ISSN 0005-1179, Automation and Remote Control, 2022, Vol. 83, No. 2, pp. 264–272. © Pleiades Publishing, Ltd., 2022.
Russian Text © The Author(s), 2022, published in Avtomatika i Telemekhanika, 2022, No. 2, pp. 122–132.

CONTROL IN TECHNICAL SYSTEMS

Approximate Evaluation of the Efficiency of Synchronous and Self-Timed Methodologies in Problems of Designing Failure-Tolerant Computing and Control Systems

I. A. Sokolov[1*], Yu. A. Stepchenkov[1], Yu. V. Rogdestvenski[1***], and Yu. G. Diachenko[1****]**

[1] *Federal Research Center "Computer Science and Control," Russian Academy of Sciences, Moscow, 119333 Russia*

*e-mail: *ISokolov@ipiran.ru, ** YStepchenkov@ipiran.ru, *** YRogdest@ipiran.ru, **** diaura@mail.ru*

Received August 2, 2021; revised September 18, 2021; accepted October 15, 2021

Abstract—The paper deals with a comparative analysis of the efficiency of using synchronous and self-timed (ST) methodologies in the design of failure-tolerant computing and control systems based on complementary metal–oxide–semiconductor (CMOS) technology. The issues of failure tolerance of technical control means are considered in detail using examples of digital circuits of various types. A significant increase (by a factor of 1.2–1.8) in the time of failure-free operation of ST circuits in comparison with synchronous counterparts is confirmed. The most significant features of ST circuitry, which provide an increase in the failure tolerance of ST systems, are highlighted. Circuitry methods are proposed for increasing the failure tolerance of ST control systems, increasing the time of failure-free operation of combinational ST circuits up to 4.0 times and sequential ST circuits up to 7.1 times.

Keywords: hardware, failure tolerance, failure, synchronous circuit, self-timed circuit, dual-rail signal, C-element, indication

DOI: 10.1134/S0005117922020084

1. INTRODUCTION

The solution of practical computational type and control problems is associated with the processing of a large amount of data for a long time. With an increase in operating time, the risk of a failure in the hardware of the computing and control system also increases due to changing operating conditions and the impact of adverse factors. Therefore, the hardware of computing and control systems must be effectively resistant to failures.

For mass-produced microprocessor chips, the failure occurrence rate in digital circuits is several orders of magnitude higher than the failure occurrence rate ($\sim 10^{-4}$ 1/h versus $\sim 10^{-7}$ 1/h) [1]. According to Viktorova et al. [1]: "Creation of the latest technologies and methods for increasing the yield of suitable very large scale integration circuits (VLSI) allows one to assume the aggravation of the existing gap and, as a result, making failures the predominant factor determining the reliability" of digital circuits. Improving the failure tolerance of computer systems is possible through the use of hardware methods for monitoring their performance and self-repair. The most advantageous in this respect is the self-timed (ST) methodology [2–5].

Null Convention Logic (NCL) circuits [5] are one of the well-known subclasses of ST circuits. NCL circuits use dual-rail inputs, outputs, and internal signals with zero spacer. The basic cell

library for designing NCL circuits includes only 29 so-called multithreshold cells, each of which indicating all of its inputs. This property is the "know-how" of the NCL methodology and simplifies the design of delay-insensitive ST circuits. The disadvantages of this methodology include significant hardware redundancy and the associated increased power consumption [6].

There are other approaches to the problem of failure tolerance of hardware for computing and control systems [7–9]. Lodhi et al. [7] propose a macro-synchronous micro-asynchronous pipeline (MSMA) method using a logic failure-tolerant and low-power version of the asynchronous NCL circuit. However, this solution is not an ST circuit. A Triple Core Lock-Step (TCLS) ARM™ Cortex™-R5 processor is described in [8, 9]. The TCLS architecture includes three Central Processing Units (CPUs) and provides system-level reliability. The proposed solution assumes that "individual CPUs do not need to be failure-tolerant, and can be implemented using commercial technology process" [8]. However, all the considered publications do not give quantitative estimates of the level of failure tolerance of real products.

The approach proposed in [2, 3] solves the indicated problem. The approach is based on a failure-tolerant ST paradigm that allows using a wide range of library cells and unary and biphase signal redundant coding in addition to dual-rail coding. The use of heterogeneous coding ensures the development of ST circuits that are simpler compared with NCL counterparts. The failure-tolerant ST methodology ensures the preservation of the operability of the technical means of the computing and control complex built on this basis in a wide range of supply voltage and ambient temperature, the detection and localization of permanent failures arising due to adverse effects, including radiation factors, and the increase in the service life of failure-free operation.

Owing to the presence of a global clock tree, synchronous circuits often perceive a failure as a fault and require the use of excessive hardware redundancy to fend off such failures. Due to inherent hardware redundancy and bi-phase operation, ST circuits are able to distinguish between a momentary failure and a fault. In addition, they have a higher natural level of failure tolerance than their synchronous counterparts.

The purpose of the article is to produce an approximate estimate for the efficiency of the ST methodology in designing hardware of computing and control systems operating under the influence of destabilizing factors and resistant to short-term failures.

The present paper analyzes the stability of digital circuits to short-term failures, the duration of which ranges from hundreds of picoseconds to units of nanoseconds [10]. In combinational circuits, they self-annihilate, but in circuits with memory (flip-flops, registers), a failure can invert a stored bit of information and become permanent. The article does not consider remote and replacement types of failures [2], the probability of which in practical ST circuits is extremely low. The term "failure tolerance" is interpreted as the ability of a piece of hardware to continue its correct operation without stopping by masking a failure or after some delay associated with waiting for the failure to self-annihilate without damaging the processed data.

2. COMPARING CLASSICAL SYNCHRONOUS AND SELF-TIMED REALIZATIONS

The design and manufacture of hardware for computing and control systems presuppose their correct operation under specified operating conditions for a certain time interval in the environment with a specified level of failure density. Almost always, this is an optimization problem that minimizes the "risk–cost" criterion under given constraints on the operating conditions and required

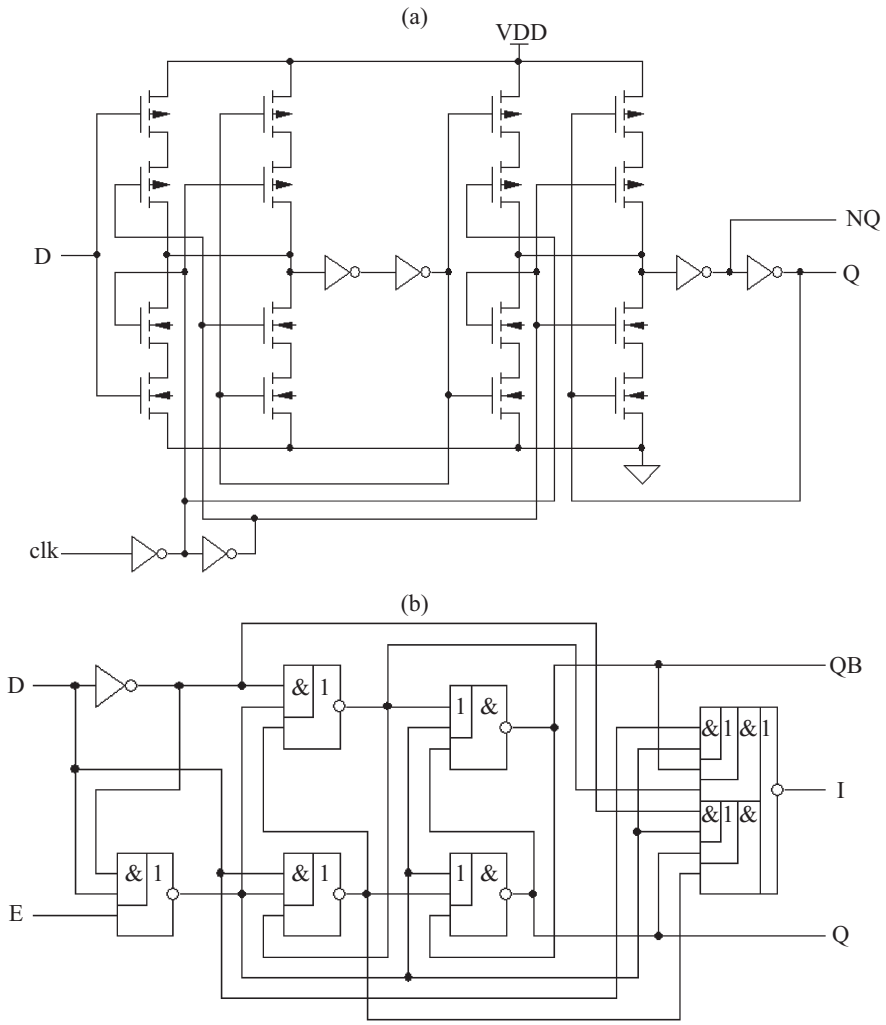

Fig. 1. D–flip-flop: (a) synchronous CMOS-transistor circuit; (b) ST circuit.

technical characteristics. One way to achieve an optimal result in solving this problem is to use the ST methodology.

The self-timed methodology assumes full control over the completion of the switching of all circuit cells whose output state change was initiated by the current set of input data [2]. Acknowledging the completion of the switching of all active circuit cells is a prerequisite for starting the processing of the next set. This control is performed by special indication circuits that collect complete information about the end of the processing of a data set. The interaction of ST circuits is based on the request-acknowledge relationship of digital units in the data processing path. Each functional ST block uses information from the next ST block about its readiness to process new data and generates a signal about the completion of its work for the previous ST block. Special ST data coding is used [2] to organize such a process.

The simplest and most common ST code is a dual-rail code, where the information bit “0” is represented by the combination “01”; the information bit “1” is represented by the combination “10”; and a special state—a spacer—is introduced (“00” or “11”). The full operation cycle of the circuit consists of two alternating phases—working and spacer ones.

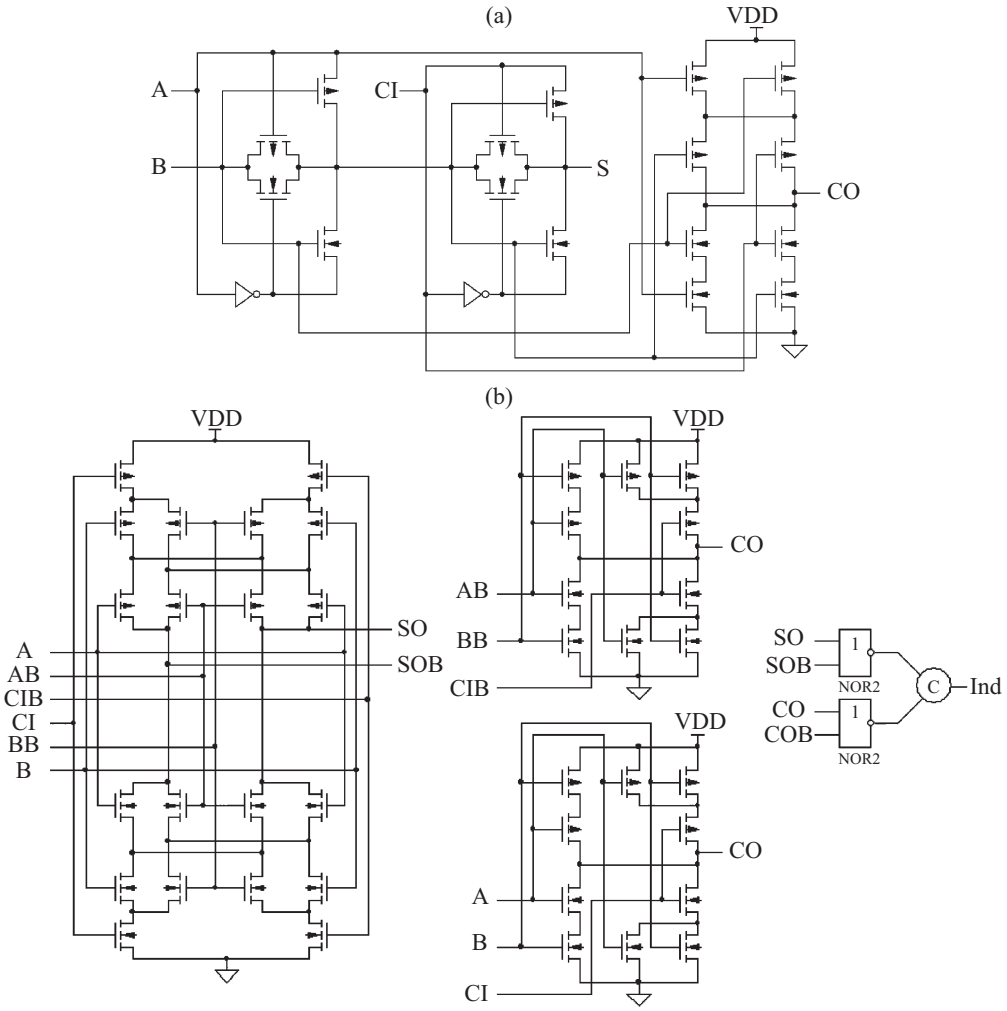

Fig. 2. Binary adder: (a) synchronous circuit; (b) ST circuit.

Owing to this data processing organization, the ST circuit acquires the following properties that distinguish it from its synchronous counterpart:

- Speed, determined only by delays in circuit cells.
- Natural 100% self-testing and self-diagnosing in relation to multiple constant failures.
- Operational safety based on test-free failure localization.
- The maximum possible area of operation determined only by the physical retention of the switching properties of active elements [3].

The disadvantages of ST circuitry include the hardware redundancy of ST circuits in relation to synchronous counterparts due to excessive signal coding and the presence of an indication subcircuit. The complexity of ST circuits turns out to be greater than the complexity of synchronous counterparts up to 1.7 times for circuits with memory and up to 3.0 times for combinational circuits. Figure 1 demonstrates examples of D–flip-flop implementation in synchronous (Fig. 1a) and ST (Fig. 1b) cases. The ST flip-flop circuit contains 1.7 times more complementary metal–oxide–semiconductor (CMOS) transistors than the synchronous counterpart. To a large extent, this is due to the circuit containing an indication element that forms output I.

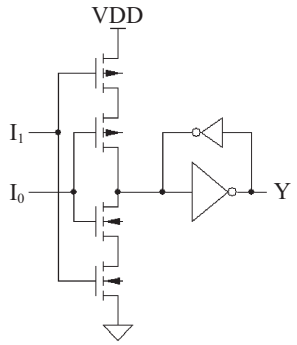

Fig. 3. Schematic diagram of C-element.

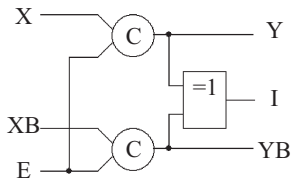

Fig. 4. ST storage register bit with C-elements.

Figure 2 shows the circuit implementation of a single-bit binary adder in synchronous (Fig. 2a) and ST (Fig. 2b) bases. The diagram of the C-element that forms the indication output of the adder is depicted in Fig. 3. The complexity of the ST CMOS-transistor circuit is 2.8 times greater than that of the synchronous implementation.

The ratio of hardware costs of the ST and the synchronous counterpart depends on the circuit type. In some cases, the complexity of ST circuits is comparable to the hardware costs of synchronous counterparts. For example, the storage register bit in the stage of the ST pipeline is usually implemented using C-elements (Fig. 4) and has the same complexity as the synchronous version.

The bi-phase nature of the operation of ST circuits and the presence of an indication subcircuit approximately halves the speed. However, there are techniques that permit one to bring the performance of a number of ST circuits closer to the level of synchronous counterparts [11].

The above-described properties of ST circuits ensure their natural immunity to failures. Studies have shown [12, 13] that, owing to the completeness of the dual-rail coding of information signals, when the state opposite to the spacer is indicated as a spacer, the bi-phase discipline of operation and the indication of all switching cells of the ST circuit are resistant to 90% of single short-term failures.

Single-time means the occurrence of a failure in only one logical cell of the circuit. The cell's circuit may consist of several CMOS transistors. Studies show that the typical effective diameter of the track of a nuclear particle that causes a failure does not exceed 2–2.5 μm [14]. In CMOS technology with design rules of 65 nm, this corresponds to the size of a standard cell. Thus, the physical cause of a failure usually affects several transistors at once. However, the probability of a failure occurring is proportional to the area of the layout implementation of the circuit, which in turn is proportional to the number of transistors in the circuit. Therefore, it is advisable to estimate the level of failure tolerance of the circuit based on the number of transistors in the circuit.

For a given flow density λ_{in} of random events initiating single failures, the failure rate function λ for a circuit is defined as the sum of failure rates of individual components (in this case, transistors) [15, formula (3.11)],

$$\lambda = N\lambda_{\text{in}}\alpha,$$

where N is the number of transistors in the circuit and α is the probability of a failure if one transistor is affected. Since the time of failure-free operation is inversely proportional to the failure intensity [15, formula (3.14)], it follows that the ratio of failure-free operation time (FFOT) for the ST circuit and its synchronous counterpart has the form

$$K_{\text{FFOT}} = \frac{\lambda_{\text{S}}}{\lambda_{\text{ST}}} = \frac{N_{\text{S}}\lambda_{\text{in}}\alpha_{\text{S}}}{N_{\text{ST}}\lambda_{\text{in}}\alpha_{\text{ST}}} = \frac{\alpha_{\text{S}}}{A_{\text{R}}\alpha_{\text{ST}}}, \tag{1}$$

where λ_{S} and λ_{ST} are the failure flow densities in the synchronous and ST circuits; N_{S} and N_{ST} are the numbers of transistors in the synchronous and ST circuits; α_{S} and α_{ST} are the probabilities of a failure caused by affecting one transistor in the synchronous and ST circuits; and $A_{\text{R}} = N_{\text{ST}}/N_{\text{S}}$ is the coefficient of hardware redundancy of the ST circuit in comparison with the synchronous counterpart.

The earlier studies [12, 13] have made it possible to evaluate the probability of a failure in case of affecting one transistor of the synchronous and ST circuits for the combinational ($\alpha_{\text{S}} = 0.5$, $\alpha_{\text{ST}} = 0.1$) and flip-flop ($\alpha_{\text{S}} = 0.5$, $\alpha_{\text{ST}} = 0.24$) circuit types under the assumption that a failure in one transistor causes a failure in the logic cell. In combinational circuits, $A_{\text{R}} = 2.8$ and the ratio of the times of failure-free operation of the ST and synchronous counterparts is

$$K_{\text{FFOT C1}} = \frac{\alpha_{\text{S}}}{A_{\text{R}}\alpha_{\text{ST}}} = \frac{0.5}{2.8 \times 0.1} \approx 1.8. \tag{2}$$

In sequential circuits, $A_{\text{R}} = 1.7$, and for these circuits

$$K_{\text{FFOT S1}} = \frac{\alpha_{\text{S}}}{A_{\text{R}}\alpha_{\text{ST}}} = \frac{0.5}{1.7 \times 0.24} \approx 1.2. \tag{3}$$

Formulas (1)–(3) give only an approximate comparative estimate of the failure tolerance of synchronous and ST circuits. However, they clearly demonstrate an increase in the failure tolerance of ST circuits compared to synchronous counterparts, even despite their hardware redundancy.

3. INCREASING FAILURE TOLERANCE OF ST CIRCUITS

The most dangerous type of failure in ST circuits during dual-rail coding of information signals with a spacer is the appearance of a dual-rail signal state opposite to the spacer. Conventional dual-rail signal indication assumes that each dual-rail signal has only one spacer state (“00” or “11”) and treats any nonspacer state as a working one. Therefore, the state opposite to the spacer (antispacer) will be indicated as working, which will lead to error propagation through the circuit.

In [13], circuitry and layout methods were proposed to prevent the propagation of an antispacer through the ST circuit. An antispacer is indicated as a spacer with the “XOR” or “XNOR” cell. This solution masks the antispacer and increases the resistance of combinational ST circuits to single failures up to the level of 95% ($\alpha_{\text{ST}} = 0.05$).

Practical ST circuits are designed as a pipeline, similar to synchronous circuits. Traditionally, the register for storing intermediate results of the pipeline stage is implemented on hysteresis latches [2]

(Muller C-elements [5]), since they store both the working and spacer states. The antispacer at the input of such a bit of the storage register is written to it in the working phase but is indicated as a spacer. This is not critical from the point of view of the discipline of the ST circuit—the indicator of the register remains in the spacer, and the register is waiting for the antispacer at the input of the faulty bit to change to the working state.

However, the antispacer inside the register bit blocks writing working state to it at the end of the failure, causing the ST circuit to stop. The circuitry methods proposed in [16, 17] ensure the removal of this blocking. They increase the failure tolerance of the ST register to the level of 95.3% ($\alpha_{\mathrm{ST}} = 0.047$).

As a result of the use of special circuitry and layout methods for improving the failure tolerance of ST circuits, the ratios of the time of failure-free operation of ST circuits and their synchronous counterparts in formulas (2)–(3) are significantly improved, viz., for combinational-type circuits we have

$$K_{\mathrm{FFOT\,C2}} = \frac{\alpha_{\mathrm{S}}}{A_{\mathrm{R}} \times \alpha_{\mathrm{ST}}} = \frac{0.5}{2.5 \times 0.05} = 4.0,$$

and for sequential ones,

$$K_{\mathrm{FFOT\,S2}} = \frac{\alpha_{\mathrm{S}}}{A_{\mathrm{R}} \alpha_{\mathrm{ST}}} = \frac{0.5}{1.5 \times 0.047} \approx 7.1.$$

The resistance of the indication subcircuit of ST circuits to failures is increased by using the DICE implementation of the C-element [18] with in-phase inputs and output, which masks failures at the inputs and inside the C-element.

Thus, ST circuits have a higher resistance to failures and longer failure-free operation compared to synchronous counterparts in both typical and failure-tolerant designs due to more reliable masking of single failures. As an example, let us evaluate the advantage of the real 54×54 multiplier in the ST version. Variants of the synchronous multiplier [19, Table V] have a complexity of 78 800, 81 600, 82 500, and 100 200 CMOS transistors. The ST multiplier with dual-rail signal encoding [20] has a complexity of 218 000 CMOS transistors. Therefore, the ratio of the time of failure-free operation of the ST multiplier and its counterpart is in the range from 3.6 to 4.6.

4. CONCLUSIONS

Approximate estimates show that the ST methodology for designing hardware for computing and control systems provides better failure tolerance than the synchronous approach. Due to the bi-phase discipline of operation and redundant coding of information signals, ST circuits are immune to 90% of failures in the combinational part and 76% of failures in the sequential part. Because of this, the hardware of computing and control systems built on the basis of the ST methodology, despite its hardware redundancy, provides a longer failure-free operation time in comparison with synchronous counterparts—1.8 times in combinational circuits and 1.2 times in sequential ones. The proposed methods of circuitry and layout design of ST circuits increase the failure-free operation time of the hardware of computing and control systems by a factor of 2.2 for combinational-type circuits and up to 5.8 times for sequential-type ones.

The scientific novelty of this article is as follows:

– It performs a comparative analysis of the possibilities of using synchronous and ST methodologies to build the element base of highly reliable hardware of computing and control systems.

– It shows that the use of ST circuitry provides a higher level of resistance of the hardware of computing and control systems to single failures than synchronous counterparts at comparable hardware costs.

Further directions of work are related to the study of hardware solutions that use the noise-immune properties of ST codes and track the moment of failure; this can give a significant gain in the stability of the hardware of computing and control systems to single failures.

FUNDING

The study was financially supported by the Ministry of Science and Higher Education of the Russian Federation, project no. 075-15-2020-799.

REFERENCES

1. Viktorova, V.C., Lubkov, N.V., and Stepanyants, A.S., *Analiz nadezhnosti otkazoustoichivykh upravlyayushchikh vychislitel'nykh sistem* (Reliability Analysis of Failure-Tolerant Control Computing Systems), Moscow: Inst. Probl. Upr. Ross. Akad. Nauk, 2016. https://www.ipu.ru/sites/default/files/card_file/VLS.pdf.
2. Kishinevsky, M., Kondratyev, A., Taubin, A., and Varshavsky, V., *Concurrent Hardware: the Theory and Practice of Self-Timed Design*, New York: John Wiley & Sons, 1994.
3. Zakharov, V., Stepchenkov, Y., Diachenko, Y., and Rogdestvenski, Y., Self-timed circuitry retrospective, *Int. Conf. Eng. Technol. Comput. Sci. (EnT)* (Moscow, 2020), pp. 58–64. https://doi.org/10.1109/EnT48576.2020.00018
4. Tabassam, Z., Naqvi, S.R., Akram, T., Alhussein, M., Aurangzeb, K., and Haider, S.A., Towards designing asynchronous microprocessors: from specification to tape-out, *IEEE Access*, 2019, vol. 7, no. 5, pp. 33978–34003. https://doi.org/10.1109/ACCESS.2019.2903126
5. Smith, S.C. and Jia, Di., Designing asynchronous circuits using NULL Convention Logic (NCL), *Synthesis Lect. Digital Circuits Syst.*, 2009, vol. 4, no. 1, pp. 61–73. https://doi.org/10.2200/S00202ED1V01Y200907DCS023
6. Stepchenkov, Yu.A., Denisov, A.N., Diachenko, Yu.G., Grinfel'd, F.I., Filimonenko, O.P., Morozov, N.V., Stepchenkov, D.Yu., and Plekhanov, L.P., *Biblioteka funktsional'nykh yacheek dlya proektirovaniya samosinkhronnykh poluzakaznykh BMK mikroskhem serii 5503/5507* (Library of Functional Cells for Designing Self-Timed Semicustom Array Chip Microcircuits of the 5503/5507 Series), Moscow: Tekhnosfera, 2017. http://www.technosphera.ru/lib/book/497.
7. Lodhi, F.K., Hasan, S., Hasan, O., and Awwad, F., Low power soft error tolerant macro synchronous micro asynchronous (MSMA) pipeline, *IEEE Comput. Soc. Annu. Symp. VLSI* (Tampa, Florida,USA, 2014), pp. 601–606. https://doi.org/10.1109/ISVLSI.2014.59
8. Iturbe, X., Venu, B., Ozer, E., and Das, S., A Triple Core Lock-step (TCLS) ARM Cortex-R5 processor for safety-critical and ultra-reliable applications, *46th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Networks Workshop (DSN-W)* (2016), pp. 246–249. https://www.researchgate.net/publication/309566878_A_Triple_Core_Lock-Step_TCLS_ARMR_CortexRR5_Processor_for_Safety-Critical_and_Ultra-Reliable_Applications. https://doi.org/10.1109/DSN-W.2016.57
9. Gkiokas, C. and Schoeberl, M.A., Fault-tolerant time-predictable processor, *IEEE Nordic Circuits Syst. Conf. (NORCAS): NORCHIP Int. Sympos. System-on-Chip (SoC)* (2019), pp. 1–6. https://www.researchgate.net/publication/337527343_A_Fault-Tolerant_Time-Predictable_Processor. https://doi.org/10.1109/NORCHIP.2019.8906947

10. Mavis, D. and Eaton, P., SEU and SET modeling and mitigation in deep submicron technologies, *IEEE Int. Reliability Phys. Sympos.* (Phoenix, Arizona, USA, April 15–19, 2007), pp. 293–305. https://doi.org/10.1109/RELPHY.2007.369907

11. Rogdestvenski, Yu.V., Stepchenkov, Yu.A., Diachenko, Yu.G., Morozov, N.V., Stepchenkov, D.Yu., and Diachenko, D.Yu., Method for increasing the speed of a self-timed multiplier, in *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem. Vyp. 1* (Problems of Developing Promising Micro- and Nanoelectronic Systems. Issue 1), Moscow: Inst. Probl. Proekt. Mikroelektron., 2020, pp. 82–88. https://doi.org/10.31114/2078-7707-2020-1-82-88

12. Sokolov, I.A., Stepchenkov, Yu.A., Diachenko, Yu.G., and Rogdestvenski, Yu.V., Improving the failure tolerance of self-timed circuits, *Inf. Primen.*, 2020, vol. 14, no. 4, pp. 63–68.

13. Stepchenkov, Y.A., Kamenskih, A.N., Diachenko, Y.G., Rogdestvenski, Y.V., and Diachenko, D.Y., Improvement of the natural self-timed circuit tolerance to short-term soft errors, *Adv. Sci. Technol. Eng. Syst. J.*, 2020, vol. 5, no. 2, pp. 44–56. https://doi.org/10.25046/aj050206.

14. Emeliyanov, V.V., Vatuev, A.S., and Useinov, R.G., Impact of heavy ion energy on charge yield in silicon dioxide, *IEEE Trans. Nucl. Sci.*, 2018, vol. 65, no. 8, pp. 1596–1502. https://doi.org/10.1109/TNS.2018.2813669

15. Dubrova, E., *Fault-Tolerant Design,* Berlin: Springer, 2013. https://doi.org/10.1007/978-1-4614-2113-9

16. Sokolov, I.A., Zakharov, V.N., Stepchenkov, Yu.A., and Diachenko, Yu.G., RF Patent no. 2725778, Self-timed failsafe storage register bit device, *Byull. Izobret.,* 2020, no. 19.

17. Sokolov, I.A., Zakharov, V.N., Stepchenkov, Yu.A., and Diachenko, Yu.G., RF Patent no. 2733263, Self-timed failsafe storage register bit device, *Byull. Izobret.,* 2020, no. 28.

18. Sokolov, I., Stepchenkov, Y., Diachenko, Y., Rogdestvenski, Y., and Diachenko, D., Increasing self-timed circuit soft error tolerance, *IEEE EastWest Design & Test Sympos. (EWDTS)* (Varna, Bulgaria, 2020), pp. 450–454. https://doi.org/10.1109/EWDTS50664.2020.9224705

19. Makino, H., Nakase, Y., Suzuki, H., Morinaka, H., Shinohara, H., and Mashiko, K., An 8.8 ns 54x54 bit multiplier with high speed redundant binary architecture, *IEEE J. Solid-State Circuits*, 1996, vol. 31, no. 6, pp. 773–783. https://doi.org/10.1109/4.509863

20. Stepchenkov, Y.A., Diachenko, Y.G., Rogdestvenski, Y.V., Diachenko, D.Yu., and Shikunov, Yu.I., Self-timed multiply-add-subtract unit alternates, *IEEE Conf. Russ. Young Researchers Electr. Electron. Eng. (EIConRus)* (2020), pp. 1864–1868. https://doi.org/10.1109/EIConRus49466.2020.9039039

This paper was recommended for publication by A.M. Krasnosel'skii, a member of the Editorial Board