FPGA-Based Regenerative Electronic Systems in the Spacecrafts

Savkin Leonid V.
Public Stock Company «Radiofizika», Scientific Research Department No.1, Moscow, Russian Federation
E-mail: leonid.itmo@mail.ru

***Abstract*—In article results of research of ways hardware-software creation of the regenerative electronic systems based on application FPGA technologies with dynamic reconfiguration are presented. These systems are offered to be used in the modern spacecraft's onboard fault-tolerant equipment. Several possible options of realization of the regenerative electronic systems capable to partially imitate processes of biological regeneration due to low-level redundancy of the configurable logic blocks are presented. The main technical and methodological difficulties arising at creation of spacecraft's onboard regenerative electronic systems are shown.**

***Index Terms*—regenerative, fault-tolerant, reconfiguration, logic block, multi-level, low-level reservation, redundance**

I. Introduction

The integration scale of electronic component basis for space application permanently grows. Thanks to actively developing FPGA (Field-Programmable Gate Array) technologies there are new ideas in creation spacecraft's fault-tolerant onboard systems of today. Many of these ideas cardinally differ from the traditional microprocessor architecture using reservation of subsystems, modules and units at high hardware level. Programmable logic integrated circuits constructed on FPGA technologies open new opportunities in hardware-software creation of the fault-tolerant onboard measuring computing and controlling complexes for spacecraft's. The speech first of all goes about creation such systems which will not only possess redundance at low hardware level, but also will be capable to imitate somewhat processes of regeneration in case of origin of different breakages and failure. It in turn allows speaking about a new class of onboard fault-tolerant systems of spacecraft's– Regenerative Electronic Systems (RES).

The operation purpose–research the methods of hardware-software creation RES on the basis of FPGA technologies. In this article the main concepts of RES will be considered. Key problematic issues on implementation of the multireserved architecture at low hardware level with use of FPGA technologies will be shown.

Manuscript received July 1, 2016; revised December 5, 2016.

II. Requirements FDIR

Creation of all spacecraft onboard systems is always followed by execution of the International requirements of FDIR (Fault Detection, Isolation and Recovery). The essence of these requirements consists in sequential implementation of the strict principles of functioning any onboard management system in case of origin in it different breakages and failures [1]-[5]. We will list the basic principles of functioning spacecraft's onboard systems meeting requirements FDIR.

- The organization of the functional and test monitoring with the given depth of failure localization.
- Creation schemes of the adaptive majority reservation with number of the reserved channels at least three.
- Application of "hot" and "cold" reservation of the main spacecraft's onboard subsystems.
- Registration of breakages and failures in the hardware and program component spacecraft's onboard systems.
- Support of lock data from faulty subsystems.
- Restoration of faulty spacecraft's onboard subsystems of due to high-level reconfiguration of the functional and reserved spacecraft's onboard systems.
- Continuous self-test of channels of processing of the functional and diagnostic information.
- Output all data in real time on channels of a telemetry on earth control complexes about each of the done procedures.

All above described procedures are realized in the majority of the modern spacecraft's.

Earlier it was already said that in most cases reconfiguration of onboard equipment is carried out at the high hardware level of systems. The FPGA technology allows us to realize reconfiguration of the hardware subsystems at low hardware level. This circumstance allows us to realize in certain cases restoration of faulty onboard systems by the principles imitating remotely biological regeneration.

III. Imitation of Biological Regeneration by Means of FPGA Technology

A. Two Important Assumptions

First, it is clear that we can't imitate completely processes of biological regeneration in literal sense as it is

doi: 10.18178/ijmerr.6.1.77-81

about solid-state electronics. However, doing a row of the assumptions concerning reservation of hardware at the level of a semiconductor crystal we will understand a possibility of low-level reservation of the functional elements as regeneration. As the functional elements in FPGA systems sets of the Configurable Logic Blocks (CLB) which form main computing cells are used.

Secondly, when we speak about low-level reservation of the hardware architecture it is necessary to consider the significant scale differences of this approach from the approaches which are based on methods of high-level reservation in onboard subsystems. It is important to consider not only the quantitative assessment of total number of the reserved computing cells, but also methods of their monitoring. These methods in some cases strongly differ from those which are applied in case of reservation of the spacecraft's onboard subsystems at high hardware level. The type of the selected programmable logic integrated circuit forming a uniform reconfigurable computing field (RCF) is important.

B. Potential Hardware Environment of the RES

Possibilities of some types FPGA systems on dynamic reconfiguration allow creating the spacecraft's onboard integrated RES which consist of two parts interacting among themselves: real hardware system and potential hardware environment (Fig. 1).

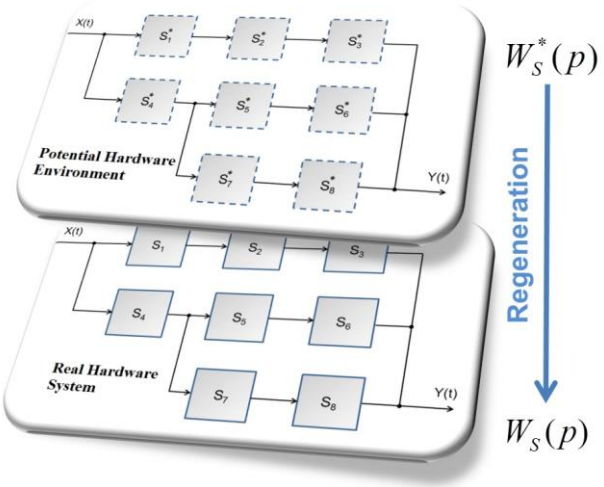

Figure 1. To the explanation of restoration processes in the RES.

From this figure it is visible that in addition to the main (real) hardware system realizing digital information processing the so-called potential hardware environment is a RES part. The real hardware system is described by gear function $W_S(p)$ which value is created on the basis of gear functions of a set of subsystems $\{S_1, S_2, \ldots, S_8\}$. The potential hardware environment is the low-level excess RCF area. Its computing resource is calculated so that thanks to dynamic reconfiguration of RCF it was possible to recover value of the gear $W_S(p)$ function due to restoration of gear function of any of elements $S_i, i = \overline{1,8}$.

One of key features of creation of onboard RES is that all of them represent uniform computing systems. Proceeding from this aspect an important stage in design of RES is the choice of the functional complexity of a computing cell of uniform RCF. Each computing cell is created of a set of CLB creating the first (i.e. the lowest) hardware level of uniform architecture of RCF. We will consider one of possible options of calculation of the general the functional complexity of a computing cell of the RES.

C. The Functional Complexity of a Computing Cell

Use of uniform architecture in fault-tolerant hardware architecture is very convenient from the point of view of application of the same monitoring and diagnostics algorithms to fragments of RCF of the RES different on the combinative complexity [6]-[8]. Optimum functional complexity $H_{\text{opt.}}$ of a computing cell in RCF of the RES shall be defined proceeding from total quantity of the computing cells which are a RCF part and also proceeding from switching flexibility of RCF (Fig. 2).

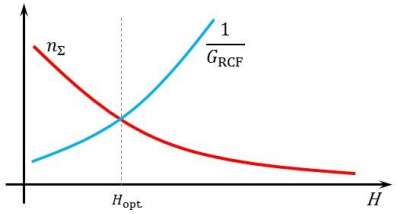

Figure 2. Optimum functional complexity of a computing cell.

In the provided figure optimum functional complexity $H_{\text{opt.}}$ is defined in cross point of two functions depending on the general complexity of a computing cell H. The first function $n_\Sigma = F(H)$ – total quantity of the computing cells created on the basis of CLB. The second function represents function reverse to the conditional value of flexibility of uniform RCF G_{RCF}. For example, for idealized full-meshed RCF with bidirectional communications between CLB

$$G_{\text{RCF}} \sim n_\Sigma^2$$

The hardware implementation of any algorithm of information processing represents "superimposing" of the digraph $G(X, K)$ on RCF. As a result of superimposing of the digraph the selected fragment of RCF of the RES is formed. Here X – a set of the arcs of the digraph describing directional logic-arithmetical communications between CLB peaks; K – a set of the CLB-peaks realizing this or that elementary logic-arithmetical operation in uniform RCF.

IV. General Requirements to Res

We will list the main requirements which shall be shown to RES when carrying out low-level reservation of the hardware architecture consisting of sets of computing cells with dynamically tunable logic-arithmetical communications:

a) Openness of the hardware architecture of low-level of the RCF;

b) Possibility of self-testing elements of low hardware level;

c) The uniform unified approach to the hardware creation of the RCF realizing the functional (real) and potential environment.

The first requirement is caused by need of support of RES with a possibility of dynamic reconfiguration at the level of basic elements.

The second requirement is directly connected with determination of RES for which before recovering a faulty element of low hardware level it is necessary to establish the fact of failure and to localize its place.

The third requirement defines the principle of restoration of faulty elements. This principle is directly connected to architectural features of creation of the functional and potential excess environment. This requirement shall consider compliance of growth of complexity of the hardware creation system with increase in volume of its functional tasks and complication of algorithms information processing. All above mentioned requirements to features of creation of RES imply the uniform and unified approach in case of creation of fault-tolerant onboard equipment on the basis of FPGA systems.

V. The Main Approaches on Creation RES

On the basis of the above requirements to hardware-software creation of RES we will consider several possible options of their implementation.

A. The Functional and Controlling Environments

The idea of this approach consists in division of uniform RCF into two fragments: the functional environment and the controlling environment. The functional environment represents set of the hardware architecture of RCF realizing digital information processing in onboard the RES spacecraft. The controlling environment also represents the computing area distributed in RCF which tasks is monitoring and "substitution" of faulty CLB on operational. Thus the controlling environment exercises control of redundance of RCF of the RES.

Interest in this approach is connected also to a possibility of the organization of the functional and controlling environments by the principles of cellular automatons. The principles of reservation in similar systems are most effectively capable to imitate the regeneration processes watched in biological systems.

B. Low-Level Majority Reservation

Use of majority diagrams of reservation of CLB assumes that all majority organs will also be realized on the basis of the selected fragments of RCF of the RES. From the point of view of practical implementation this option is the simplest as differs in nothing from widespread methods of majority reservation of the whole subsystems of spacecraft's.

Besides, for the purpose of reliability augmentation of functioning of majority organs, use of the multi-layer circuits of majority reservation which are also realized on the basis of the selected fragments of RCF of the RES is quite possible.

C. Architectures with Multi-Level Reconfiguration

The fact that all hardware hierarchy of RCF is divided into $q=\overline{1,N}$ the conditional hardware levels is characteristic of this approach. To the first conditional hardware level (q = 1) there corresponds the CLB level. Further all computing field uniformly "breaks" into matrix fragments so that each of these fragments was a component of a matrix of RCF. Each of these fragments will represent the functional elements of RCF of the second conditional hardware q = 2 level. Continuing iterations to some $q = v$ value, the hardware structure of uniform of RCF will represent a block matrix over which elements also as well as over CLB it will be possible to make reconfiguration procedure. Functioning of similar system is illustrated in a Fig. 3.

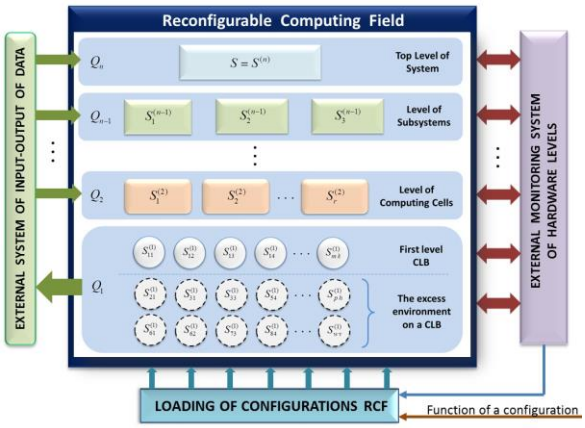

Figure 3. The functioning of RES with multi-level reconfiguration.

The architecture of each conditional hardware RES level represents set of a look

$$Q_j = S_1^{(j)} \bigcup S_2^{(j)} \bigcup \ldots \bigcup S_k^{(j)},\ j = q = \overline{1,N},$$

where j–sequence number of the conditional hardware of RCF level; $S_k^{(j)}$ –the functional element at number k which forms architecture Q_j.

Configuration function of the functional element $S_k^{(j)}$ it is possible to present dependences in the form

$$K_r^{(j)} = f(S_r^{(j)}) = f(\mathbf{L}_r^{(j)}, G(\mathbf{L}_r^{(j)})), r = \overline{1,k}, \quad (1)$$

where $\mathbf{L}_r^{(j)}$–matrix set of CLB creating the selected RCF fragment with functions $S_k^{(j)}$; $G(\mathbf{L}_r^{(j)})$ –digraph describing topology of logic arithmetical communications within a matrix set of CLB $\mathbf{L}_r^{(j)}$.

By this principle creation enough difficult functional architectures of RCF of the RES which will represent generally fractal architecture is possible. Use of methods of the built-in multi-level monitoring of architecture of RCF will be characteristic of this method that is interesting from the point of view of the uniform unified principle of creation fault-tolerant onboard systems. The multi-level hierarchy of this RES in a type of a possibility of reconfiguration each of its levels also has multi-level character [9].

In Fig. 4 the principle of restoration the functional element of the second functional hardware level is shown.

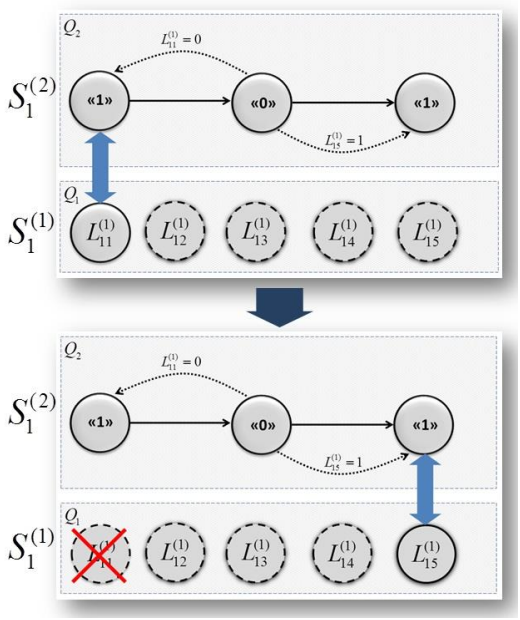

Figure 4. Principle of restoration of the functional element.

In order that the functional element $S_1^{(2)}$ again became operable the RCF makes reconfiguration in case of which substitution CLB $L_{11}^{(1)}$ with CLB $L_{15}^{(1)}$ is carried out.

Thus procedures of reconfiguration are carried out for the functional elements at all conditional hardware levels.

D. The Low-level "Sliding" Reservation

This approach is completely similar to the approaches which are applied today in high level hardware subsystems of spacecraft's. It is quite convenient in case of implementation the simple parallel and pipeline architecture.

In case of the considerable complication of fragments of RCF architecture this option becomes unsuitable because of complexity its implementation in RCF.

However in certain cases this approach manages to be realized quite effectively when using the cyclic architecture reminding a computer game "Snake".

E. Architectures with the Parallel Differential Testing

Implementation of RES with use of this approach consists in simultaneous creation in RCF of conjugate and parallel architecture. One of these architecture realizes the required function, and the second – Boolean derivative of this function. The advantage of this option is the possibility of single-digit identification of the failure arising in separate CLB. This method is based on the hardware implementation of architecture which is capable to calculate value of a Boolean derivative in control points of the functional architecture.

Application of such computing cells in addition to execution of the functional tasks, allows to realize algorithms of self-checking and monitoring of a status of adjacent computing cells. This approach is interesting from the point of view of hardware-software creation of RCF with properties of some self-sufficiency of the selected functional fragments.

In all cases the hardware architecture of RES can be described by means of configuration function (1).

VI. Conclusion

Thus, the choice of a method of hardware-software creation of RES, first of all, is defined proceeding from technical characteristics of FPGA system.

In the inference it is necessary to mark two important problems relating to effective creation of onboard RES in the modern spacecraft's.

The first problem is connected to opportunities of FPGA technologies for implementation of low-level dynamic reconfiguration of the hardware architecture insufficient today in real time. For this reason in most cases RCF is realized on the basis of several programmable logic integrated circuits. Try to obtain the mode of dynamic reconfiguration by means of duplicating of configuration sets of the hardware architecture of RCF.

The second problem is connected to distribution of memory of RCF status among all computing cells participating in the functional information processing. Some solutions of this problem inevitably lead to creation of architecture of RES in the form of artificial neural networks, in particular, to different models of perceptron's.

Each of the considered approaches in creation of spacecraft's onboard FPGA-based RES represents both theoretical and practical interests.

References

[1] A. Zolghadri, "Advanced model-based FDIR techniques for aerospace systems: Today challenges and opportunities," *Progress in Aerospace Sciences*, 2012.

[2] R. Gessner, B. Kosters, A. Hefler, R. Eilenberger, J. Hartmann, and M. Schmidt, "Hierarchical fdir concepts in s/c systems," in *Proc. 8th International Conference on Space Operations (SpaceOps)*, Montreal, Canada, May. 2004.

[3] R. Patton, Ed, *Issues of Fault Diagnosis for Dynamic Systems*, London [u.a.]: Springer, 2000.

[4] Fault-Detection, Fault-Isolation and Recovery Techniques. Utilize FDIR Design Techniques to provide for Safe and Maintainable On-Orbit Systems/NASA. [Online]. pp. 1-6. Technique DFE-7. Available: http://paso.esa.int/5_training_materials/training_11_fault%20techniques.pdf

[5] A. Wander and R. Forstner, "Innovative fault detection, isolation and recovery strategies on-board spacecraft: State of the art and research challenges," Deutscher Luft-und Raumfahrtkongress, 2012.

[6] S. Hauck, *Reconfi Gurable Computing. The Theory and Practice of FPGA-Based Computation*, Morgan Kaufmann Publ., 2007. p. 944.

[7] M. Gokhale and P. S. Graham, *Reconfigurable Computing–Accelerating Computation with Field-Programmable Gate Arrays*, Springer Publ., 2005, p. 238.

[8] I. A. Kalyaev, I. I. Levin, A. I. Dordopulo, and L. M. Slasten, "FPGA –based reconfigurable computer systems," in *Proc. Science and Information Conference, SAI 2013*, London, October, Springer, 2013, pp. 148-155.

[9] L. V. Savkin, "Regenerative electronic systems in space systems and complexes," *Herald of Cybernetics*, vol. 2, no. 18, pp. 3-32, 2015.

Savkin Leonid V. Was born on February 8, 1986. In 2008 graduated from the Moscow Space Forces Military Institute of Radioelectronics, the expert in the field of the reconfigurable measuring and computing systems. Works as the expert of the 1st category in the Scientific Research Department No. 1 of PSC "Radiofizika", Moscow, Russian Federation
E-mail: leonid.itmo@mail.ru