Mathematical Problems of Computer Science 56, 7--17, 2021.

UDC **004.052**

Functional Safety Compliant Test & Repair Framework for System-on-Chip Lifecycle Management

Gurgen E. Harutyunyan, Samvel K. Shoukourian, Grigor A. Tshagharyan and Yervant A. Zorian

Synopsys
e-mail: gurgen.harutyunyan@synopsys.com, samvel.shoukourian@ synopsys.com,
grigor.tshagharyan@synopsys.com, yervant.zorian@synopsys.com

Abstract

The share of safety-critical systems in electronic and electrical (E/E) devices across multiple domains, especially in automotive industry, is growing at a constant rate. An unhandled failure of any component within the system may compromise the safety of the entire ecosystem. Therefore, regardless of the context of use and level of the hierarchy, all system components must follow the requirements of appropriate safety standard for the development process and in-field operation. In this context, the traditional built-in self-test (BIST) scheme must also meet the safety requirements defined in ISO 26262 in order to be approved for automotive applications. The paper presents a functional safety compliant BIST infrastructure concept that helps to ensure safe test execution throughout the entire System-on-Chip (SoC) lifecycle while maintaining high test quality.

Keywords: Functional safety, Automotive, ISO 26262, ASIL, Built-in self-test, in-field test, Test and repair.

1. Introduction

The automotive industry is currently one of the fastest growing sectors in the semiconductor industry. The growing demands of consumers for safety, reliability and increased safety requirements continue to drive the automotive market growth. The trend towards greater safety and better driving experience is forcing automakers to continually integrate a large number of electronic and electrical components into their vehicles, such as advanced driver assistance systems (ADAS) and in-vehicle infotainment (IVI). A few examples of such systems are adaptive cruise control, parking assist, emergency vehicle braking, lane change assist, etc. as the list continues to grow.

Functional safety sensitive systems are traditionally prone to using well-established technologies that provide high performance, reliability and lower production cost over the system lifecycle. However, given the growing demand, automakers have had to adopt solutions with increased processing power and communication performance, which require usage of relatively new technology nodes such as 3D transistors. Especially during the recent years, with increasing pressure and high competition in the consumer market, the time to market margin has drastically decreased. This trend creates additional challenges for automotive application developers and original equipment manufacturers (OEMs) to meet the demands for higher performance and energy efficiency along with natural requirements for safety, reliability and quality. As a result, the modern automotive industry faces a number of challenges that need to be addressed, including functional safety and reliability, quality of testing in the production process, as well as security and robustness. These various requirements can sometimes become conflicting and even mutually exclusive, making them difficult to fully meet.

One of the most important aspects to consider in the automotive industry is the requirement for high quality testing. Here, special attention should be paid to the development of a reliable solution for test and repair, applicable not only at the production stage, but throughout the entire life cycle of the product, from the design stage to silicon bring up and series production right up to in-field operation. This is especially related to embedded memories, covering most of the system-on-a-chip (SoC) area and making a major contribution to achieving high performance and defective parts per billion (DPPB) rate. BIST has traditionally been the preferred approach for testing and repairing embedded memories, providing a reasonable trade-off between cost and achievable yield. Until recently, the existing BIST solutions were entirely focused on the SoC production phase to provide high fault coverage and optimal yield. However, today the situation has changed dramatically, especially in the automotive industry, given the recent breakthrough in the market. Currently, safety and security in mission mode are considered the highest priority requirements for vehicles, therefore, comprehensive testing capabilities are required not only at the production stage, but also in the field.

Many techniques and new methods have already been proposed in the literature to address post-production testing problems that address specific aspects of testing. In [1] and [2], the concept of transparent memory BIST is described, where the key idea is to make the test execution transparent to the system, keeping the contents of the memory intact. Other versions of the transparent BIST architecture with improved coverage and test times were also introduced later in [3]-[5]. The implementation of memory and logic BIST for automotive SoCs described in [6] introduces several new features specifically designed for in-field testing. For example, the idea of non-destructive and destructive self-tests is presented. In the former case, test control units are not included in the test, allowing for more control in the field and planning of test procedures. Meanwhile, in the latter case, the entire device is tested, and after the test, a hard reset or system restart is required, as the system state is lost. In [7], a short burst-based BIST technique is introduced, where the idea is to divide the memory into smaller chunks and partition the BIST execution into a series of short bursts testing one chunk of the memory at a time.

There are other works as well on this topic showing the benefits of using structural tests (mostly built-in) for field testing and diagnosis purposes. For example, [8] shows the trade-offs and benefits that the automotive field can gain by reusing production testing methods for in-system testing. In addition to structural testing, other alternative methods recently proposed in the literature can also be adopted for in-field testing. This includes software-based self-test methods described in [9]-[10] and functional test methods, a good review of which can be found in [11].

Two other aspects related to in-field testing are the problem of integrated circuit aging in mission mode and the importance of lifetime testing. With regard to aging, several methods of online aging monitoring are proposed, such as the architecture presented in [12], which is specially designed for safety-critical applications. The proposed built-in sensors work only when the car is turned on, therefore they are resistant to aging and threshold voltage fluctuations. Aging monitoring is done by observing propagation delays in critical parts of the chain. In [13], the authors carried out a study on aging faults investigation in the field and proposed an algorithmic method for the detection of aging-induced faults in an earlier stage of SoC lifetime. Meanwhile, the importance of adopting a functional approach to characterize the reliability and operating lifetime of SoCs is discussed in [14]. A new approach to automatically generate appropriate test patterns for use in the mission mode is presented, with experimental results demonstrating the advantages of the proposed method.

The purpose of this paper is to combine the existing best practices and present the concept of a full-scale functional safety compliant solution for testing embedded memories in automotive SoCs with the help of the proposed universal BIST engine. In other words, the proposed BIST concept represents a one-stop test solution from production to in-field test. The next section of this paper provides an introduction to the ISO 26262 [14] standard and the certification process in automotive. Section 3 takes a closer look at the phases of automotive in-field testing and the various requirements they impose. The details of the proposed solution for automotive SoCs are described in Section 4. Finally, Section 5 draws the conclusions.

2. Functional Safety and ISO 26262

One of the key requirements to consider when developing any application for an automotive SoC is functional safety and reliability. Known requirements for applications critical in terms of functional safety, established in industries such as the military, nuclear and aerospace, are now being widely transferred to the automotive industry. The main goal of safety and reliability is to tolerate the risk of physical injury or of damage to the health of people. Safety primarily aims to reduce the risk of systematic as well as random hardware failures in the system during manufacturing or in-field operations. At the same time, reliability determines the probability that the system will perform the functions assigned to it in a given period of time.

The increased attention to safety and reliability aspects in the automotive industry has led to the need for common criteria to measure the level of their compliance in the system. This was the reason for the emergence of ISO 26262 standard called “Road vehicles: Functional Safety”, which establishes the definitions and requirements for functional safety for automotive equipment applicable throughout the life cycle of all safety-oriented automotive E/E systems. ISO 26262 specifies requirements for achieving an acceptable level of functional safety for electrical and/or electronic systems intended for use in vehicles. Depending on meeting these requirements, the final product can be qualified with one of the four available Automotive Safety Integrity Levels (ASIL) A to D.

ASIL refers to an abstract classification of the inherent safety risk in an automotive system or elements of such a system. The ASIL classification is used in ISO 26262 to express the level of risk reduction required to avoid a particular hazard, where ASIL-D is the highest level and ASIL-A is the lowest level. The ASIL evaluated for that hazard is then assigned to the safety goal established to eliminate that hazard, and then inherited by the safety requirements resulting from that goal. ASIL is determined based on a combination of the probability of exposure, the possible controllability by the driver, and the severity of the possible consequences in case critical event occurs.

The general ASIL certification process for random hardware faults in automotive products consists of the following main steps:

1. The product is handed over to the certification body.
2. Safety Goal Violations (SGVs) of the product are determined:
   - Safety Goal (SG) is a safety requirement imposed on a product in order to reduce the risk of one or more hazardous events to an acceptable level;
   - Safety Goal Violation (SGV) is a violation of a safety goal due to product malfunction.
3. The failure modes (FM) of the product are defined:
   - Faults that lead directly to the violation of a safety goal are called Single Point Faults (SPFs);
   - MPFs (Multiple Point Faults) are a combination of two or more independent faults leading directly to the violation of a safety goal.
4. Diagnostics Coverage (DC) for each FM and SGV is calculated:
   - At this stage, the certification body determines the impact of each FM on each SGV and calculates the corresponding DC of the product able to detect that impact.
5. ASIL-X level is defined based on the DC numbers:
   - Based on the obtained DC number, the ASIL level of the product is determined from Table 1.
6. FIT rate of the product is calculated:
   - Using known formulas, the certification body calculates the FIT rate of the product.
7. FMEDA report is prepared:
   - The FMEDA report contains all the information obtained during the certification process including SGVs, FMs, DCs, ASIL and FIT numbers.
8. The certification body provides the ASIL-X level compliance certificate.

Table 1. ASIL and FIT requirements

ASIL	**SPF**	**MPF**	**FIT Rate**
ASIL B	≥ 90 %	≥ 60 %	100 (recommended)
ASIL C	≥ 97 %	≥ 80 %	100 (required)
ASIL D	≥ 99 %	≥ 90 %	10 (required)

Probabilistic Metric for random Hardware Failures (PMHF) is yet another quantitative analysis method used in the scope of ISO 26262, which defines the average probability of failure per hour over the operational lifetime of the item. The formula for PMHF estimation looks like the following:

$$PMHF = \lambda_{SPF} + \lambda_{RF} + \lambda_{DPF\_detected} \times \lambda_{DPF\_latent} \times T_{Lifetime}$$

where

λ_{SPF}	is the single point failure rate;
λ_{RF}	is the residual failure rate;
$\lambda_{DPF\_detected}$	is the detected and notified dual point failure rate;
λ_{DPF_latent}	is the latent dual point failure rate (mitigated but not notified);
$TL_{ifetime}$	is the vehicle lifetime.

2. Silicon Lifecycle Test Requirements

ISO 26262 defines clear requirements for all products before they can be used in vehicles, and the components responsible for built-in self-test and repair are no exception. In the past, the primary requirement was the test quality and the timely detection of manufacturing defects. In the automotive industry, the picture is different since the safety enters the scene. In order to better understand automotive requirements, we need to take a closer look at the various stages of the SoC life cycle and the key test requirements that need to be considered in each stage. In fact, the SoC life cycle can be divided into three main modes: production, power-on and mission modes.

A. Production Mode

In the automotive industry, as in any other high-tech industry, achieving efficient yield is a vital requirement. Therefore, it is crucial to have a comprehensive test and repair mechanism with all the necessary capabilities. To do this, at the design stage, test structures are most often built into the chip for test, repair and diagnosis purposes. The built-in structures provide the chip ability to self-test itself, reducing the complexity of test setup and the cost of using sophisticated test equipment as well as shortening time to market.

The first step to building a comprehensive test BIST infrastructure is to understand the full range of realistic fault models for the various memory architectures and technologies that will be used on the chip. Memory faults are an abstract representation of physical defects that can occur during the chip production phase. However, not only the faults occurring in the memory array need to be taken into account, but also the faults occurring in the surrounding blocks of the memory array, including the address decoder, the write driver, the sense amplifier, etc., must also be taken into account. Once a set of target faults has been determined, test algorithms need to be developed to ensure complete fault coverage and optimal yield.

B. Power-On Mode

Maintaining test structures active in the SoC even after the production phase is a particular requirement for applications focused on functional safety. Over time, the circuits wear out and negative effects can begin to appear within a few years after production, or even a couple of months in the worst case. The aging effect is the most common problem and is usually defined as the degradation of circuit performance over time. The effects of aging can include increased power consumption, reduced speed, timing delays, which can eventually lead to failures and malfunction of the system. With the rapid reduction of technology nodes, the effects of aging have increased significantly and can no longer be ignored, especially in applications such as automotive. The main causes of aging are the effects of negative bias temperature instability (NBTI) and hot carrier injection (HCI), but recently the positive bias temperature instability (PBTI) effect has also increased in importance.

Unlike the production phase, in-field test requirements are more stringent due to space, power and time constraints. Therefore, a test solution from a production mode cannot fully meet these criteria, so alternative solutions are usually offered. It is not necessary to consider all manufacturing defects during power-on testing, since as the study [13] shows only a subset of these defects may appear due to aging or electromigration. The power-on test starts whenever the engine is turned on and the main purpose is to quickly check if all devices are working properly before the system goes into mission mode, or otherwise report if any problem is detected. For the power-on test, it appears that the only known limitation is the test time, which must remain in the certain range. Therefore, unlike production testing, this mode uses lighter complexity test algorithms that target only the most common types of permanent faults. Sometimes, depending on the application, the power off or key off mode is also used for this purpose to perform additional checks before shutting down the engine.

C. Mission Mode

After successful completion of the power-on phase, the system enters the mission mode. Mission mode is the most critical mode, as the reaction time is extremely limited if something goes wrong, otherwise it can lead to fatal consequences. In mission mode, failures usually occur either due to soft errors, or permanent faults resulting from aging. Therefore, in this mode two categories of test mechanisms can be distinguished, depending on whether they are always active or are activated periodically.

The first category includes mechanisms such as Error Correction Codes (ECC), which constantly check whether data or address integrity is maintained during in-field operation. The main requirement in this case is to minimize the time from the detection of a problem to warning message delivery to the system. The primary goal of such test mechanisms is the detection of soft errors, which are usually caused by alpha particles and cosmic rays hitting the integrated circuits [16]-[17]. Soft errors, compared to hard errors, are temporary and do not cause permanent damage. However, with the growing safety and reliability concerns in the automotive industry, detecting and correcting these types of errors in mission mode is critical.

The second category includes test mechanisms that are periodically activated to test system components that are not always available. For example, memory and logic BIST cannot run continuously, as memory and other logic blocks may be occupied by the system. Only when the memory or the other blocks are freed by the system they can be checked for faults. The tests in this category are called mission periodic tests. The periodic check runs once during the safety interval, which is defined in the system specification as the period of time during which a failure can occur. There are a number of challenges that need to be addressed by a periodic memory test:

1. The specific memory instances can be tested only when they are in an idle state
2. The allotted time for a memory test is usually not enough to test the entire memory macro
3. The content of the tested memories must be preserved intact by the test
4. When an interrupt command is received from the system, the memory test should be stopped and restarted only after the memory is freed.

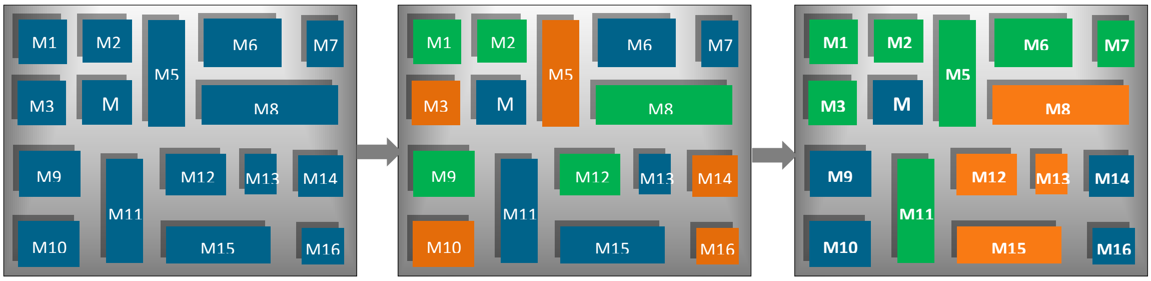

Fig. 1. Periodic test in the field: blue-colored blocks are in idle state, green-colored blocks are in mission and orange-colored blocks are in test mode.

Therefore, advanced methods are applied to divide the test into parts, and each time execute one part of the entire test. This means only a predefined subset of memory cells is tested during each test session. It is important to note that the state after each test session must be saved in the system in order to continue the test from exactly the same place when starting the next session. Fig. 1 shows an example of periodic in-field test at different points in time. In the first scenario, all memory blocks are idle, neither is being used by the system, nor being tested. In the second scenario, the memory blocks M1, M2, M8, M9, and M12 are in mission mode while M3, M5, M10, M14 and M16 are in test mode. Finally, in the third scenario, memory blocks M8, M12, M13 and M15 have been released by the system and can be put into test mode while M3 and M5 have been fully tested and are called into mission by the system.

2. BIST Architecture for Automotive SoCs

Fig. 2 shows a proposed BIST memory architecture for automotive SoCs. It consists of the following blocks:

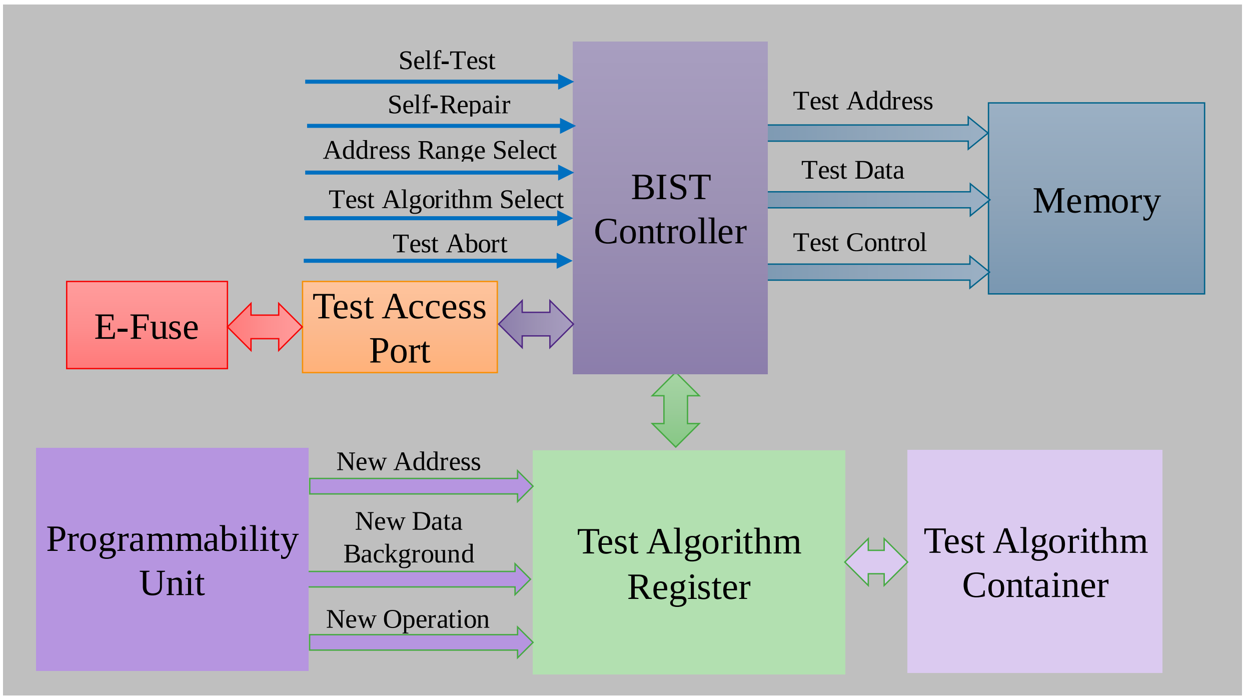

Fig. 2. The Proposed Automotive BIST Architecture.

- Test Algorithm Register - the proposed BIST architecture contains a set of predefined test algorithms designed to test and provide high fault coverage for different memory technologies. Specifically, fault coverage includes:
  - static single-cell and coupling faults
  - dynamic single-cell and coupling faults
  - links between static and dynamic faults
  - process variation faults
  - other technology-specific faults

  In addition to the comprehensive set of predefined test algorithms, it is possible to program new test algorithms or modify the existing ones if necessary. For this purpose, a programming interface is provided not only for adding new test algorithms, but also for new test operations, addressing types and background data patterns. In the case of automotive, in production mode, runtime and performance requirements are usually sacrificed in favor of higher fault coverage, which means that complex but robust test algorithms are generally recommended.

- Test Algorithm Container - allows to store test algorithms at the design stage, and then select the appropriate one for launching in power-on and mission periodic modes. Thus, by default, along with the predefined complex test algorithms for production mode, two or more shorter test algorithms are included in the test suite with reduced length, designed for power-on and mission modes. Unlike production mode, access to the built-in test in these modes can be granted through a dedicated test interface since after production access via JTAG test access port (TAP) is usually prohibited for security reasons. In addition to the self-test capability, a quick self-repair mechanism is also provided through the same interface to quickly start the repair procedure after the test is completed.
- BIST Controller - applies selected test algorithms to memory, and also provides a number of functions necessary for testing automotive SoCs:
    - The “Self-Test” function allows to check if the BIST circuit itself is working correctly. This especially concerns the logic units, which are responsible for collecting and sending the SoC state information to the outside world. The reason for such a test is that in case anything goes wrong, incorrect system status information may be captured, compromising the overall safety of the vehicle and the driver. A flexible error injection mechanism is provided for this purpose, which allows to inject errors into the memory array and surrounding blocks.
    - The proposed solution also helps to address the challenges related to periodic test. Due to critical time constraints, complex test algorithms cannot be used for in-field test, however, periodically running simpler test algorithm compensates and minimizes the risk of possible fault escapes to some extent. Depending on the specified interval length, either the test can be performed over the entire memory space or over a range of memory addresses using the “Address Range Select” function.
    - Another more advanced feature is the capability to make the test algorithm execution transparent. This is especially important to store the content of memories containing sensitive data during the periodic test. The scheduling of a periodic test execution is determined by the system's main processor, as it depends on many factors, including the length of the available test interval, memory availability, power restrictions, etc. For that purpose, the ability to run a periodic test on selected memory blocks is also provided.
- Finally, the “Test Abort” function is provided, which allows to stop the test execution and return the memory to mission mode in case of a request from the system.

The proposed BIST architecture makes it possible not only to test, but also to repair faulty memories. During the test phase a set of detected faults with information about their location is stored in a dedicated area in the top control unit. For this purpose, an array of one-time or reprogrammable electrical fuses is usually used to store this data. After the BIST run is complete, a self-repair procedure is started based on the stored information. At the beginning of this process, a redundancy analysis is performed to read the configuration of redundant elements (rows and columns) and fault locations, and then to determine the best redundancy allocation scheme for repair. If such a scheme is found, then the repair procedure is launched and all the faulty rows and columns in the memory array are replaced with redundant counterparts. Only after the successful repair, memory is declared ready for operation.

Optionally, a powerful diagnostic capability is also part of the proposed solution in case debugging and understanding of the root cause of detected faults is required. The developed test algorithms take as input a list of detected faults and determine the type of faults and attempt to identify common physical defect patterns, whether it is a single cell defect, a two-cell defect, a quadruple defect, a row/column defect, and so on. After the diagnosis is completed, a detailed report is generated with information about the types of faults found and the observed patterns of physical defects.

5. Conclusions

Functional safety has become increasingly important over the past decade, and especially with the introduction of the ISO 26262 standard, new requirements are placed on automotive SoCs and, ultimately, on all the components that they consist of. This paper presents the concept of an embedded memory self-test and repair solution that meets all functional safety requirements throughout the SoC's life cycle, including production, power-on, and mission modes.

References

[1] M. Nicolaidis, “Transparent bist for rams”, *International Test Conference* (ITC), pp. 598-607, 1992.

[2] M. Nicolaidis, “Theory of Transparent BIST for RAMs”, *IEEE Transactions On Computers*, vol. 45, no. 10, pp. 1141-1156, October 1996.

[3] M.G. Karpovsky, V.N. Yarmolik, “Transparent memory BIST”, *IEEE International Workshop on “Memory Technology, Design and Testing*”, pp. 106-111, 1994.

[4] S. Boutobza, M. Nicolaidis, K.M. Lamara, A. Costa, “A transparent based programmable Memory BIST”, *IEEE European Test Symposium* (ETS), pp. 89-96, 2006.

[5] I. Voyiatzis, C. Efstathiou, C. Sgouropoulou, “Symmetric transparent online BIST for arrays of word-organized RAMs”, *International Conference on Design & Technology of Integrated Systems in Nanoscale Era* (DTIS), pp. 122-127, 2013.

[6] A. Dutta, S. Alampally, A. Kumar andR. A. Parekhji, “A bist implementation framework for supporting field testability and confligurability in an automotive SOC”, Workshop on Dependable and Secure Nanocomputing, 2007.

[7] A. Becker, “Short burst software transparent on-line MBIST,” *IEEE VLSI Test Symposium* (VTS), pp. 1-6, 2016.

[8] A. Cook, D. Ull, M. Elm, H. Wunderlich, H. Randoll and S. Dohren, “Reuse of structural volume test methods for in-system testing of automotive ASICs”, *IEEE Asian Test Symposium* (ATS), pp. 214-219, 2012.

[9] F. Reimann, M. Glass, A. Cook, L. Rodriguez Gomez, J. Teich, D. Ull, H. J. Wunderlich, U. Abelein, and P. Engelke, “Advanced diagnosis: SBST and BIST integration in automotive E/E architectures,”, ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 1–6, 2014.

[10] P. Bernardi, R. Cantoro, S. D. Luca, E. Sánchez, A. Sansonetti, “Development flow for on-line core self-test of automotive microcontrollers”, *IEEE Transactions on Computers*, vol. 65, no. 3, pp. 744 – 754, March 2016.

[11] A. Jutman, M. S. Reorda and H.-J. Wunderlich, “High quality system level test and diagnosis”, *IEEE Asian Test Symposium* (ATS), pp. 298-305, 2014.

[12] J. C. Vázquez et al., “Built-in aging monitoring for safety-critical applications”, *IEEE European Test Symposium* (ETS), pp. 9-14, 2009.

[13] G. Tshagharyan, G. Harutyunyan, Y. Zorian, A. Gebregiorgis, M. S. Golanbari, R. Bishnoi and M. B. Tahoori, “Modeling and testing of aging faults in FinFET memories for automotive applications”, *IEEE International Test Conference* (ITC), pp. 1-10, 2018.

[14] D. Appello et al., “Automatic functional stress pattern generation for SoC reliability characterization”, *IEEE European Test Symposium* (ETS), pp. 93-98, 2009.

[15] ISO 26262:2018 Road vehicles - Functional safety [Online]. Available: https://www.iso.org/standard/68383.html.

[16] (Jan. 2004) Tezzaron Semiconductor, “Soft Errors in Electronic Memory – A White Paper”, [Online]. Available: http://www.tezzaron.com.

[17] R. C. Baumann, “Radiation-induced soft errors in advanced semiconductor technologies”, *IEEE Transactions on Device and Materials Reliability*, vol. 5, no. 3, pp. 305-316, Sep. 2005.

Submitted 15.06.2021, accepted 08.10.2021.

Ֆունկցիոնալ անվտանգությանը համապատասխանող թեստավորման և վերանորոգման լուծում բյուրեղի վրա համակարգի կյանքի ցիկլի կառավարման համար

Գուրգեն Է. Հարությունյան, Սամվել Կ. Շուքուրյան, Գրիգոր Ա. Ճաղարյան և Երվանդ Ա. Զորյան

Սինոփսիս
e-mail: gurgen.harutyunyan@synopsys.com, samvel.shoukourian@ synopsys.com,
grigor.tshagharyan@synopsys.com, yervant.zorian@synopsys.com

Ամփոփում

Անվտանգության նկատմամբ զգայուն համակարգերի մասնաբաժինը էլեկտրոնային և էլեկտրական սարքերում տարբեր ոլորտներում, հատկապես ավտոմոբիլային արդյունաբերության մեջ, աճում է շարունակական տեմպերով: Համակարգի ներսում որևէ բաղադրիչի չվերահսկվող խափանումը կարող է վտանգել ամբողջ համակարգի ապահովությունը: Հետևաբար, անկախ օգտագործման կոնտեքստից և հիերարխիայի մակարդակից, համակարգի բոլոր բաղադրիչները պետք է հետևեն մշակման և աշխատանքային ռեժիմում շահագործման գործընթացի համար անվտանգության համապատասխան ստանդարտի պահանջներին: Այս համատեքստում ավանդական ներկառուցված ինքնաթեստավորման սխեման (ՆԹՀ) պետք է նաև համապատասխանի ISO 26262-ով սահմանված անվտանգության պահանջներին, որպեսզի թույլատրվի ավտոմոբիլային կիրառությունների համար: Հոդվածը ներկայացնում է ֆունկցիոնալ անվտանգության համապատասխան ՆԹՀ ենթակառուցվածքի հայեցակարգ, որը երաշխավորում է ապահով թեստավորում համակարգի աշխատանքի ամբողջ ցիկլի ընթացքում՝ պահպանելով թեստավորման բարձր որակը:

Բանալի բառեր՝ ֆունկցիոնալ անվտանգություն, ավտոմոբիլաշինություն, ISO 26262, ներկառուցված ինքնաթեստավորման համակարգ, թեստավորում աշխատանքային ռեժիմում, թեստավորում և վերանորոգում:

Система тестирования и ремонта, соответствующая требованиям функциональной безопасности для управления жизненным циклом системы на кристалле

Гурген Е. Арутюнян, Самвел К. Шукурян, Григор А. Джагарян и Ерванд А. Зорян

Синопсис
e-mail: gurgen.harutyunyan@synopsys.com, samvel.shoukourian@ synopsys.com,
grigor.tshagharyan@synopsys.com, yervant.zorian@synopsys.com

Аннотация

Доля критических с точки зрения безопасности систем в электронных и электрических устройствах во многих областях, особенно в автомобильной промышленности, растет на постоянной основе. Необработанный отказ любого из компонентов системы может поставить под угрозу безопасность всецелой системы. Поэтому, независимо от контекста использования и уровня иерархии, все компоненты системы должны соответствовать требованиям подходяшего стандарта безопасности для процесса разработки и эксплуатации в полевом режиме. В этом контексте традиционная схема встроенной системы самотестирования (ВСТ) также должна соответствовать требованиям безопасности, определенным в ISO 26262, чтобы быть одобренной для автомобильных приложений. В этой работе представлена концепция инфраструктуры ВСТ, отвечающая требованиям функциональной безопасности, которая помогает обеспечить безопасное тестирование на протяжении всего жизненного цикла системы на кристалле, при этом сохраняя высокое качество тестирования.

Ключевые слова: функциональная безопасность, автомобильная промышленность, ISO 26262, система встроенного самотестирования, тестирование в полевом режиме, тестирование и ремонт.