An Soft Error Propagation Analysis Considering Logical Masking Effect on Re-convergent Path

Shuhei Yoshida, Go Matsukawa, Shintaro Izumi, Hiroshi Kawaguchi, and Masahiko Yoshimoto
Graduate School of System Informatics, Kobe University
1-1 Rokkodai, Nada, Kobe, 657-8501, Japan
E-mail: yoshida.shuhei@cs28.cs.kobe-u.ac.jp

***Abstract*—This paper presents an accurate soft error propagation analysis technique. Especially, we focus on Single Event Upset (SEU) in flip-flop. The proposed technique can calculate the accurate error propagation probability considering logical masking on re-convergent paths with SAT solver efficiently. Experimental result shows that the proposed technique improves the computation time by 94.6% compared with the method with only SAT solver and the accuracy by 93.3% compared with the conventional method respectively.**

Keywords*—*single event effect; single event upset; soft error propagation; logical masking; temporal masking

I. Introduction

Recently, highly dependable VLSI processors have played a significant role for system implementation in various mission-critical application fields. On the other hand, process technology scaling degrades soft error tolerance of VLSI processors [1]. Therefore, soft error mitigation techniques must be applied to VLSI processors to maintain soft error tolerance. However, verifying efficacy of applied soft error mitigation technique is not easy. Radiation testing using test chip requires great deal of time and cost. [2] Consequently, an evaluation technique of processor SER without the radiation testing is required.

An accurate analysis technique of soft error propagation is essential in order to evaluate a processor SER [3]. This paper describes an accurate analysis technique of soft error propagation. The soft error propagation analysis evaluates the probability that soft error propagates from the erroneous flip-flop to the downstream flip-flop. Existing techniques are fault injection techniques [4–6] and analytical techniques [3, 8–11]. Fault injection simulation using comprehensive input vectors takes a large amount of time because soft error propagation depends on input vector sequences in the case of sequential circuits. The number of test cases is enormous even for small circuits. Monte-Carlo simulation using random input vectors cannot ensure that all circuit states are verified uniformly. FPGA-based fault injection for speed up [5, 6] cannot evaluate temporal effects depending on circuit delays. Analytical techniques estimate the state of the circuit probabilistically without the test cases for each input vector sequence and are practical techniques for fast and accurate soft error propagation analysis.

The remainder of this paper is organized as follows. Section II provides the brief information about SEU propagation and conventional soft error propagation analysis techniques. Section III describes a problem of conventional analysis techniques and presents the proposed analysis technique. Section IV presents experimental results. Section V gives the conclusion.

II. Conventional Technique for Soft Error Propagation Analysis

SEUs in Flip-flops might not be captured in downstream flip-flops because it could be masked temporally and logically. These masking effects are called temporal masking and logical masking respectively. The probability that the error propagates to the downstream flip-flop considering these two masking effects, defined as *MF* (*Masking Factor*), is calculated as follows:

$$MF = TMF \times LMF \quad (1)$$

where *TMF* (*Temporal Masking Factor*) is the error propagation probability considering temporal masking and *LMF* (*Logical Masking Factor*) is the error propagation probability considering logical masking. Methods for analyzing each masking effect have been proposed individually. This section describes conventional analysis techniques for each masking effect.

A. Temporal Masking

If an SEU occurs after a certain period in a clock cycle and the error does not have time to propagate to the downstream flip-flop before the next clock edge, the error is not captured by the downstream flip-flop. As a result, the error is overwritten by the error-free data [13]. This masking effect is temporal masking. The error propagation probability considering temporal masking, *TMF*, depends on occurrence timing of SEU and propagation delays between flip-flops. [7]. *TMF* is the ratio of a vulnerable window (T_{error}) to a clock cycle (T_{cycle}). An SEU which occurs in the vulnerable window can propagate to the downstream flip-flop temporally. In Fig. 1, t_{setup} and t_{hold} are setup time and hold time of the downstream flip-flop respectively. t_{prop} is the sum of the delays of the erroneous flip-flop and the combinational logic. α is a constant value from 0 to 1, which depends on leaching characteristics of the downstream flip-flop. *TMF* is calculated as follows:

$$TMF = \frac{T_{error}}{T_{cycle}} = \frac{T_{cycle} - t_{setup} - t_{prop} + \alpha(t_{setup} + t_{hold})}{T_{cycle}} \quad (2)$$

The propagation delays are evaluated by Static Timing Analysis (STA).

In addition, the method proposed in [13] considers temporal masking disablement, which occurs when data latching of the erroneous flip-flop is controlled by the enabling signal. The effect can be evaluated by the probability that the erroneous flip-flop enables at clock edges, defined as $p(enable)$. TMF is compensated as follows:

$$TMF_{comp} = p(enable) \times TMF_{org} + (1 - p(enable)) \tag{3}$$

where TMF_{comp} is compensated TMF and TMF_{org} is the probability calculated by the equation (2).

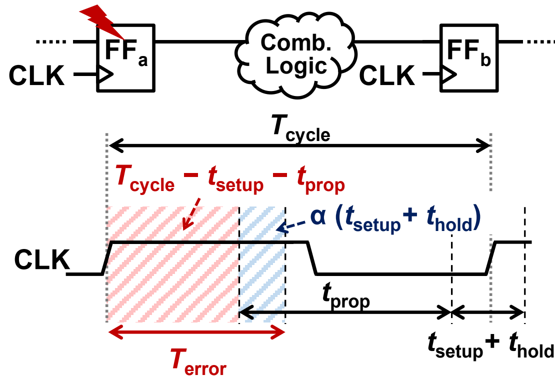


Fig. 1. An analysis technique for temporal masking.

B. Logical Masking

If an error propagate to a gate input of which value does not affect the gate output logically (e.g. one input of two-input AND gate when the other input is "0"), the error is masked. This masking effect is logical masking. Whether or not an error is masked logically depends on the input vector of the circuit. As described before, analytical techniques are practical for logical masking analysis [8–11].

The analysis technique using Binary Decision Diagrams (BDD) analyzes the Boolean function of combinational logic and error propagation efficiently [8]. The technique proposed in [9] evaluates the logical behavior of the circuit and the error propagation by Probabilistic Transfer Matrix (PTM) and Ideal Transfer Matrix (ITM). In [10], the error propagation considering logical masking is evaluated by error propagation rules using four-value logic. However, the accuracy of this technique degrades due to re-convergent paths because the signal correlation of errors is not considered on re-convergent paths. L. Chen proposes the analysis method considering signal correlation to improve the accuracy[11]. In this technique, error propagation is modeled by Boolean function. The model is implemented by combinational logic and connected to the circuit under evaluation. The error propagation probability considering logical masking is calculated by signal probability in the circuit. Signal probability $p(x)$ is the probability that a signal x is "1" and estimated by Correlation Coefficient Method (CCM) proposed in [12].

III. Proposed Technique for Soft Error Propagation Analysis

A. Signal Probability Calculation with SAT Solver

This paper proposes the method to calculate accurate LMF based on the method proposed in [11]. In [11], there is a problem that the accuracy of calculated LMF degrades when the circuit under evaluation includes re-convergent paths. The degradation of the accuracy is caused by the inaccuracy of signal probability estimated by CCM. In CCM, signal probability of a gate output is estimated considering only first-order correlations between the gate inputs. Therefore, signal probability cannot be estimated with accuracy in case that a target node is a sink gate output of a re-convergent path with three or more paths from the source node as shown in Fig. 2. In the proposed technique, accurate signal probability of such gate output is calculated with SAT solver, which can solve the Boolean Satisfiability Problem (SAT) effectively. An example of a signal probability calculation by proposed scheme is shown in Fig. 3. The formula for calculation of $p(Y)$, which is signal probability of Y, is given from the truth table of the given logic circuit. $p(Y)$ is calculated from $p(X_0)$, $p(X_1)$, and $p(X_2)$, which are signal probability of the gate inputs. A SAT solver used in the proposed scheme is required to find all satisfying solutions.

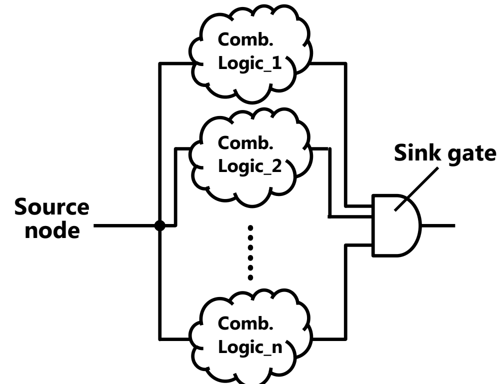
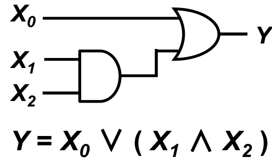


Fig. 2. A re-convergent path with three or more paths from the source node.

B. Efficient Signal Probability Calculation

As described above, the calculation method with SAT solver can calculate accurate signal probability. However, the computation time required for this method increases exponentially along with a circuit scale. To reduce the computation time, we propose the combined method of CCM, which requires polynomial time for calculating signal probability, and the calculation method with SAT solver. In the proposed method, signal probability of a sink gate output on a re-convergent path as shown in Fig. 2 is calculated with SAT solver. If target node is not a output of sink gate on re-convergent path, its signal probability is calculated by CCM because CCM calculate accurate signal probability of such node. Consequently, the proposed method realizes reduction of a computation time while maintaining accuracy. If accuracy degradation is allowed, further reduction of the computation time is possible by restricting the length of a re-convergent

path on which signal probability of the sink gate output is calculated with SAT solver.



(a)

X_0	X_1	X_2	Y
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

$$\begin{aligned} p(Y) = & \; p(1-p(X_0)) \cdot p(X_1) \cdot p(X_2) \\ & + p(X_0) \cdot (1-p(X_1)) \cdot (1-p(X_2)) \\ & + p(X_0) \cdot (1-p(X_1)) \cdot p(X_2) \\ & + p(X_0) \cdot p(X_1) \cdot (1-p(X_2)) \\ & + p(X_0) \cdot p(X_1) \cdot p(X_2) \end{aligned}$$

(b)

Fig. 3. An method for calculating signal probability with SAT solver.

IV. EXPERIMENTAL RESULT

This section shows comparison results of the proposed technique and the conventional technique in terms of accuracy and computation time. In both techniques, the error propagation probability (*MF)* is calculated by the equation (1) and the error propagation probability considering temporal masking (*TMF*) is calculated by the equation (3). In conventional technique, *LMF* is calculated by Correlation Coefficient Method (CCM). However, in the proposed technique, *LMF* is calculated by the proposed scheme described in Section III. In this evaluation, we use the SAT solver proposed in [14], which can find all satisfying solutions efficiently.

A. Accuracy Evaluation

This section shows accuracy comparison of the error propagation probability. The probability calculated by each method are compared with the probability calculated by SEU injected Monte-Carlo simulation respectively. The accuracy is evaluated by *Mean Absolute Error* (*MAE*) as follows:

$$MAE = \frac{1}{N} \sum_{i} \left| MF_{sim}(i) - MF(i) \right| \qquad (4)$$

where N is the number of pairs of the erroneous flip-flop and the downstream flip-flop, $MF(i)$ is the error propagation probability evaluated by the conventional technique or the proposed technique for pair i, and $MF(i)$ is the probability evaluated by the Monte-Carlo simulation for pair i.

Circuits for this experiment are benchmark circuits from ITC'99 and ISCAS'89 implemented in 65-nm process. In the evaluation, the circuit from the erroneous flip-flop to the downstream flip-flop is extracted from the benchmark circuit because the analysis by Monte-Carlo simulation is difficult for sequential circuits as described in section I. The extracted circuit can be evaluated by Monte-Carlo simulation because it is substantially combinational circuit. The clock frequency is 1 GHz. The propagation delays between flip-flops are calculated by STA using Synopsys Design Compiler. α in the equation (2) is configured as 0.5 in the same value as [7]. Monte-Carlo simulation is conducted using the gate-level circuits annotated delay information. SEU injection timing is varied uniformly in the unit of 1 ps in a cycle. p(*enable*) is assumed that 0.1, 0.3, and 0.5 because the benchmark circuits do not have flip-flop controlled by enabling signal and clock-gating scheme.

The comparison results are depicted in Fig. 4. The proposed technique is more accurate for all benchmark circuits than the conventional technique when p(*enable*) is 0.1, 0.3, and 0.5. The proposed technique improves average *MAE* by 97.5%, 92.5%, and 87.7% when p(*enable*) is 0.1, 0.3, and 0.5, respectively. The accuracy of *TMF* becomes better as p(*enable*) decreases because the effect of *TMF* calculated by equation (2) is smaller by compensation with lower p(*enable*). Consequently, the whole accuracy rate of the proposed technique becomes worse as p(*enable*) decreases.

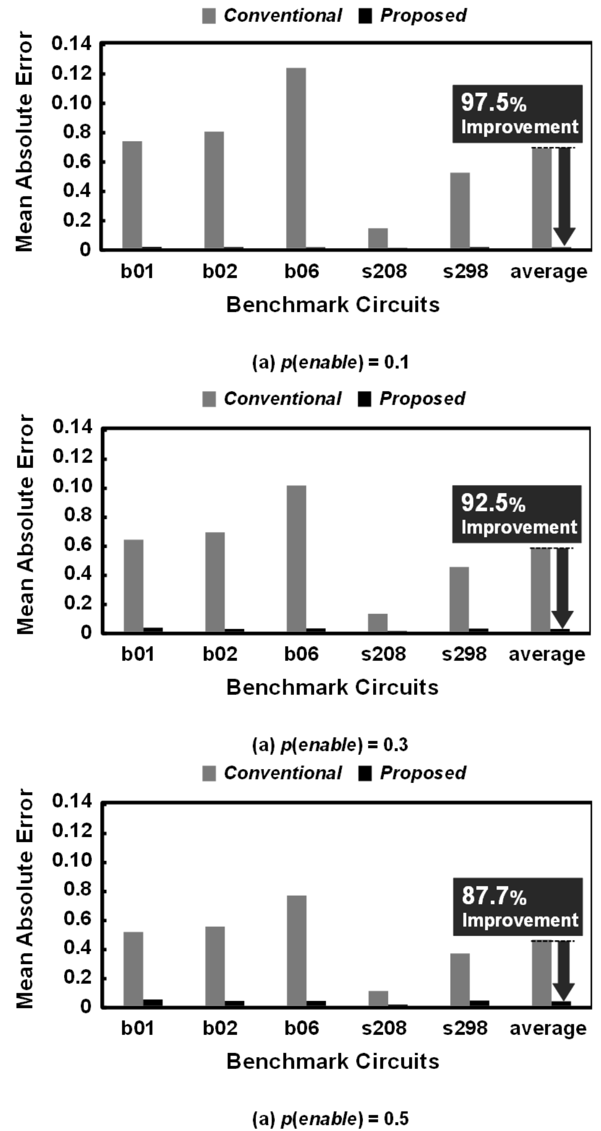


Fig. 4. Accuracy comparison of the conventional technique and the proposed technique when p(*enable*) is (a) 0.1, (b) 0.3, and (c) 0.5.

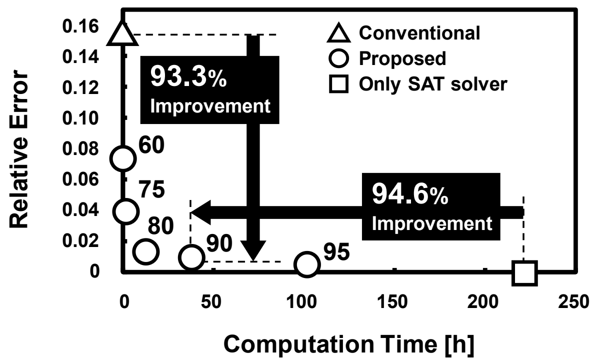


Fig. 5. Computation time comparison of the conventional technique and the proposed technique.

B. Computation Time Evaluation

This section shows comparison of the proposed technique and the conventional technique in terms of computation time. Simultaneously, accuracy is evaluated because computation time and accuracy is a trade-off relation. The difference of the proposed technique and the conventional technique is the method for calculating *LMF*. Furthermore, computation time of calculating *LMF* accounts for a large part of the whole computation time. Consequently, computation times for calculating *LMF* are compared in this evaluation. As described in Section III, the proposed technique can reduce computation time by restricting the length on which signal probability of the sink gate output is calculated with SAT solver. The maximum length is denoted by L_{max}. In this experiment, the proposed technique which L_{max} is 60, 75, 80, 90, and 95 are evaluated. In addition, a method with only SAT solver is evaluated. This method calculates signal probability of a primary output in the circuit with SAT solve without the scheme for reducing computation time described in Section III. The accuracy is evaluated by relative error. The truth value is signal probability calculated by the method with only SAT.

Circuits for this experiment is the benchmark circuit c880, which is a combinational circuit and consists of 383 gates, from ISCAS'85. All the experiments have been performed on an Intel® Core™ i7 875K processor with 4GB RAM. The comparison results are depicted in. 5. The result shows that the proposed technique which L_{max} is 90 improves the computation time by 94.6% compared with the method with only SAT solver and the accuracy by 93.3% compared with the conventional method respectively. This shows that the proposed technique can reduce computation time without accuracy degradation.

V. CONCLUSION

This paper presents an accurate soft error propagation analysis technique with SAT solver. The propose technique reduce computation time by restricting the number of nodes of which signal probability is calculated with SAT solver. The result of accuracy evaluation shows that the proposed technique realizes 97.5% better accuracy of the error propagation probability when $p(enable)$ is 0.1. The result of computation time evaluation shows that the proposed technique which L_{max} is 90 improve the computation time by 94.6% compared with the method only using SAT solver and the accuracy by 93.3% compared with the conventional method respectively.

REFERENCES

[1] P. Shivakumar, M. Kistler, S.W. Keckler, D. Burger, and L. Alvisi, "Modeling the effect of technology trends on the soft error rate of combinational logic," in Proceedings of International Conference on Dependable Systems and Networks, pp. 389–398, 2002.

[2] B. Gill, N. Seifert, and V. Zia, "Comparison of Alpha-particle and Neutron-induced Combinational and Sequential Logic Error Rates at the 32nm Technology Node," in Proceedings of IEEE International Reliability Physics Symposium, pp. 199–205, 2009.

[3] M. Ebrahimi, A. Evans, M.B. Tahoori, R. Seyyedi, E. Constenaro, and D. Alexandrescu, "Comprehensive Analysis of Alpha and Neutron Particle-induced Soft Errors in an Embedded Processor at Nanoscales," in Proceedings of Design, Automation and Test in Europe Conference and Exhibition, pp. 1–6, 2014.

[4] W. Mansour and R. Velazco, "An Automated SEU Fault-Injection Method and Tool for HDL-Based Designs," IEEE Transactions on Nuclear Science, vol.60, no. 4, pp. 2728–2733, 2013.

[5] A. Mohammadi, M. Ebrahimi, A. Ejlali, and S.G. Miremadi, "SCFIT: A FPGA-based fault injection technique for SEU fault model," in Proceeding of Design, Automation and Test in Europe Conference and Exhibition, pp. 586–589, 2012.

[6] C. L. Ongil, M.G. Valderas, M.P. Garcia, and L. Entrena, "Autonomous Fault Emulation: A New FPGA-Based Acceleration System for Hardness Evaluation," IEEE Transacions on Nuclear Science, vol. 54, no. 1, pp. 252–261, 2007.

[7] D. Alexandrescu, "Soft Errors in modern electronic systems, Chapter 5," Springer, 2011.

[8] N. Miskov-Zivanov, P.A. Pittsburgh, and D. Marculescu, "Soft Error Rate Analysis for Sequential Circuits," in Proceedings of Design, Automation and Test in Europe Conference and Exhibition, pp. 1–6, 2007.

[9] D.T. Franco, M.C. Vasconcelos, L. Naviner, and J.-F. Naviner, "Signal probability for reliability evaluation of logic circuits," Microelectronics Reliability, vol. 48, no. 8–9, pp. 1586–1591, 2008.

[10] H. Asadi and M.B. Tahoori, "Soft Error Derating Computaion in Sequential Circuits," in Proceedings of IEEE/ACM International Conference on Computer-Aided Design, pp. 497–501, 2006.

[11] L. Chen, M. Ebrahimi, and M.B. Tahoori, "CEP: Correlated Error Propagation for Hierarchical Soft Error Analysis," Journal of Erectronic Testing, vol. 29, no. 2, pp. 143–158, 2013.

[12] S. Ercolani, M. Favalli, M. Damiani, P. Olivo, and B. Ricco, "Estimate of signal probability in combinational logic networks," in Proceedings of European Test Conference, pp. 132–138, 1989.

[13] Y. Kimi, G. Matsukawa, S. Yoshida, S. Izumi, H. Kawaguchi, and M. Yoshimoto, "An Accurate Soft Error Propagation Analysis Technique Considering Temporal Masking Disablement," in Proceedings of On-Line Testing Symposium, pp. 23–25, 2015.

[14] T. Toda, and K. Tsuda, "BDD Construction for All Solutions SAT and Efficient Caching Mechanism," in Proceedings of the 30th Annual ACM Symposium on Applied Computing, pp. 1880–1886, 2015.