On Reducing Circuit Malfunctions Caused by Soft Errors[*]

Ilia Polian[1], Sudhakar M. Reddy[2], Irith Pomeranz[3], Xun Tang[2], Bernd Becker[1]

[1]Institute for Computer Science
Albert-Ludwigs-University
Freiburg, Germany
{polian|becker}informatik.uni-freiburg.de
[2]ECE Department
University of Iowa
Iowa City, Iowa, USA
{reddy\xutang@engineering.uiowa.edu}
[3]School of ECE
Purdue University
West Lafayette, Indiana, USA
{pomeranz@ecn.purdue.edu}

Abstract

Soft errors due to radiation are expected to increase in nano-electronic circuits. Methods to reduce system failures due to soft errors include use of redundancy and making circuit elements robust such that soft errors do not upset signal values. Recent works have noted that electronic circuits have partial intrinsic immunity to soft errors since single event upsets on a large percentage of signal lines do not cause errors on circuit outputs. Using ISCAS-89 benchmark circuits we present experimental evidence that the partial immunity to single event upsets is in most cases due to redundancy in the circuits and thus immunity to soft errors may not be available in irredundant circuits. Thus goals on immunity to soft errors may not be achievable in highly optimized circuits without adding circuit redundancy and/or relaxing the requirements on system failures due to soft errors.

[*] This work was supported in part by the DFG project RealTest (BE 1176/15-1), Alexander von Humboldt Foundation and by SRC Grants 2007-TJ-1642 and 2007-TJ-1643.

1. Introduction

Soft errors are a major factor limiting dependability of micro- and nanoelectronic circuits [Shivakumar 02]. While it is possible to protect circuits against soft errors, hardening solutions impose prohibitive costs in terms of area and power consumption. Recent publications suggest that a significant fraction of soft errors do not have any critical effect on [Wang 04, Li 07]. Moreover, some authors were even able to locate spots in the circuit such that a soft error on these spots did not result in critical errors on system outputs.

In [Polian 06], over 70% of the logic signals in an MPEG motion estimator were shown to be non-critical in the sense that soft errors on these signals led to errors on circuit outputs which were guaranteed to disappear after a few clock cycles. Thus single event upsets on a large majority of circuit lines in the MPEG motion detector effect circuit outputs for only a very limited time and specifically the errors caused by single event upsets disappear after only a few clock cycles. This concept was generalized in a probabilitic way in [Hayes 07]. In [Seshia 07], the properties which are used to formally verify a telecommunication chip are proven to hold even under errors in some of the flip-flops. In [Nowroth 08], errors on over 54% of the flip-flops in a JPEG compressor were shown not to result in perceptible reduction of image quality.

The fact that a significant fraction of spots in the system is non-critical, i.e., the system continues to operate meaningfully even in the presence of errors on these spots, allows the application of area- and power-efficient selective hardening techniques where only selected parts of a circuit are hardened against soft errors [Mohanram 03, Hayes 07]. On the other hand, it is counter-intuitive that errors on some parts of the system seemingly do not matter. In this paper, we study the possible reasons for the large number of non-critical spots. We evaluate the following three hypotheses.

First, the non-critical spots may be redundant. We call a spot redundant if its logical value has no influence on the circuit's operation. Permanent defects and soft errors on a redundant spot never produce effects on the circuit outputs. Since redundant spots can be eliminated from the design without changing its

functionality, most circuits contain no or very few redundant spots. Hence, we assume that only a small fraction of non-critical spots are redundant.

The second hypothesis is that the non-critical spots are single-cycle redundant, i.e., a soft error on that spot with a duration of one cycle is not propagated to the circuit outputs irrespective of the circuit's state or input values. This is equivalent to the error effect being eliminated by a mechanism called logical masking [Shivakumar 02]. There are two other masking mechanisms based on the observation that realistic soft errors are pulses of width significantly less than one clock cycle. Hence, single-cycle redundancy is a stronger requirement than immunity to the majority of the realistic soft errors, though it is a weaker requirement than redundancy. In general, single-cycle redundancy is not a sufficient condition to eliminate logic associated with a spot from the circuit.

The third hypothesis is that the non-critical spots are neither redundant nor single-cycle redundant. In this case, errors on non-critical spots do not violate the essential properties of the system either because no system errors occur under normal operating conditions or because the errors caused were within acceptable limits.

In this paper, we focus on the relationship between the first and the second class of the non-critical spots, i.e., the connection between redundancy and single-cycle redundancy. Intuitively, single-cycle redundancy is a weaker requirement than redundancy. Hence, one may be tempted to believe that there are more single-cycle-redundant spots than redundant spots. We generated empirical data to quantify the numbers of redundant and single-cycle redundant spots in ISCAS circuits to resolve this question.

The remainder of the paper is organized as follows. The method to identify redundant and single-cycle redundant spots is presented and the empirical data is reported in Section 2. The results are discussed in Section 3. Section 4 concludes the paper.

2. Identification of Redundant and Single-Cycle Redundant Spots

We determined the numbers of single-cycle-redundant and redundant spots in ISCAS-89 sequential benchmark circuits based on the observation that a (single-cycle-) redundant spot corresponds to a (single-cycle-) untestable fault on this spot. We used a modified version of a tool that identifies redundant stuck-at faults in sequential circuits [Reddy 99]. This tool identified almost all redundant spots in ISCAS-89 circuits. A spot is redundant if the fault-free circuit and the faulty circuits are synchronizable and there is no test to detect the fault [Pomeranz 96]. The method uses indirect implications derived through sequential static learning to determine that no tests exist under normal circuit operation using an Iterative Logic Array (ILA) of finite length (m+n) to model the sequential circuit [Reddy 99]. The ILA model used is shown in Figure 1 given below. It should be noted that the fault being considered is injected only in the last m right most cells and the n left most cells cells are fault-free. If the effect of the fault under consideration cannot be propagated to a primary output or to a the pseudo-primary outputs (the horizontal outputs of the right most cell of the ILA) then the fault is untestable [Reddy 99] and also redundant if the faulty circuit is synchronizable [Pomeranz 96]. To establish that a single-cycle fault is redundant we use the ILA of Figure 2 in which the target fault is injected only in the (m+1)th cell and the requirement that the faulty circuit is synchronizable is removed since single cycle faults do not effect synchronizability of a circuit.

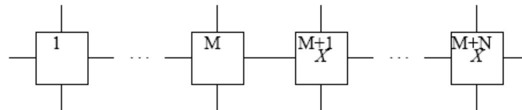

Figure 1: ILA model used to determine redundant spots

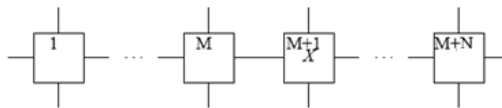

Figure 2: ILA model used to determine single-cycle redundant spots

3. Experimental Results

The results on redundant spots and single cycle redundant spots are given in Tables 1 and 2. We used the set of uncollapsed single stuck-at faults in the combinational logic of the synchronizable ISCA-89 benchmark sequential circuits. In Table 1, after the circuit name we give the total number of stuck-at faults followed by the number of (permanent) redundant faults and single-cycle-redundant faults. In reporting the numbers of faults we consider stuck-at-0 and stuck-at-1 as separate faults. It can be seen that the numbers of stuck-at faults that are redundant and the number of single cycle redundant faults are the same in many of the circuits. In thirteen circuits (shown by entries in bold) where the numbers of single cycle redundant faults are higher than the number of redundant stuck-at faults the difference is small with few exceptions. For all the circuits in our experiment the set of single-cycle-redundant faults contains the set of redundant stuck-at-faults. However, we have not been able to prove that this observation will hold in general.

In Table 2 we give the number of flip-flops at which the output stuck-at faults are redundant or the single-cycle redundant faults at the output of the flip-flops are redundant. After the circuit name we give the total number of delay flip-flops (DFFs) in the circuit followed by the numbers of stuck-at or single-cycle redundant faults at the outputs of flip-flops. In the last two columns we give the numbers of DFFs at the outputs of which both stuck-at and stuck-at faults are redundant and both the single-cycle faults are redundant. These are the flip-flops that need not be augmented to be fault-tolerant to transient faults as suggested in [Seshia 07]. From Table 2 it can be seen that number of circuits in which flip-flop outputs that have redundant stuck-at and/or single-cycle redundant faults are small. Furthermore the numbers of redundant and single cycle redundant faults at the outputs of flip-flops are identical.

Table 1: Redundant faults in combinational logic

circuit	total number of faults	permanent redundant	single cycle redundant
S208	**416**	**121**	**124**
S298	596	60	60
S344	670	18	18
S349	680	22	22
S382	764	8	8
S386	**772**	**76**	**81**
S400	800	20	20
S420	**840**	**438**	**447**
S444	888	30	30
S499	954	163	163
S526	1052	56	56
S641	**1278**	**140**	**152**
S713	**1426**	**218**	**230**
S820	**1640**	**60**	**71**
S832	**1664**	**79**	**96**
S838	**1676**	**1070**	**1107**
S953	**1906**	**10**	**14**
S967	**1928**	**21**	**24**
S1196	**2392**	**3**	**5**
S1238	2476	83	83
S1269	**2522**	**3**	**4**
S1423	2846	40	40
S1488	2976	68	68
S1494	2988	94	94
S3384	**6750**	**1**	**17**
S4863	9598	210	210
S5378	10590	2112	2112
S35932	71224	7344	7344

Table 2: Redundant faults in D-flip-flops

circuit	total number of DFFs	redundant DFF faults		double redundant DFFs	
		permanent	single cycle	permanent	single cycle
S208	8	0	0	0	0
S298	14	0	0	0	0
S344	15	0	0	0	0
S349	15	0	0	0	0
S382	21	0	0	0	0
S386	6	0	0	0	0
S400	21	0	0	0	0
S420	16	11	11	0	0
S444	21	0	0	0	0
S499	22	0	0	0	0
S526	21	0	0	0	0
S641	19	0	0	0	0
S713	19	0	0	0	0
S820	5	0	0	0	0
S832	5	0	0	0	0
S838	32	27	27	0	0
S953	29	0	0	0	0
S967	29	0	0	0	0
S1196	18	0	0	0	0
S1238	18	0	0	0	0
S1269	37	0	0	0	0
S1423	74	0	0	0	0
S1488	6	0	0	0	0
S1494	6	0	0	0	0
S3384	183	0	0	0	0
S4863	104	0	0	0	0
S5378	179	79	79	23	23
S35932	1728	0	0	0	0

3. Discussion

In the benchmark circuits we studied the sizes of the sets of single-cycle-redundant and redundant spots are very similar for combinational logic and always identical for flip-flops. Thus, soft error immune signal lines of the circuits are essentially the same as the redundant logic lines of the circuits. This suggests that there is very little difference between redundancy and single-cycle redundancy in practice. If a spot is single-cycle redundant, it is most probably also redundant and can be eliminated. On the other hand, given that a circuit is optimized and thus contains very little redundant spots, it also contains very little single-cycle redundant spots. Hence, hypothesis two is as unlikely to explain the presence of a large number of non-critical spots as hypothesis one. Consequently, hypothesis three must be valid: most non-critical spots are neither redundant nor single-cycle redundant.

The observations above seem to suggest that the circuits analyzed were unintentionally over-specified and, consequently, over-designed. In our opinion, the key to the design of dependable systems is the understanding of critical versus non-critical errors with respect to the specification. For instance, soft errors in performance-enhancing modules of a microprocessor, such as branch predictors, are typically non-critical, except in application with hard real-time constraints. Tolerating occasional pixel deviations in imaging application such as video processing may be preferred to spending additional hardware and power consumption for massive redundancy needed to prevent the errors from happening. A range of applications, e.g., in communication, can handle certain classes of errors at the system level. Additional fields in which this concept appears to be applicable include recognition, mining, synthesis, tracking and control.

In our opinion, design and implementation of cost-effective nanoelectronic systems will have to incorporate dependability aspects with respect to soft error resilience as part of specification. We need novel methods of *specification-aware synthesis*, which satisfy the dependability requirements by a carefully chosen mix of techniques such as hardware redundancy, commit-rollback recovery, error-resilient information coding and algorithmic fault tolerance in software. This is in contrast to today’s approach where a circuit is designed first and then its

dependability is enhanced without modifying the circuit's Boolean function. The future specification formalisms must distinguish between functionality considered as essential, and functions that are allowed to fail occasionally due to soft errors.

4. Conclusions

We demonstrated that the concepts of redundancy and single-cycle redundancy essentially fall together in practice. As a consequence, most non-critical spots are neither redundant nor single-cycle redundant. This suggests large headroom for design of resilient systems which does not rely exclusively on hardware fault tolerance. Without loosening the system specification, employing massive redundancy for full protection of the circuits cannot be avoided.

References

[Hayes 07] J. Hayes, I. Polian, and B. Becker, An analysis framework for transient-error tolerance, VLSI Test Symp., 2007.

[Li 07] X. Li and D. Yeung. Application-level correctness and its impact on fault tolerance. Int'l Symp. on High Perf. Comp. Arch., 2007.

[Mohanram 03] N. Mohanram and N. Touba. Partial error masking to reduce soft error failure rate in logic circuits. Int'l Symp. on DFT, 2003.

[Nowroth 08] D. Nowroth, I. Polian and B. Becker. A study of cognitive resilience in a JPEG compressor. Int'l Conf. on Dependable Systems and Networks, 2008

[Polian 06] I. Polian et al., Low-cost hardening of image processing applications against soft errors, Int'l Symp. on DFT, 2006.

[Pomeranz 96] I. Pomeranz and S.M. Reddy, On removing redundancies from synchronous sequential circuits with synchronizing sequences, IEEE Trans. on Computers, pp. 20-32, Jan. 1996.

[Reddy 99] S.M. Reddy, I. Pomeranz, X. Lin and N.Z. Basturkmen, New procedures for identifying undetectable and redundant faults in synchronous sequential circuits, VLSI Test Symp., 1999.

[Seshia 07] S.A. Seshia, W. Li, and S. Mitra. Verification-guided soft error resilience. DATE Conf., 2007.

[Shivakumar 02] P. Shivakumar et al., Modeling the effect of technology trends on the soft error rate of combinational logic, Int'l Conf. on Dependable Systems and Networks, 2002.

[Wang 04] N.J. Wang, et al.. Characterizing the effects of transient faults on a high-performace processor pipeline. Int'l Conf. on Dependable Systems and Networks, 2004.