A Fault-tolerant Sequential Circuit Design for Soft Errors Based on Fault-Secure Circuit

S. Ostanin, A. Matrosova, N. Butorina, V. Lavrov
Tomsk State University
{sergeiostanin, mau11}@yandex.ru, nnatta07@mail.ru, neverlva@gmail.com

Abstract

This paper presents a fault-tolerant synchronous sequential circuit design based on fault-secure system with low overhead. The scheme has only one fault-secure sequential circuit, a normal (unprotected) sequential circuit, a checker and rather simple XOR circuit. It is proved the reliability properties of the suggested scheme not only for single stuck-at faults at gate poles but for path delay faults transient and intermittent. It is supposed that each next fault appears when a previous one has disappeared.

1. Introduction

In modern military, space, medical, etc. computer systems requirements for hardware reliability are increased. Continuous improvements in CMOS technology entering the nanometer scale has resulted into quantum mechanical effects creating many technological challenges for further scaling of CMOS devices. Nano-scale devices are limited by higher defect rates and increased susceptibility to soft errors (transient or intermittent). High performance integrated circuits have to be protected not only for single stuck-at faults (SAFs) (transient or intermittent) at gate poles but also for delays that arise in a circuit operation. Delays may be caused by a high level of circuit integration, low voltages and high frequency operation. One of the most widespread and useful in practice delay models is a model of a path delay fault (PDF). In this model, it is considered that for small delays in path elements and connections between its elements a delay in propagating a change in a signal value may exceed an admissible level for a circuit as a whole. This leads to incorrect operation of an entire circuit.

One of the approaches to increase reliability of the system is fault tolerance. A fault-tolerant system is one that can continue the correct performance of its specified tasks in the presence of faults. Fault tolerance is assumed to add some of the redundancy: hardware redundancy, software redundancy, information redundancy or time redundancy.

One of the most common techniques providing the fault-tolerant property is triple modular redundancy (TMR). The basic idea of TMR is to triplicate the circuit and perform a majority vote to determine the output of the system. The main difficulties with TMR are the voter (if the voter fails, the complete system fails) and high area overhead.

In the work [1] the synthesis of totally self-checking synchronous sequential circuits that are able to recover after an occurrence of soft errors is proposed.

A fault-tolerant system that is based on two replicas of a self-checking circuit and on error-masking interface has been suggested in [2]. They use two checkers and rather complicate error-masking interface containing flip-flops.

In [6] it is suggested a fault-tolerant sequential circuit design also based on two self-checking circuits. It includes two self-checking circuits, one self-testing checker and more simple error-masking interface than one in [2]. This technique was oriented towards soft single stuck-at faults at gate poles of sequential circuit and then was spread to soft PDFs of the circuit [4].

In [5] a fault-tolerant scheme based on totally self-checking system with low overhead in comparison with architectures suggested in [4] and [2] is considered. In contrast with these schemes it has only one self-checking combinational circuit and another circuit is conventional one. Such scheme implements the correct behavior of a combinational circuit when any permissible (among SAFs) soft fault (transient or intermittent) occurs. The reliability of the proposed scheme is higher than TMR systems or fault-tolerant systems based on two self-checking circuits.

In the paper [6] was proposed a fault-tolerant sequential circuit design based on self-checking module, unprotected module and checker, may be not self-testing, for stuck-at faults and path delay faults. The

978-1-5090-0693-9/16/$31.00 ©2016 IEEE

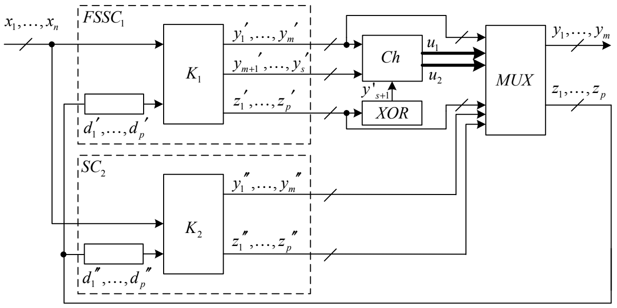

Figure 1. Fault-tolerant scheme

checker observes all outputs of combinational part of self-checking sequential circuit.

In this paper we suggest a modification of the architecture suggested in [6]. In new architecture is used a fault-secure module instead of self-checking module from [6]. A circuit is called a fault-secure iff outputs of the circuit realize error free code word or non-code word in presence of any fault from permissible set. It is sufficient for ensuring a fault-tolerant property of the system.

The scheme consists of one fault-secure sequential circuit, one normal (unprotected) sequential circuit, XOR circuit, not self-testing checker and multiplexor. The checker observes outputs of the fault-secure sequential circuit and one output of XOR circuit that defines parity (oddness) of code words for FSM states. Such scheme implements the correct behavior of a sequential circuit when any permissible (among SAFs and PDFs) transient or intermittent fault occurs.

This paper is structured as follows. In Section II the fault-tolerant scheme construction is described. Section III gives the analysis of the fault-tolerant properties for suggested scheme. Finally, conclusions are drawn in Section IV.

2. Fault-tolerant architecture

Let it has the State Transition Graph (STG) description of a synchronous finite state machine behavior. It is necessary to derive sequential circuit masking single stuck-at faults at gate poles of the circuit and delay faults.

We assume that each fault is transient or intermittent, and a next fault appears after a previous one has disappeared, and only one module of the fault-tolerant architecture can be faulty. We suggest applying the modified architecture of fault-tolerant scheme for sequential circuits from [6]. The implementation of a fault-tolerant sequential circuit shown in Fig. 1.

Here ***FSSC*₁** is a fault-secure sequential circuit. Assume that outputs under observation are primary output lines of sequential circuit and additional outputs providing an error detecting code.

We may use any technique providing the unidirectional error manifestation for combinational part of sequential circuit. Single stuck-at faults in combinational part of sequential circuit can be detected if results in unidirectional errors at the outputs and the outputs are encoded using a unidirectional error detecting code, for example, (k, l)-code (here l – length of code word and k – weight of code word) or Berger code. One of those techniques is suggested in [7].

Here we consider the technique described in [8, 9]. We encode FSM states with (q, p)-code words and then change each 0 value in a code word for don’t care. As symbols of an output alphabet have already been encoded we add output variables ($y'_{m+1},\dots, y'_s$) to encode sequential circuit outputs also by the proper (h, s)-code words. This encoding provides the unidirectional manifestation of single stuck-at faults at gate poles of a combinational part of sequential circuit when using the proper circuit design.

Considering sequential circuit as a whole (Fig. 1) we admit single stuck-at faults on flip-flop poles. These faults also manifest themselves as unidirectional ones. It means that ***FSSC*₁** is fault-secure sequential circuit for single stuck-at faults at gate poles of its combinational part and the same faults at flip-flop poles.

***SC*₂** is a sequential circuit realizing STG description of FSM. It has the same encoding states (like ***FSSC*₁**), but has no additional outputs that are used for providing unidirectional error detection.

***K*₂** is a combinational part of sequential circuit ***SC*₂**, implementing the system of partially monotonous Boolean functions (without functions corresponding to additional outputs). The circuit is derived by using any method that provides low cost realization. For increasing reliability properties of the system it is desirable applying different synthesis methods for ***FSSC*₁** and ***SC*₂**. Such approach allows decreasing a probability of appearance of identical faults.

Variables $y'_1,\dots, y'_m$, ($y''_1,\dots, y''_m$) correspond to outputs of sequential circuits; variables $y'_{m+1},\dots y'_{m+s}$ correspond to additional outputs for ***FSSC*₁** providing (h, s)-code words; variables $z_1',\dots, z_p'$ ($z''_1,\dots, z''_p$) are state ones, and $d_1',\dots, d'_p$ ($d''_1,\dots, d''_p$) are flip-flops corresponding to these variables.

XOR is a tree-like fan-out free 1-output subcircuit realizes function $z'_1\oplus\dots\oplus z'_p$. The output of subcircuit ***XOR*** y'_{s+1} gives value 1 (0) if proper state code words have odd (even) number of 1’s and opposite value for non-proper state code words.

Ch is an arbitrary checker for (h+1, s+1)-code [10, 11] in case of an odd number of 1’s proper state code words or (h, s+1)-code for an even number of 1’s. It may be not self-testing. It is supposed that each next fault appears when a previous one has disappeared. If checker fault manifests itself, it is masked by correct outputs of ***SC***$_2$. This fault does not effect on any next transient or intermittent fault. The checker detects erroneous code words on outputs: $y'_1,\dots,y'_m$, $y'_{m+1},\dots$, y'_{s+1}.

MUX is a multiplexor with control inputs u_1, u_2 and data inputs $y'_1,\dots,y'_m$, $z_1',\dots,z_p'$, $y''_1,\dots y''_m$, $z''_1,\dots,z''_p$. The ***MUX*** connects lines $y'_1,\dots,y'_m$, $z'_1,\dots,z'_p$ with lines $y_1,\dots y_m$, $z_1,\dots,z_p$ when checker outputs have "10" values otherwise the ***MUX*** connects lines $y''_1,\dots y''_m$, $z''_1,\dots,z''_p$ with lines $y_1,\dots,y_m$, $z_1,\dots,z_p$.

3. Fault-tolerance analysis

We consider single stuck-at faults at gate poles of the combinational parts of sequential circuit ***FSSC***$_1$ and its flip-flops, and single path delay faults of ***FSSC***$_1$. As for ***SC***$_2$ and ***Ch*** their faults may be arbitrary but any fault keeps circuit as combinational one. All above mentioned faults must be transient or intermittent, and a fault occurs one at a time and a next fault from permissible set can appear only after a forgoing fault has disappeared. Only one circuit among ***FSSC***$_1$, ***SC***$_2$, ***XOR***, ***Ch***, ***MUX*** may be faulty.

A. Stuck-at faults. Notice as V_{FSSC1} a set of permissible faults of ***FSSC***$_1$. V_{FSSC1} consists of single stuck-at faults at gate poles and single stuck-at faults at inputs and outputs of flip-flops. All these faults manifest themselves as unidirectional ones on outputs of subcircuit ***K***$_1$. If the fault v from V_{FSSC1} manifests at the outputs $y'_1,\dots,y'_m$, $y'_{m+1},\dots,y'_s$ that will be detected by the checker and multiplexer will use erroneous free outputs from ***K***$_2$. If the fault v manifests only at the outputs $z_1',\dots,z_p'$ and it changes parity (oddness) that will be detected by checker (by output of XOR circuit y'_{s+1}) during fault manifestation. In case parity (oddness) doesn't change consequences of the fault will extend to the next operation clock and will be detected or all consequences of the fault will disappear [1].

Let V_{XOR} be a set of arbitrary faults of XOR circuit. As XOR circuit has one output (y'_{s+1}) therefore it's any arbitrary fault leads to one-bit error at inputs of the checker and that will be detected.

Let V_{Ch} be a set of arbitrary faults of the checker. The fault-free checker for code words on inputs generates signals "10", for non-code words – "00", "01", "11". In presence of any fault from V_{Ch} the checker can produce arbitrary signals ("00", "11", "01", "10") at the outputs that drives multiplexer switching between error free outputs from ***K***$_1$ or ***K***$_2$.

Let V_{SC2} be a set of arbitrary faults of circuit ***SC***$_2$. It is supposed only one module of the system may be faulty. This means that other modules are fault-free and the multiplexor uses error free outputs from ***K***$_1$.

Let V_{MUX} be a set of permissible faults of the multiplexer. These faults can change connection of some lines $y'_1,\dots,y'_m$, $z'_1,\dots,z'_p$ for corresponding lines $y''_1,\dots y''_m$, $z''_1,\dots,z''_p$. In this case ***K***$_1$ and ***K***$_2$ are fault free and both have error free outputs.

Faults on primary inputs ($x_1, x_2,\dots,x_n$) and primary outputs $(y_1, y_2,\dots, y_m)$ are not considered.

Note $V = V_{\mathrm{FCSC1}} \cup V_{\mathrm{XOR}} \cup V_{\mathrm{Ch}} \cup V_{\mathrm{SC2}} \cup V_{\mathrm{MUX}}$.

Proposal 1. The scheme of Fig. 1 keeps correct functioning in the presence of any fault from V.

B. Path delay faults. Consider a combinational circuit in which at time moment t vector v_1 of values of input variables of a circuit is replaced by another vector v_2. Let τ be a maximal admissible path delay in the circuit. If in time period τ after the time moment t, the expected value of vector v_2 on the circuit output does not appear, we say that the circuit has path delay faults for some paths. We say that a pair (v_1, v_2) detects such fault, and the fault manifests itself on this pair.

We call a pair that detects a delay of a signal on a circuit output changing from 0 to 1 as a test for a rising transition; a delay of a signal on the circuit output changing from 1 to 0 as a test for a falling transition.

They distinguish single and multiple path delay faults, meaning faults of one or several paths, but we consider only single PDFs.

In [12] we reduce construction of a test pair for a PDF to testing the constant fault in the corresponding literal of the equivalent normal form (ENF) originated by the sub-circuit that contains the path considered. Dealing with a single literal fault we mean either turning the literal to constant 1 which leads to this literal disappearing from ENF products (*bp*-fault) or turning the literal to constant 0 which leads to disappearing all products that contain the literal (*ap*-fault).

A test pattern for this literal is vector v_2 of a test pair. Note that a PDF manifests itself on this test pattern only if previous vector v_1 differs from v_2 by a value of the variable marking the beginning of this path and possibly values of other variables. Otherwise this PDF does not manifest itself on the circuit output.

Note that only vector v_1 in a test pair indicates if the test pair is robust or non-robust. This vector does not directly affect PDF manifestation; it only provides manifestation of the PDF on vector v_2. Thus a PDF differs from the corresponding literal constant fault by manifestation not on each test pattern for the literal.

Take into consideration that disappearing of literal from ENF products increases on-set of the sub-circuit function, and disappearing of ENF products decreases on-set of the sub-circuit function. This means that a literal constant fault manifests itself in a combinational part K_1 of the scheme as a single stuck-at fault of this sub-circuit, but on the only sub-circuit output.

Masking PDFs we don't need to know what path is fault. That is why we do not differ robust testable and non-robust testable PDFs and we have to pay attention only to the literal originated by the path.

Note that for any path opposite delays of signals are feasible at the same time. Test patterns for constant 1 and constant 0 fault of the proper ENF literal are different. Consequently, opposite delays of the same path manifest themselves on different input vectors. Thus when PDF of a falling transition occurs, we observe 1 value instead of 0 value on the certain test pattern for *bp*-fault, and for a rising transition – 0 value instead of 1 value on the certain pattern for *ap*-fault.

As we consider transient or intermittent PDFs we may observe these faults only during time T, $T \geq \tau$. During time T path delay may manifest itself several times both for rising and falling transitions, but only on the same circuit output.

Let $V_{PDF} = V^{PDF}_{FSSC1} \cup V^{PDF}_{Others}$. Here V^{PDF}_{FSSC1} is a set of single path delay faults in $\boldsymbol{FSSC}_1$.

If the fault v from V^{PDF}_{FSSC1} manifests at the outputs $y'_1,\ldots,y'_m$, $y'_{m+1},\ldots$, y'_s that will be detected by the checker. If the fault v manifests only at the outputs $z_1',\ldots, z_p'$ then it changes parity (oddness) and that will be detected by checker (by output of XOR circuit y'_{s+1}).

V^{PDF}_{Others} is a set of PDFs in circuits ***Ch***, ***XOR*** and $\boldsymbol{K}_2$. If ***Ch*** or ***XOR*** has a PDF then combinational parts $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ are fault free. If $\boldsymbol{K}_2$ has a PDF then $\boldsymbol{K}_1$ is fault free. Both cases provide correct outputs.

Proposal 2. The scheme of Fig. 1 keeps correct functioning in the presence of any fault from V_{PDF}.

4. Conclusion

In this paper we have proposed a fault-tolerant sequential circuit design based on a fault-secure circuit with low overhead. The suggested scheme masks not only transient (intermittent) single stuck-at faults at gate poles but path delay faults as well.

5. Acknowledgment

The reported study was partially supported by Russian Science Foundation, research project № 14-19-00218.

Participation in EWDTS2016 is partially supported within the "Program of increase of the international competitiveness of Tomsk State U. for 2013-2020".

6. References

[1] I. Levin, A. Matrosova, and S. Ostanin, "Survivable Self-checking Sequential Circuits", *Int. Symp. DFT'01*, IEEE, San Francisco, USA, October 2001, pp. 395-402.

[2] M. Lubaszewski, B. Courtois, "A reliable fail-safe system", *Tran. On Comp.*, IEEE, v.47, №2, 1998, pp.236-241.

[3] A. Matrosova, V. Andreeva, Yu. Sedov, "Survivable discrete circuits design", *Int. On-Line Testing Workshop (IOLTW)*, IEEE, Bendor, France, 2002, pp. 13-17.

[4] A. Matrosova, S. Ostanin, I. Kirienko, E. Nikolaeva, "Fault-tolerant High Performance Scheme Design", *East-West Design & Test Symposium (EWDTS)*, IEEE, Batumi, Georgia, 2015, pp. 286-289.

[5] S. Ostanin, I. Kirienko, V. Lavrov, "Fault-Tolerant Combinational Circuit Design", *East-West Design & Test Symposium (EWDTS)*, IEEE, Batumi, Georgia, 2015, pp. 302-305.

[6] A. Matrosova, S. Ostanin, I. Kirienko, "A Fault-tolerant Sequential Circuit Design for SAFs and PDFs Soft Errors", *On-Line Testing Symposium (IOLTS)*, IEEE, Sant Feliu de Guixols, Spain, 2016, pp.1-2.

[7] F. Busaba, P. Lala, "Self-Checking Combinational Circuit Design for Single and Uniderectional Multibit Error", *JETTA*, no. 5, 1994, pp. 19-28.

[8] A. Yu. Matrosova, S.A. Ostanin, "Self-Checking Synchronous FSM network Design", *On-Line Testing Workshop (IOLTW)*, IEEE, Capri, Italy, 1998, pp. 162-166.

[9] A.Yu. Matrosova, I. Levin, S.A. Ostanin, "Self-checking synchronous FSM network design with low overhead", *VLSI Design*, Hindawi, Vol. 11, № 1, 2000, pp. 47-58.

[10] N. Butorina, S. Ostanin, "Implementation by the special formula of an arbitrary subset of code words of (m, n)-code for designing a self-testing checker", *East-West Design & Test Symposium (EWDTS)*, IEEE, Sevastopol; Ukraine, 2011, pp. 255-258.

[11] D. Efanov, V. Sapozhnikov, Vl. Sapozhnikov, A. Blyudov, "On the Problem of Selection of Code with Summation for Combinational Circuit Test Organization", *East-West Design & Test Symposium (EWDTS)*, IEEE, Rostov-on-Don, Russia, 2013, pp. 261-266.

[12] A. Matrosova, V. Lipsky, A. Melnikov, V. Singh, "Path delay faults and ENF", *East-West Design&Test Symposium (IOLTS)*, IEEE, St. Petersburg, Russia, 2010, pp. 164-167.