems and critical infrastructures; Green information technology; Critical computing; Augmented reality