IEICE TRANS. FUNDAMENTALS, VOL.E99–A, NO.6 JUNE 2016

PAPER

Error Propagation Analysis for Single Event Upset considering Masking Effects on Re-Convergent Path

Go MATSUKAWA[†a)], ***Student Member***, **Yuta KIMI**[†], **Shuhei YOSHIDA**[†], ***Nonmembers***, **Shintaro IZUMI**[†], **Hiroshi KAWAGUCHI**[†], ***and*** **Masahiko YOSHIMOTO**[†], ***Members***

SUMMARY As technology nodes continue to shrink, the impact of radiation-induced soft error on processor reliability increases. Estimation of processor reliability and identification of vulnerable flip-flops requires accurate soft error rate (SER) analysis techniques. This paper presents a proposal for a soft error propagation analysis technique. We specifically examine single event upset (SEU) occurring at a flip-flop in sequential circuits. When SEUs propagate in sequential circuits, the faults can be masked temporally and logically. Conventional soft error propagation analysis techniques do not consider error convergent timing on re-convergent paths. The proposed technique can analyze soft error propagation while considering error-convergent timing on a re-convergent path by combinational analysis of temporal and logical effects. The proposed technique also considers the case in which the temporal masking is disabled with an enable signal of the erroneous flip-flop negated. Experimental results show that the proposed technique improves inaccuracy by 70.5%, on average, compared with conventional techniques using ITC 99 and ISCAS 89 benchmark circuits when the enable probability is 1/3, while the runtime overhead is only 1.7% on average.

key words: *single event effect, single event upset, soft error propagation, logical masking, temporal masking*

1. Introduction

Technology scaling degrades the critical charge, decreases operating voltage and increases the circuit scale on a chip. For those reasons, soft errors have come to occur easily in recent years [1]. Accurate soft error rate (SER) estimation is therefore necessary to design soft-error-tolerant processors. However, verifying the effectiveness of soft error mitigation techniques is difficult because processors have become extremely complicated. Radiation testing of fabricated test chip, which entails a great deal of time and cost, is conducted only for simple test circuits [2], [3]. Moreover, system-level verification by radiation testing is difficult. Consequently, a processor SER evaluation technique that uses no radiation testing is required.

In the past, soft errors in large SRAM arrays were regarded as the main contributor to processor SER. However, error correction code (ECC) with a bit-interleave structure can mitigate SRAM SER sufficiently [4]. In recent years, soft errors have come to occur in logic blocks, creating growing concern about processor reliability [4], [5]. For soft error of logic blocks, we particularly examine single event upsets (SEUs) in flip-flops. Radiation-hardened flip-flops are affected by area and power overhead. Therefore, it is inefficient to apply soft error mitigation techniques uniformly to all flip-flops. Contributions to processor SER differs among flip-flops. Consequently, identifying flip-flops that are susceptible to soft error can efficiently mitigate the impact of soft error on processors because developers can apply soft error mitigation techniques selectively to these flip-flops and design soft-error-tolerant and cost-effective processors.

Soft errors occur in flip-flops and propagate to downstream flip-flops through combinational circuits. There, the error might cause a primary output flip and system-level malfunction. The processor SER evaluation should be conducted using the following three steps: device-level soft error occurrence analysis, circuit-level propagation analysis, and system-level effect analysis [4]. As described herein, we propose an accurate analysis technique for soft error propagation. It evaluates the probability that an SEU occurring at a flip-flop flips the value in a downstream flip-flop and thereby affects the primary output. Existing techniques are classifiable into fault injection techniques [6]–[9] and analytical techniques [4], [12]–[15]. Exhaustive fault injection techniques require vastly numerous input vectors. Fault injection simulation using comprehensive input vectors for a complicated circuit takes an enormous amount of time because the number of input vectors increases exponentially according to the number of primary inputs. Monte Carlo simulation using random input vectors cannot ensure that all circuit states are verified uniformly. FPGA-based fault injection for speeding up [8], [9] cannot evaluate temporal masking effects depending on circuit delays. Analytical techniques estimate the circuit state probabilistically without a test case for each input vector sequence. They are suitable for analysis of complicated processors. First we describe an effect that conventional analytical techniques cannot evaluate. Then we propose an analytical technique that can evaluate the effect.

The remainder of this paper is organized as follows. Several masking effects and conventional SEU propagation analysis techniques are described in Sect. 2. Section 3 explains problems of the conventional and proposed techniques. Section 4 shows the experimental setup and results. Finally, Sect. 5 presents a summary of the results.

Manuscript received September 14, 2015.
Manuscript revised February 4, 2016.

[†]The authors are with Kobe University, Kobe-shi, 657-8501 Japan.
a) E-mail: matsukawa@cs28.cs.kobe-u.ac.jp

DOI: 10.1587/transfun.E99.A.1198

2. Conventional Methods for Soft Error Propagation Analysis

Actually, SEUs in flip-flops do not always propagate to the downstream flip-flops. The possibility exists that SEUs are masked temporally and logically. These masking effects are designated respectively as temporal masking and logical masking. Therefore, the probability that the error propagates to the downstream flip-flop, defined as the *Masking Factor* (*MF*), is calculated as shown below.

$$MF = TMF \times LMF \tag{1}$$

Therein, the *Temporal Masking Factor* (*TMF*) is the error propagation probability considering temporal masking. The *Logical Masking Factor* (*LMF*) is the error propagation probability considering logical masking. Analytical techniques have been proposed, but the conventional technique evaluates these masking effects individually. This section presents a description of these two masking effects and conventional analysis techniques.

2.1 Temporal Masking

When an SEU occurs at a certain time in a clock cycle, the SEU cannot be sampled temporally at the downstream flip-flops before the next clock edge. Therefore, the downstream flip-flop does not latch the error; the erroneous flip-flop latches error-free input data at the next clock edge. As a result, all error effects are masked in the circuit, as presented in Fig. 1. This masking effect is *temporal masking*, which depends on the occurrence timing of SEU and propagation delays between flip-flops. The probability of error propagation considering the temporal masking, *TMF*, is evaluated by the propagation delays between flip-flops [10], [20], [21]. *TMF* is the ratio of a vulnerable window (T_{error}) to a clock cycle (T_{cycle}) in Fig. 2. When an SEU occurs in the vulnerable window, the SEU can propagate to the downstream flip-flop temporally. In Fig. 2, t_{setup} and t_{hold} respectively denote the setup time and hold time of the downstream flip-flop. t_{prop} is the sum of the delays of the erroneous flip-flop and the combinational logic. α is the probability that the SEU which propagates to the downstream flip flop between setup and hold time is latched. α is a constant value from 0 to 1, which depends on reaching characteristics of the downstream flip-flop. For this study, α is set to 0.5 because we consider that the probability for latching the SEU decreases linearly from the start of the setup time to end of the hold time. TMF_{org} is calculated as shown below.

$$TMF_{org} = \frac{T_{error}}{T_{cycle}} = \frac{T_{cycle} - t_{setup} - t_{prop} + \alpha(t_{setup} + t_{hold})}{T_{cycle}} \tag{2}$$

Static timing analysis (STA) is used to evaluate the propagation delays.

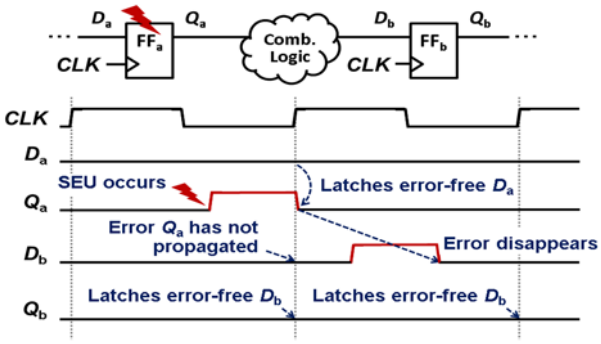

Fig. 1 Temporal masking.

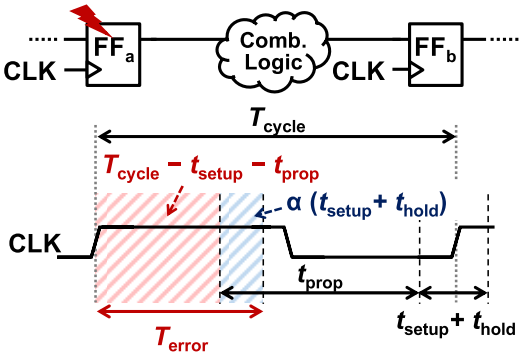

Fig. 2 Analytical technique for temporal masking.

2.2 Logical Masking

When an error propagates to a gate input with a value that does not affect the gate output logically (e.g. two-input AND gate when another input is "0"), the error is masked. This masking effect, *logical masking*, depends strongly on the input vector of the circuit. As described before, analytical techniques are practical for logical masking analysis [12]–[15].

An analysis technique using binary decision diagrams (BDD) analyzes the Boolean function of combinational logic and error propagation efficiently [12]. A technique proposed in an earlier report [13] evaluates the logical behavior of the circuit and the error propagation using a Probabilistic Transfer Matrix (PTM) and Ideal Transfer Matrix (ITM). In [14], error propagation considering logical masking is evaluated using error propagation rules with four-value logic. However, the accuracy of this technique is degraded because of re-convergent paths because the signal correlation of errors is not considered on re-convergent paths. Chen proposed the Correlation Coefficient Method (CCM) considering signal correlation [15]. In this technique, error propagation is modeled as a Boolean function. The model is implemented by combinational logic and is connected to the circuit under evaluation. The error propagation considering logical masking is evaluated by the signal probability of the circuit. The signal probability, the probability $p(x)$ that a signal x is "1",

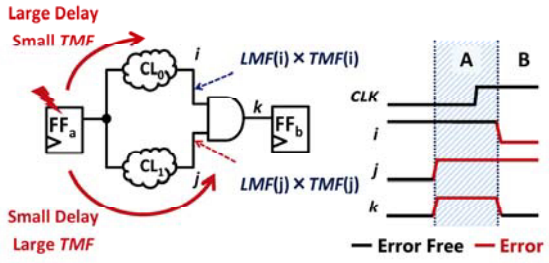

Fig. 3 Re-convergent path having two paths with different propagation delays.

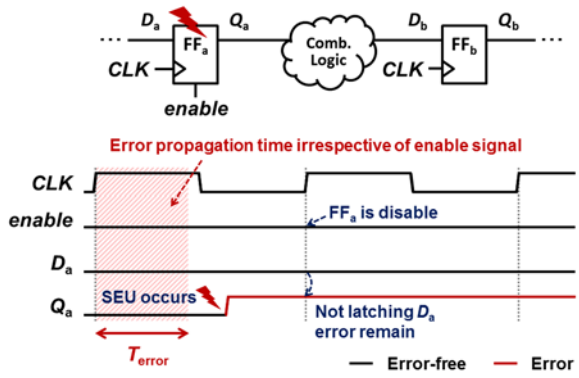

Fig. 4 Temporal masking disablement depending on the enabled state of the erroneous flip-flop.

is calculated using an earlier reported technique [16] that realizes fast and accurate error propagation analysis considering logical masking. As described in this paper, CCM is used for logical masking analysis, but the proposed technique is not restricted to the analysis technique.

2.3 Re-Convergent Path

A re-convergent path comprises a series of two or more paths with a common source register or gate (divergent site) and a common sink register or gate (convergent site) as shown on the left side of Fig. 3. Generally, the propagation delay differs among paths, so the timing by which an SEU reaches a convergent site differs for each path. Figure 3 shows a case in which the SEU propagates through CL_0 and CL_1 and converges at a two input AND gate. Assuming that the propagation delay of CL_0 is longer than that of CL_1, a time region A exists when an SEU reaches only node j because of the difference of propagation delay. Then, whether an SEU is masked or not is determined by the logical function and inputs of the gate. The SEU propagates to node k only in time region A in this case. When an SEU occurs at FF_a in a certain timing, the SEU might be sampled at FF_b in the next clock edge. To conduct accurate analyses of the masking effects for SEU propagation, a difference of propagation delay on the re-convergent path must be considered. However, conventional techniques do not consider masking effects on the re-convergent path [12]–[15].

3. Proposed Analysis Technique

3.1 Temporal Masking Disablement

Temporal masking can be disabled when the enable signal can control data latching of the erroneous flip-flop, as shown in Fig. 4. The disabled erroneous flip-flop does not latch error-free input data at the next clock edge; the error remains. The remaining error can propagate in the next clock cycle even if the error cannot propagate in the error occurrence cycle. In the next clock cycle, the error is affected only by logical masking without temporal masking because the error exists at the beginning of the cycle. Similarly, temporal masking can be disabled when the clock-gating scheme controls data latching of the erroneous flip-flop. Conventional individual analysis for temporal masking and logical masking cannot evaluate temporal masking disablement because the temporal and logical behaviors of the circuit influence the effect, thereby producing overly optimistic analysis results.

Temporal masking disablement can be evaluated by the probability that the erroneous flip-flop enables at clock edges, defined as $p(enable)$. To determine $p(enable)$ for a flip-flop, signal probability analysis without error injection is carried out in advance. Then, $p(enable)$ can be obtained from the log file of signal probability analysis because the enabled state depends on logical behavior of the circuit. The signal probability is calculated in the logical masking analysis. In the proposed analysis technique, *TMF* is compensated by $p(enable)$ considering that the error can be affected by temporal masking if the enable signal of the erroneous flip-flop is asserted and temporal masking is disabled if it is negated, as

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

where TMF_{comp} stands for the compensated error propagation probability considering the enable state of erroneous flip-flop and TMF_{org} represents the probability calculated using Eq. (2).

3.2 Masking Effects on Re-Convergent Path

The impact of masking effects on the re-convergent path depends on logical masking and temporal masking. Therefore, we propose a soft error propagation analysis technique that calculates *LMF* with consideration of temporal masking effects based on conventional logical masking factor analysis [15]. In the conventional technique, the *LMF* of output of a gate is calculated using *LMF* and the signal probability of inputs of the gate. However, the proposed technique uses the product of *LMF* and *TMF* of input of the gate instead of *LMF*. Multiplying *TMF* means that *LMF* is corrected by considering the effects of the difference of convergent timing, which are treated as simultaneous in logical masking analysis. For example, a re-convergent path exists that comprises a path with a small *TMF* and another path with large *TMF*. Small *TMF* means that the propagation delay is large.

Therefore, multiplying smaller *TMF* is equal to delay the convergent timing more. In this way, propagation probability is fixed according to propagation delay on each path and error convergent timing on re-convergent path is considered.

A) Method

The proposed technique requires multiplication of the node-proper *TMF* to calculate the error propagation probability at each node. Consequently, *TMF* between flip-flops, which is obtainable using Eq. (2), is divided into node-proper *TMFs*. The *TMF* division method is described later. The node-proper *TMF* is designated as the divided *TMF*. The method that calculates the *MF* of each node is presented in this subsection. In conventional techniques, the signal probability and *LMF* of each node are calculated successively from the primary inputs and/or outputs of flip-flops to primary outputs and/or inputs of flip-flops in order. Using the proposed technique, after evaluating $LMF(i)$, which is the *LMF* at node i, $MF(i)$ is calculable by multiplying $LMF(i)$ and $TMF(i)$. $MF(i)$ represents the probability that SEU propagates to node i without being masked logically or temporally. Then, *LMF* of the downstream gate output is calculated using *MF* calculated earlier. The calculation flow is presented in Fig. 5. $MF(n\_0)$ is calculated by multiplying $TMF(n\_0)$ and $LMF(n\_0)$,which is 1.0 when FF_a is a erroneous flip-flop. $LMF(n\_1)$ is calculated from the signal probability of the input of gate g_1 and $MF(n\_0)$ using the conventional method. Then, $MF(n\_1)$ is calculated by multiplying $LMF(n\_1)$ and $TMF(n\_1)$. These calculation flows are repeated for primary outputs and inputs of downstream flip-flops. Eventually, we can calculate the error propagation probability for a flip-flop considering temporal masking and logical masking on a re-convergent path.

B) Divided *TMF*

As described above, divided *TMF* is used to calculate *MF* at each node. Divided *TMF* of input signal of a gate g ($TMF(i)$) is the probability that a bit-flip occurring on node i propagates to the output of gate g_i by the next clock edge. Let n_0, n_1, ..., be nodes on the path between erroneous flip-flop and downstream flip-flop in order starting from the near side of the erroneous flip-flop. The divided *TMF* of output n_0 of the erroneous flip-flop is

$$\begin{cases} TMF\,(n\_0) = \frac{T_{base} - delay(FF_{SEU})}{T_{cycle}} \\ T_{base} = T_{cycle} - t_{setup} + \alpha\left(t_{setup} + t_{hold}\right) \end{cases} \tag{4}$$

where $delay(FF_{SEU})$ is the erroneous flip-flop delay, and T_{base} is *TMF* assuming no delay between FFs. Therefore, T_{base} is T_{error} when t_{prop} is 0. Subsequently, the divided *TMF* of node n_i on path between the erroneous flip-flop and downstream flip-flop is calculated using Eq. (5).

$$TMF\,(n\_i) = \frac{T_{base} - delay\,(FF_{SEU}) - delay\_out\,(n\_i)}{T_{base} - delay\,(FF_{SEU}) - delay\_in\,(n\_i)} \quad (i \geq 1) \tag{5}$$

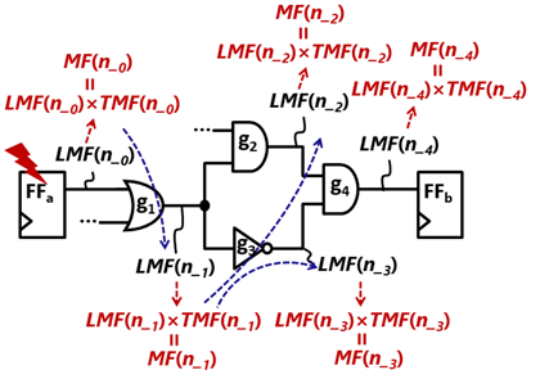

Fig. 5 Calculation flow using the proposed technique.

In that equation, $delay\_out(n\_i)$ is a propagation delay from the erroneous flip-flop to node n_i; $delay\_in(n\_i)$ is a propagation delay from erroneous flip-flop to input of gate with output n_i. Assuming wiring delay is 0, when the calculation is performed according to Eqs. (4) and (5), the product of all divided *TMFs* on the path corresponds to TMF_{org}. When the wiring delay is 0, $delay\_out(n\_1)$, $delay\_out(n\_2)$ and $delay\_in(n\_1)$ are equal to $delay\_in(n\_2)$, $delay\_in(n\_4)$ and 0, respectively in the case of Fig. 5. Thus in this case, the product of $TMF(n\_0)$, $TMF(n\_1)$, $TMF(n\_2)$ and $TMF(n\_4)$ corresponds to TMF_{org} as shown in Eq. (6) because sum of $delay\_out(n\_4)$ and $delay(FF_{SEU})$ is equal to propagation delay of the path between FF_a and FF_b, respectively.

$$\begin{aligned} &TMF\,(n\_0) \cdot TMF\,(n\_1) \cdot TMF\,(n\_2) \cdot TMF\,(n\_4) \\ &= \frac{T_{base} - (delay\,(FF_{SEU}) + delay\_out(n\_4))}{T_{cycle}} \\ &= \frac{T_{base} - t_{prop}}{T_{cycle}} = TMF_{org} \end{aligned} \tag{6}$$

The divided *TMF* should be compensated by Eq. (3) to consider temporal masking disablement. The $delay(FF_{SEU})$, $delay\_in(n\_i)$, and $delay\_out(n\_i)$ are obtainable using a static timing analyzer (STA).

4. Experiment

The proposed technique can assess soft error propagation probability accurately considering masking effects on a re-convergent path and temporal masking disablement. No conventional technique considers these effects.

4.1 Experimental Setup

The proposed technique is applied to sequential circuit benchmarks ITS 99 [17] and ISCAS 89 [18] for experimentation. The circuits are synthesized using the e-shuttle 65 nm CMOS library provided by Fujitsu [22]. In this evaluation, the circuit from the erroneous flip-flop to the downstream flip-flop is extracted from the benchmark circuit because it is difficult to accurately analyze sequential circuits using Monte Carlo simulation, as described in section I. Monte

Carlo simulation can evaluate the extracted circuit because it is substantially a combinational circuit. The *LMF* of each node is evaluated using the correlation coefficient method (CCM) [15] because the method can evaluate *LMF* rapidly and accurately. The signal probability of primary inputs is 0.5, except for the reset and enable signal. The *TMF* of each node is calculated using Eqs. (2), (3) and (4). We use Synopsys Design Compiler [19] as the Static Timing Analysis tool to calculate the propagation delay of gates and flip-flops. Monte Carlo simulation is done using gate-level circuit annotated delay information. The SEU is injected randomly in the unit of 1 ps in a cycle. We use Monte Carlo simulation with 1,000,000 test cases for each benchmark circuit while the input vector and SEU injection timing are varied.

4.2 Results

This section presents a comparison of the accuracy of error propagation probability as evaluated using the conventional technique and the proposed technique. The accuracy is evaluated using the *Absolute Error* (*AE*) and *Mean Absolute Error* (*MAE*) as shown below.

$$AE(i) = MF_{sim}(i) - MF(i) \tag{7}$$

$$MAE = \frac{1}{N}\sum_{i}|AE(i)| \tag{8}$$

Therein, N stands for the number of pairs of the erroneous flip-flop and the downstream flip-flop, $MF(i)$ represents the error propagation probability evaluated by the analytical techniques for pair i, and $MF_{sim}(i)$ denotes the probability evaluated using the Monte Carlo simulation for pair i. In the conventional technique, $MF(i)$ is performed by multiplying *LMF* obtained with CCM and TMF_{org} as shown in Eq. (1), only at the final node. In the proposed technique, on the other hand, multiplication of *LMF* and divided *TMF* is performed at each node.

Figure 6 presents experimentally obtained results without temporal masking disablement ((a) $p(enable) = 1.0$) and with $p(enable)$set to 1/3 [23] (b) using ITC 99 benchmark circuit b01. In Fig. 6, the x axis shows the names of the erroneous flip-flop and the downstream flip-flop. The y axis shows *AE*. The proposed technique improves the average AE by 47.3% and 78.9% when the respective $p(enable)$ are 1.0 and 1/3. In Fig. 6(a), *AE* of path between stato[0] and outp is unchanged because the path does not include a re-convergent path. When $p(enable)$ is 1.0, the *AE* of path between stato[0] and stato[0] is worse because of inaccuracy inherent in the logical masking analysis technique. *MAE* evaluation for ITC 99 and ISCAS 89 benchmark circuits is depicted in Fig. 7. Compared with the conventional technique, the proposed technique improves average *MAE* by 37.8% and 70.5%, respectively, when $p(enable)$ is 1.0 and $p(enable)$ is 1/3. The proposed technique improves *MAE* for all circuits. The conventional technique underestimates the propagation probability of flip-flops with low $p(enable)$ because the temporal masking disablement is not considered. Thus in particular, the proposed technique is superior to the conventional

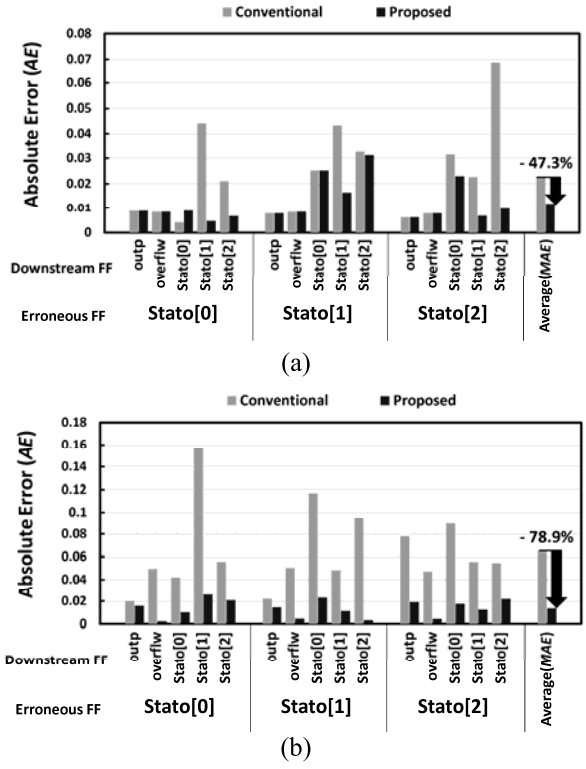

Fig. 6 Absolute Error of the experimentally obtained result for b01 using conventional and proposed techniques when (a) $p(enable)$ is 1 and (b) $p(enable)$ is 1/3.

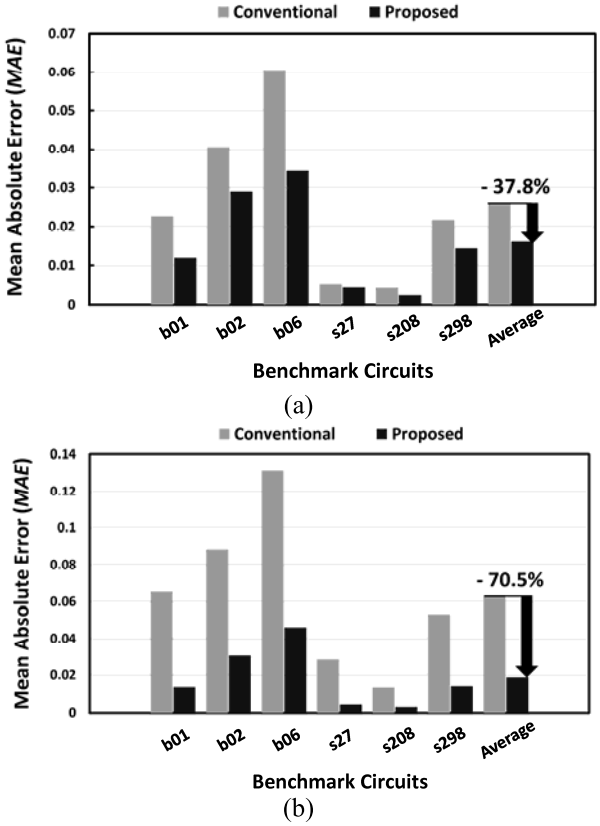

Fig. 7 Mean Absolute Error of the experimentally obtained result obtained using the conventional technique and the proposed technique when (a) $p(enable)$ is 1 and (b) $p(enable)$ is 1/3.

Table 1 Characteristics of benchmark circuits and runtime overhead from the proposed techniques.

Benchmark Circuits	*Analysis Technique*	*Evaluation Time [s]*	*Normalized Evaluation Time*	*# of Gates*	*# of FFs*
b01	Conventional	7.073	1.000	49	5
	Proposed	7.173	1.014		
b02	Conventional	6.966	1.000	28	4
	Proposed	7.030	1.009		
b06	Conventional	15.82	1.000	56	9
	Proposed	16.10	1.018		
s208	Conventional	39.82	1.000	112	8
	Proposed	40.56	1.019		
s298	Conventional	70.12	1.000	119	14
	Proposed	71.32	1.017		
Average	Conventional	27.96	1.000		
	Proposed	28.44	1.017		

technique with low $p(enable)$. This tendency particularly appears when erroneous flip-flop and downstream flip-flop are the same instance; *TMF* calculation needs not to be considered and thus logical masking can be cancelled out when the enable signal of the flip-flop is negated. The proposed technique requires more analysis time than the conventional technique to calculate *MF* using circuits annotated with divided *TMF*. We evaluate the runtime overhead for analysis. The results for each benchmark circuit are shown in Table 1. The maximum runtime overhead is 1.9% for s208. The average is 1.7%, so the runtime overhead is insignificant. The experimentally obtained results demonstrate that the proposed technique can perform more accurate analysis than conventional technique by considering masking effects on re-convergent path and temporal masking disablement with slight runtime overhead. Thereby, the proposed technique is useful for identifying flip-flops susceptible to soft error.

5. Conclusion

This paper presents a technique for accurate soft error propagation analysis technique for use in processor soft error rate evaluation. Conventional analysis techniques evaluate temporal and logical effects individually. They do not consider temporal masking and logical masking effects on the re-convergent path and temporal masking disablement. The proposed technique evaluates soft error propagation probability through combinational analysis of temporal and logical effects. Experimentally obtained results demonstrate that the proposed technique achieves 70.5% better accuracy of the error propagation probability, on average, than the conventional technique when the enabled probability of the technique can evaluate the soft error propagation more accurately and identify flip-flops that are susceptible to soft error.

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," Proc. 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," Proc. 2009 IEEE International Reliability Physics Symposium, pp.199–205, 2009.
[3] N.N. Mahatme, S. Jagannathan, T.D. Loveless, L.W. Massengill, B.L. Bhuva, S.-J. Wen, and R. Wong, "Comparison of combinational and sequential error rates for a deep submicron process," IEEE Trans. Nucl. Sci., vol.58, no.6, pp.2719–2725, 2011.
[4] M. Ebrahimi, A. Evans, M.B. Tahoori, R. Seyyedi, E. Costenaro, and D. Alexandrescu, "Comprehensive analysis of alpha and neutron particle-induced soft errors in an embedded processor at nanoscales," Design, Automation & Test in Europe Conference & Exhibition (DATE), pp.1–6, 2014.
[5] S. Mitra, N. Seifert, M. Zhang, Q. Shi, and K.S. Kim, "Robust system design with built-in soft-error resilience," Computer, vol.38, no.2, pp.43–52, 2005.
[6] W. Mansour and R. Velazco, "An automated SEU fault-injection method and tool for HDL-based designs," IEEE Trans. Nucl. Sci., vol.60, no.4, pp.2728–2733, 2013.
[7] L. Berrojo, I. Gonzalez, F. Corno, M.S. Reorda, G. Squillero, L. Entrena, and C. Lopez, "New techniques for speeding-up fault-injection campaigns," Proc. 2002 Design, Automation and Test in Europe Conference and Exhibition, pp.847–852, 2002.
[8] A. Mohammadi, M. Ebrahimi, A. Ejlali, and S.G. Miremadi, "SCFIT: A FPGA-based fault injection technique for SEU fault model," 2012 Design, Automation & Test in Europe Conference & Exhibition (DATE), pp.586–589, 2012.
[9] C. Lopez-Ongil, M. Garcia-Valderas, M. Portela-Garcia, and L. Entrena, "Autonomous fault emulation: A new FPGA-based ac-

celeration system for hardness evaluation," IEEE Trans. Nucl. Sci., vol.54, no.1, pp.252–261, 2007.
[10] D. Alexandrescu, Soft Errors in Modern Electronic Systems, Springer, 2011.
[11] A. Evans, S.J. Wen, and M. Nicolaidis, "Case study of SEU effects in a network processor," Proc. IEEE Silicon Errors in Logic — Sytem Effects, pp.1–6, 2012.
[12] N. Miskov-Zivanov and D. Marculescu, "Soft error rate analysis for sequential circuits," Proc. 2007 Design, Automation & Test in Europe Conference & Exhibition, pp.1–6, 2007.
[13] D.T. Franco, M.C. Vasconcelos, L. Naviner, and J.-F. Naviner, "Signal probability for reliability evaluation of logic circuits," Microelectron. Reliab., vol.48, no.8-9, pp.1586–1591, 2008.
[14] H. Asadi and M.B. Tahoori, "Soft error derating computation in sequential circuits," Proc. 2006 IEEE/ACM international conference on Computer-aided design, ICCAD'06, pp.497–501, 2006.
[15] L. Chen, M. Ebrahimi, and M.B. Tahoori, "CEP: Correlated error propagation for hierarchical soft error analysis," J. Electron. Test., vol.29, no.2, pp.143–158, 2013.
[16] S. Ercolani, M. Favalli, M. Damiani, P. Olivo, and B. Ricco, "Estimate of signal probability in combinational logic networks," Proc. 1st European Test Conference, pp.132–138, 1989.
[17] CAD Group, Politecnico de Torino, "ITC'99 Benchmarks" http://www.cad.polito.it/downloads/tools/itc99.html
[18] ISCAS89 Benchmark Circuits, https://filebox.ece.vt.edu/~mhsiao/iscas89.html
[19] Synopsys Design Compiler, http://www.synopsys.com/Tools/Implementation/RTLSynthesis/DesignCompiler/Pages/default.aspx
[20] N. Seifert and N. Tam, "Timing vulnerability factors of sequentials," IEEE Trans. Device Matel. Rel., vol.4, no.3, pp.516–522, 2004.
[21] H.T. Nguyen, Y. Yagil, N. Seifert, and M. Reitsma, "Chip-level soft error estimation method," IEEE Trans. Device Matel. Rel., vol.5, no.3, pp.365–381, 2005.
[22] Fujitsu e-Shuttle, http://www.fujitsu.com/jp/group/fsl/en/documents/resources/news/press-releases/2008/20081009e.pdf
[23] S. Wimer and I. Koren, "Design flow for flip-flop grouping in data-driven clock gating," IEEE Trans. VLSI Syst., vol.22, no.4, pp.771–778, 2014.

Go Matsukawa received a B.E and M.E degree in Computer and Systems Engineering from Kobe University, Hyogo, Japan in 2013 and 2014, respectively. He is currently working in the doctoral course at Kobe University. His research interests include fault tolerant processor designs, multi-core processor architecture and error propagation analysis. He is a student member of IEICE.

Yuta Kimi received a B.E degree and M.E in Computer and Systems Engineering from Kobe University, Hyogo, Japan in 2013 and 2015, respectively. He researched variation-tolerant processor designs and multi-core processor architecture.

Shuhei Yoshida received a B.E. degree and M.E. in computer and Systems Engineering from Kobe University, Hyogo, Japan in 2014 and 2016, respectively. He researched soft error propagation analysis and multi-core processor architecture.

Shintaro Izumi respectively received his B.Eng. and M.Eng. degrees in Computer Science and Systems Engineering from Kobe University, Hyogo, Japan, in 2007 and 2008. He received his Ph.D. degree in Engineering from Kobe University in 2011. He was a JSPS research fellow at Kobe University from 2009 to 2011. Since 2011, he has been an Assistant Professor in the Organization of Advanced Science and Technology at Kobe University. His current research interests include biomedical signal processing, communication protocols, low-power VLSI design, and sensor networks. He is a member of the IEEE, IEICE, and IPSJ.

Hiroshi Kawaguchi received B.Eng. and M.Eng. degrees in electronic engineering from Chiba University, Chiba, Japan, in 1991 and 1993, respectively, and earned a Ph.D. degree in electronic engineering from The University of Tokyo, Tokyo, Japan, in 2006. He joined Konami Corporation, Kobe, Japan, in 1993, where he developed arcade entertainment systems. He moved to The Institute of Industrial Science, The University of Tokyo, as a Technical Associate in 1996, and was appointed as a Research Associate in 2003. In 2005, he moved to Kobe University, Kobe, Japan. Since 2007, he has been an Associate Professor with The Department of Information Science at that university. He is also a Collaborative Researcher with The Institute of Industrial Science, The University of Tokyo. His current research interests include low-voltage SRAM, RF circuits, and ubiquitous sensor networks. Dr. Kawaguchi was a recipient of the IEEE ISSCC 2004 Takuo Sugano Outstanding Paper Award and the IEEE Kansai Section 2006 Gold Award. He has served as a Design and Implementation of Signal Processing Systems (DISPS) Technical Committee Member for IEEE Signal Processing Society, as a Program Committee Member for IEEE Custom Integrated Circuits Conference (CICC) and IEEE Symposium on Low-Power and High-Speed Chips (COOL Chips), and as an Associate Editor of IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences and IPSJ Transactions on System LSI Design Methodology (TSLDM). He is a member of the IEEE, ACM, IEICE, and IPSJ.

Masahiko Yoshimoto joined the LSI Laboratory, Mitsubishi Electric Corporation, Itami, Japan, in 1977. From 1978 to1983 he had been engaged in the design of NMOS and CMOS static RAM. Since 1984 he had been involved in the research and development of multimedia ULSI systems. He earned a Ph.D. degree in Electrical Engineering from Nagoya University, Nagoya, Japan in 1998. Since 2000, he had been a professor of Dept. of Electrical & Electronic System Engineering in Kanazawa University, Japan. Since 2004, he has been a professor of Dept. of Computer and Systems Engineering in Kobe University, Japan. His current activity is focused on the research and development of an ultra low power multimedia and ubiquitous media VLSI systems and a dependable SRAM circuit. He holds on 70 registered patents. He has served on the program committee of the IEEE International Solid State Circuit Conference from 1991 to 1993. Also he served as Guest Editor for special issues on Low-Power System LSI, IP and Related Technologies of IEICE Transactions in 2004. He was a chair of IEEE SSCS (Solid State Circuits Society) Kansai Chapter from 2009 to 2010. He is also a chair of The IEICE Electronics Society Technical Committee on Integrated Circuits and Devices from 2011–2012. He received the R&D100 awards from the R&D magazine for the development of the DISP and the development of the realtime MPEG2 video encoder chipset in 1990 and 1996, respectively. He also received 21th TELECOM System Technology Award in 2006.