Advanced Soft-Error-Rate (SER) Estimation with Striking-Time and Multi-Cycle Effects

Ryan H.-M. Huang, and Charles H.-P. Wen
Dep. of Electrical and Computer Engineering, National Chiao Tung University, Taiwan
E-mail: hmhuang.eed00g@g2.nctu.edu.tw and opwen@g2.nctu.edu.tw

ABSTRACT

Soft error rate (SER) has become a critical reliability issue for CMOS designs due to continuous technology scaling. However, the *striking-time* and *multi-cycle* effects have not been properly considered in SER for advanced CMOS designs. Therefore, in this paper, the *striking-time* and *multi-cycle* effects are formulated into the problem of SER estimation, and then a SER analysis framework is proposed, accordingly. Experimental results show that SERs on the benchmark circuits are seriously underestimated when ignoring both effects. Moreover, SERs increase more on those high-performance or low-power CMOS designs. New treatment to SER needs to be explored in the future.

Categories and Subject Descriptors

J.6 [**Computer Applications**]: Computer-Aided Engineering-Computer-aided design (CAD)

General Terms

Reliability, Algorithms

Keywords

Soft error, transient fault

1. INTRODUCTION

The continued scaling of integrated circuit (IC) technologies leads to more benefits for IC designs, such as smaller area and higher clock frequencies. However, more challenges on circuit reliability (especially, soft error rate [1][2]) also come up with the technology scaling. A soft error is a radiation-induced transient fault latched by a state-holding and will result in a system failure. With technology scaling, it precipitates higher operating frequency, shorter logic depth and smaller transistor-to-transistor spacing. All of these phenomena magnify the sensitivity of transient faults, leading to an exponential growth of soft error rates (SERs)
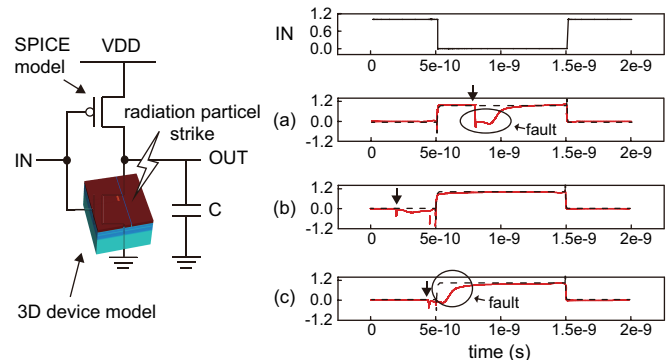
Figure 1: An example of *striking-time* effect on soft error estimation

in combinational circuits. In other words, soft errors greatly degrade the reliability of a system and can no longer be ignored in nanometer technologies, especially for the safety-critical (high reliability) applications such as automotive, aerospace, medical and etc. Note that due to the impact of soft error increases, Automotive Electronics Council also includes soft error tests for automotive LSI designs in AEC-Q100-Rev.G [3].

For this reason, characterizing soft errors and analyzing their behaviors in combinational circuits become indispensable for circuit reliability. Hence, several studies have been contributed to estimate SER correctly. Rao et al. [4] proposed an efficient framework to compute SER in combinational circuits using parameterized descriptors. SEAT-LA [5] modeled the propagation of transient faults and estimated SER with analytical equations and pre-characterized cell libraries. AnSER [6] proposed an efficient and accurate signature-based SER framework considering the logic-masking effect. SERA [7] combined several techniques, including graph theory, fault simulation, probability theory and circuit simulation, to evaluate SERs. Garg et al. [8] estimated the pulse width of transient fault for a gate based on their analytical model, which considers the ion-track establishment constant. In [9], the authors proposed a flow to compute SER by combining simulation-based and probabilistic method. MARS-C [10] and FASER [11] proposed highly-accurate SER estimation frameworks using symbolic techniques, including binary decision diagrams (BDDs) and algebraic decision diagrams (ADDs). A learning-based method was also developed to analyze the soft error under process variation [12].
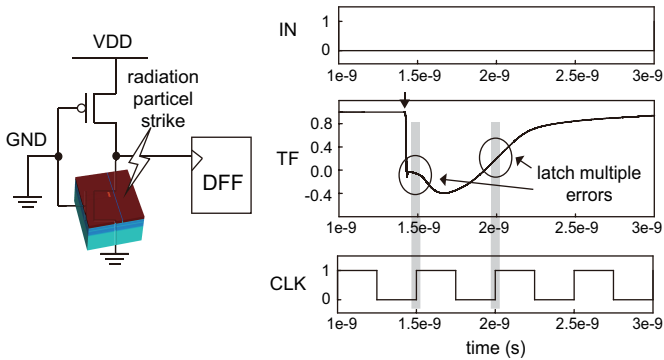
Figure 2: An example of *multi-cycle* effect on soft error estimation

However, all of these above works assume that a transient fault is generated when the particle strikes under a **passing** logic condition (i.e. the particle strikes at NMOS (PMOS) when the output is with logic-1 (logic-0)).

To take Fig. 1 as example, an inverter is described in 3D mixed-mode simulation where the NMOS transistor is modeled in 3D device domain and the PMOS transistor is modeled using SPICE model. In such example, a radiation particle strikes the drain region of a NMOS transistor. As a result, a negative transient is generated and results in a fault at the output. Hence, logically, a transient fault is assumed to occur as the output of the inverter is with logic-1. As shown in Fig. 1(a), the black dashed line represents the expected logic signal, the red solid line represents the signal with a transient fault and the arrow marks the striking time of a particle. However, a radiation-induced transient results from charge deposition and collection. Hence, a question rises, *"What will happen if the particle strikes under a **blocking** logic condition (i.e. the particle strikes NMOS (PMOS) when the output is with logic-0 (logic-1))?"*

The example in Fig. 1 is used again. As shown in Fig. 1(b), the striking time is set at 0.2ns, and the result shows no transient fault. However, a transient fault can be observed when the striking time is shifted from 0.2ns to 0.45ns as shown in Fig. 1(c). In other words, the result suggests that the transient fault will be generated if a particle strikes the device under a proper logic condition. Such *striking-time* effect should be considered during SER estimation.

Moreover, with continuous technology scaling, especially in nanometer technologies, operating frequency significantly increases to satisfy high performance demand. But, higher operating frequency is not a free lunch. Particularly, additional challenge for SER estimation arises - *a transient fault does not only result in a single-cycle error but also a multi-cycle error*. As shown in Fig. 2, the pulse width of generated transient fault is equal to 698ps, which means if the operating frequency is larger than 1.5GHz, an in-correct value will be captured in multiple cycles. As we can see, transient faults with larger pulse width are prone to causing multi-cycle errors. A similar issue was also reported by Dodd in [15], and states a possibility that a transient fault will span more than one cycle of circuit.

In addition, transient fault with large pulse width is not only generated from high-energy particles but also generated from an effect called *propagation-induced pulse broadening (PIPB)* [13]. The PIPB effect leads to an increase on the pulse width of a transient fault after propagating through a long chain of cells. For example, in [13], a generated transient fault with 200ps pulse width turns into nano-second order after propagation. To sum up, the *multi-cycle* effect should not be ignored during SER estimation.

Therefore, in this work, we propose an SER estimation framework considering both the *striking-time* and *multi-cycle* effects. Experimental result shows that SER is *underestimated* by 38% in average if both the *striking-time* and *multi-cycle* effects are ignored. Furthermore, the result also indicates the *multi-cycle* effect becomes more critical as the operating frequency increases. Finally, our observation demonstrates reliability issues also increase for low-power designs and thus suggests that the low-power optimization should be developed with more care.

The remainder of this paper is organized as follows: Section 2 introduces the background of transient faults and elaborates three masking mechanisms to prevent transient faults from becoming failures. Section 3 describes the proposed SER estimation framework considering both the *multi-cycle* and *striking-time* effects. Experimental results on IS-CAS'85 benchmark circuits, a series of multipliers as well as several industrial circuits are presented in Section 4. to show SER difference with and without both effects. Finally, Section 5 concludes this paper.

2. BACKGROUND

In this section, we will review the background information related to radiation particle strikes and transient faults. Basically, the background can be divided into two parts, **generation** and **propagation** of transient faults, described in Sec. 2.1 and Sec. 2.2, respectively.

2.1 Generation of Transient Faults

The generation of transient fault can be summarized as two steps, *charge deposition* and *charge collection*. As shown in Fig. 3, when a radiation particle strikes a node and passes through the device, it will generate electron-hole pairs along its passing path. This step is called *charge deposition* [14][15]. After the charge is deposited from such particle strike, the deposited charge will be *collected* by the charge-collection mechanism [14][15], including drift-diffusion, bipolar effect, and alpha-particle source-drain penetration (ALP-EN). As a result, the transient current will be generated at the drain node of struck device regardless of any charge collection mechanisms. In general, this transient current can be further modeled by an exponential current pulse at the circuit level [2][4][10], shown as follows:

$$I(t) = \frac{Q}{\tau}\sqrt{\frac{t}{\tau}}e^{-t/\tau} \tag{1}$$

where Q is the total amount of collected charges and τ is the time constant related to the process-related factors. More details about *charge deposition* and *charge collection* mechanisms can be referred to [1][14][15].

2.2 Propagation of Transient Faults

When a transient fault is induced by a high-energy charged particle, it may propagate to the primary output of the circuit and thus results in a soft error. However, not every generated transient fault can be latched by a memory element, three masking mechanisms [1][2][10] (Fig. 4) exist
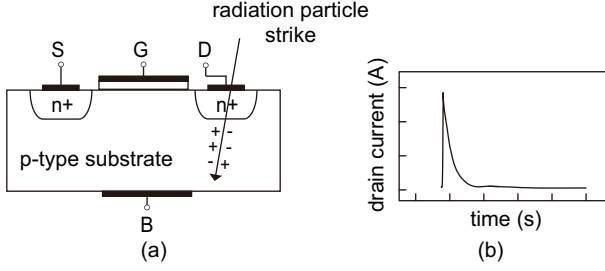
Figure 3: Generation and modelling of transient fault. (a) Interaction of a striking particle with a transistor (b) radiation-induced transient current
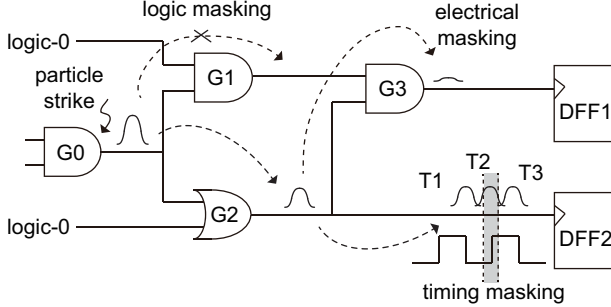

Figure 4: Three masking mechanisms for transient faults in combinational circuit

to prevent those transient faults from becoming soft errors. These three effects are:

2.2.1 electrical masking

when a transient fault propagates through a subsequent gate, it may be attenuated due to the electrical properties of propagated gates. This transient fault may disappear if the attenuation effect is strong enough. As illustrated in Fig. 4, the transient fault is masked because its amplitude is not high enough during propagating through G3.

2.2.2 logical masking

the logic masking occurs when there is no sensitizable path from the struck node to any output of the circuit. In Fig. 4, the transient fault is blocked after it propagates through G1 whose other side-input is with a controlling value (logic-0).

2.2.3 timing masking

because the flip-flop is insensitive to any arrival signal outside the latching window (i.e. setup time + hold time), the arrival transient fault will be masked since it falls outside the latching time or its pulse width is smaller then latching window. As shown in Fig. 4, only one transient fault, T2, will be captured because its arrival time falls within the latching window.

3. OUR SER ESTIMATION FRAMEWORK

In this section, we propose a SER analysis framework considering *striking-time* and *multi-cycle* effects. Fig. 5 shows the flowchart of overall SER estimation which consists of three stages: (1) *sensitized-probability computation*, (2) *generation and propagation of transient faults*, and (3) *total*
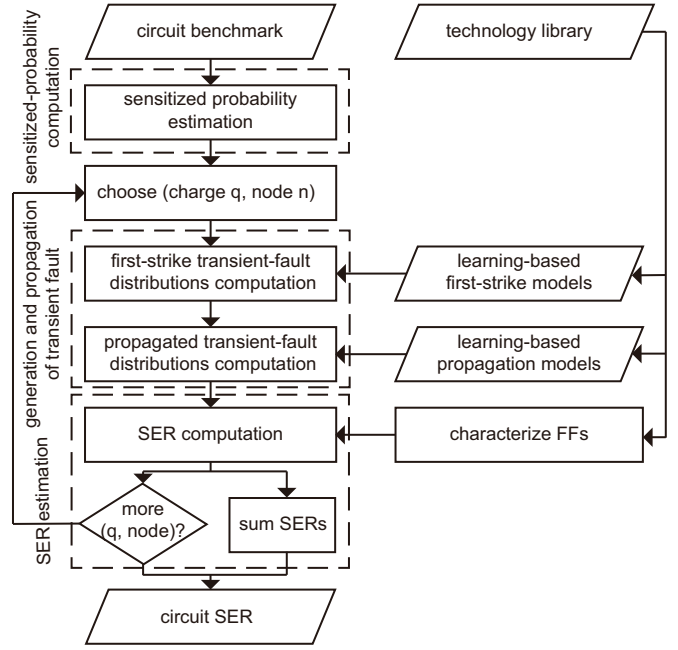

Figure 5: The flowchart of proposed framework

SER estimation. The following sections, starting reversely from *total SER estimation*, provide the details of each stage.

3.1 Total SER Estimation

First, we introduce the estimation of total SER for the circuit under test (CUT). The total SER can be computed as the summation of each individual node n in the circuit. That is,

$$SER_{CUT} = \sum_{n=1}^{N_{node}} SER_n \qquad (2)$$

where N_{node} is the total number of nodes susceptible to be struck by radiation particles in the CUT.

Each SER_n can be further formulated by integrating the products of frequency of particle-hit and the error probability over the range of charge Q_{MIN} to Q_{MAX}. Hence,

$$SER_n = \int_{q=Q_{MIN}}^{Q_{MAX}} F(q) \times P_{err}(n, q) \, dq \qquad (3)$$

where $F(q)$ represents the effective frequency of a particle hit in unit time could be found in [19]. $P_{err}(n, q)$ denotes the probability of a transient fault, induced by a particle with collected charge q strikes at node n, becomes a soft error after propagating to flip-flops.

In (3), the error probability $P_{err}(n, q)$ comprises generation and propagation of transient fault, including all three masking effects, as described in Sec. 2, and can be further formulated as:

$$P_{err}(n, q) = \sum_{i=1}^{N_{ff}} P_{sen}(n, i) \times \sum_{case=1}^{3} P_{gen}(n) \times P_{lat}(n, q, i) \qquad (4)$$

where N_{ff} and i indicates the total number of flip-flops and flip-flop i in CUT, respectively. $P_{sen}(\cdot)$ is the sensitized probability of a transient fault with respect to logical mask-

ing. $P_{gen}(\cdot)$ and $P_{lat}(\cdot)$ are generation probability and latching probability of a transient fault with respect to electrical masking and timing masking. Moreover, in order to incorporate the *striking-time* effect, the generation of a transient fault will be classified into three cases. More details about each component and three cases are elaborated in Sec. 3.2 and Sec. 3.3, respectively.

3.2   Sensitized-Probability Computation

In this section, we will discuss the computation of sensitized probability - $P_{sen}(\cdot)$. $P_{sen}(n, i)$ denotes the overall sensitized probability of transient faults propagating through the combinational logic from node n to flip-flop i along a path. This probability can be computed by the accumulated logic probability (P_{side}) for non-controlling values on all side-inputs along the path. The expression for $P_{sen}(n, i)$ is thus defined as:

$$P_{sen}(n, i) = \prod_{k \in n \to i} P_{side}(k) \tag{5}$$

where k represents one node on the path ($n \to i$). Note that in our framework the logic probability of each signal is computed by correlation coefficient method (CCM) [16], the most accurate approach for handling errors induced by reconvergent fanout nodes (RFONs) so far.

3.3   Generation- and Latching-Probability Computation

In (4), the latching probability $P_{lat}(n, q, i)$ reflects the electrical and timing masking effects when a transient fault induced by charge q propagates from node n to flip-flop i and is defined as:

$$
\begin{aligned}
P_{lat}(n, q, i) &= \int_{t=0}^{t_{clk}} \lambda_t(pw_{(t,i)}, w_i)\, dt \tag{6} \\
&= \int_{t=0}^{t_{clk}} \lambda_t(\lambda_e(n, q, i, t), w_i)\, dt \tag{7}
\end{aligned}
$$

In (7), t and t_{clk} denote the striking time and clock period, respectively. $pw_{(t,i)}$ indicates the pulse width of such transient fault struck at time t and latched by flip-flop i. w_i is the latching-window size of flip-flop i. λ_t and λ_e are timing- and electrical-masking functions, respectively. The timing-masking function λ_t is given by [2]:

$$
\lambda_t(pw, w) = \begin{cases} 0, & pw < w \\ \frac{pw - w}{t_{clk}}, & w \le pw \le t_{clk} + w \\ 1, & pw > t_{clk} + w \end{cases} \tag{8}
$$

However, in order to consider the *multi-cycle* effect, the λ_t should be modified into λ'_t defined as:

$$
\begin{aligned}
\lambda'_t(pw, w) &= N_{err} + \lambda_t(pw', w) \\
N_{err} &= \left\lfloor \frac{pw}{t_{clk}} \right\rfloor \\
pw' &= pw - N_{err} \cdot t_{clk} \tag{9}
\end{aligned}
$$

where N_{err} denotes the transient fault being latched in N cycles and pw' denotes the remaining pulse width after being latched in N cycles.

Next, the electrical-masking function λ_e in (7) indicates the change on the pulse width of a transient fault from generation, propagation and being latched, and can be formulated

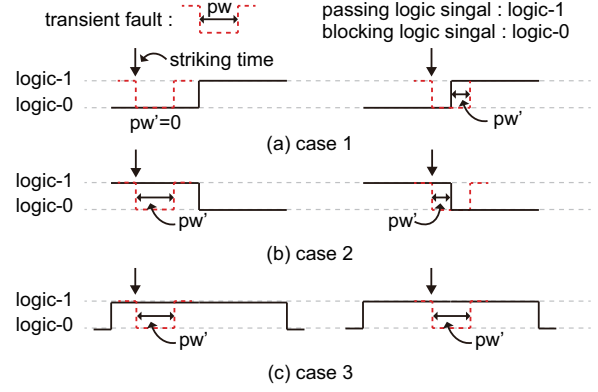**Figure 6: An example of classified three cases on NMOS transistor**

into:

$$
\begin{aligned}
\lambda_e(n, q, i, t) &= \\
&\underbrace{\psi_{prop}(\cdots(\psi_{prop}(\psi_{prop}(pw_0, 1), 2), \cdots), m)}_{m \text{ times}} \tag{10}
\end{aligned}
$$

where pw_0 is the generated pulse width induced by a particle with collected charge q strikes at node n can be obtained from function $\psi_{hit}(n, q)$. $\psi_{prop}(\cdot)$ is the propagation function used to estimate the behavior of a transient fault during propagation. Note that the $\psi_{hit}(\cdot)$ and $\psi_{prop}(\cdot)$ functions were implemented by learning-based method from [12] in our framework for efficiently and accurately considering the complicated process-variation impact on soft errors.

However, in order to consider the *striking-time* effect as illustrated in Fig. 1. We need to classify the generation of transient fault into three cases, including (1) current signal is *blocking* signal but next signal is *passing* signal, (2) current signal is *passing* signal but next signal is *blocking* signal, (3) continues *passing* signals. As shown in Fig. 6, a NMOS transistor is used to illustrate three cases where the *passing* logic condition is logic-1 and *blocking* logic condition is logic-0.

In case (1), in general, there should be no transient fault since a particle strikes under *blocking* logic condition (i.e. logic-0). But when the pulse width of a generated transient fault added with the striking time is larger than the pulse width of signal, the extra pulse width will result a transient fault as shown in Fig. 6(a). In case (2), in contrast, an intact transient fault will be propagated to the next level when a particle strikes under *passing* logic condition (i.e. logic-1). However, the extra pulse width will be masked when the pulse width added with the striking time is larger than the pulse width of signal, as shown in Fig. 6(b). In case (3), the transient fault can completely propagate to the next level because of the it is under continuous *passing* logic signal as shown in Fig. 6(c). It is worth noting that the *multi-cycle* effect only occurs under case (3) even though the pulse width is increased by the *PIPB* effect. The reason is that the extra pulse width will be masked when the next signal is under the *blocking* logic condition as shown in case (2).

To sum up, the pw_0 should be modified into pw_0' and

Table 1: Experimental results of various benchmark circuits

circuits	#gate	#PI	#PO	original SER(FIT)	only s.t. SER(FIT)	diff	only m.c SER(FIT)	diff	s.t. & m.c. SER(FIT)	diff	time(s)
c17	12	5	2	4.09E-05	5.40E-05	31.87%	4.54E-05	11.02%	5.85E-05	42.88%	0.54
c432	233	36	7	5.50E-04	6.61E-04	20.02%	6.23E-04	13.11%	7.33E-04	33.13%	1.77
c499	638	41	32	3.48E-04	4.96E-04	42.62%	3.86E-04	10.96%	5.34E-04	53.59%	9.38
c880	433	60	26	8.91E-04	1.14E-03	27.50%	1.00E-03	12.28%	1.24E-03	39.79%	1.79
c1355	629	41	33	9.80E-04	1.22E-03	24.46%	1.10E-03	12.15%	1.34E-03	36.61%	9.63
c1908	425	33	25	8.58E-04	1.06E-03	23.80%	9.58E-04	11.74%	1.16E-03	35.55%	4.29
c2670	872	157	64	7.75E-04	1.02E-03	32.03%	8.63E-04	11.36%	1.11E-03	43.39%	2.47
c3540	901	50	22	1.01E-03	1.32E-03	30.21%	1.13E-03	11.49%	1.44E-03	41.70%	9.29
c5315	1833	178	123	1.94E-03	3.05E-03	57.62%	2.11E-03	9.11%	3.23E-03	66.73%	8.59
c6288	2788	32	32	5.81E-04	8.22E-04	41.59%	6.44E-04	10.85%	8.85E-04	52.44%	120.5
c7552	2171	207	108	1.54E-03	2.02E-03	31.10%	1.71E-03	11.19%	2.19E-03	42.29%	13.2
mul_4	158	8	8	1.31E-04	1.68E-04	28.45%	1.46E-04	11.88%	1.83E-04	40.33%	1.04
mul_8	728	16	16	2.86E-04	3.67E-04	28.03%	3.19E-04	11.30%	3.99E-04	39.34%	8.4
mul_16	3156	32	32	6.49E-04	8.20E-04	26.41%	7.22E-04	11.28%	8.93E-04	37.68%	135.71
mul_24	7234	48	48	1.00E-03	1.26E-03	25.89%	1.12E-03	11.29%	1.38E-03	37.18%	685.8
mul_32	13017	64	64	1.36E-03	1.71E-03	25.64%	1.51E-03	11.29%	1.86E-03	36.93%	2213.9
CAN-bus ECU	26024	1774	1226	9.03E-01	1.09E-00	21.08%	1.01E-00	11.33%	1.20E-00	32.41%	23.2
bench1	110539	3975	3935	7.35E-01	8.43E-01	14.65%	8.22E-01	11.84%	9.30E-01	26.48%	116.1
bench2	242347	5705	5661	1.63E-00	1.88E-00	15.43%	1.82E-00	11.72%	2.07E-00	27.15%	282.3
bench3	49858	2429	2409	2.56E-01	3.02E-01	18.04%	2.86E-01	11.80%	3.32E-01	29.84%	44.06
bench4	899618	17871	17823	7.24E-00	8.11E-00	11.97%	8.10E-00	11.91%	8.97E-00	23.89%	1102.3
bench5	105334	4738	4718	6.98E-01	8.00E-01	14.56%	7.81E-01	11.92%	8.83E-01	26.48%	110
average						26.95%		11.49%		38.45%	

Algorithm 1 Overall SER estimation framework

1: select a particle with collected charge q
2: $Freq = GetFrequency(q)$; /*obtain from [19]*/
3: **for** each node n in benchmark **do**
4:   $P_{sen} = GetSignalProp(n)$; /*estimate by (5)*/
5:   $pw_0 = FirstHit(n, q)$; /*estimate by ψ_{hit}*/
6:   **for** each case $case$ **do**
7:     $pw_0' = GetPW(pw_0, t)$; /*estimate by (11)*/
8:     $pw = Propagate(pw_0')$; /*estimate by (10)*/
9:     $P_{lat} = GetLatProp(pw)$; /*estimate by (9)*/
10:    $P_{gen} = GetGenProp(n)$; /*estimate by (12)*/
11: **end for**
12: $SER_{CUT} = SER_{CUT} + Freq \times P_{sen} \times P_{gen} \times P_{lat}$;
13: **end for**
14: **return** SER_{CUT}

given by:

$$pw_0' = \begin{cases} 0, & pw_0 + t \leq t_{clk} \text{ in case (1)} \\ pw_0 + t - t_{clk}, & pw_0 + t > t_{clk} \text{ in case (1)} \\ pw_0, & pw_0 + t \leq t_{clk} \text{ in case (2)} \\ t_{clk} - t, & pw_0 + t > t_{clk} \text{ in case (2)} \\ pw_0, & \text{in case (3)} \end{cases} \quad (11)$$

Finally, based on these three cases, the generation probability of a transient fault in node n, $P_{gen}(n)$ in (4), can be formulated as:

$$P_{gen}(n) = \begin{cases} (1 - P_{pass}(n)) \times \alpha, & \text{in case (1)} \\ P_{pass}(n) \times \alpha, & \text{in case (2)} \\ P_{pass}(n) \times (1 - \alpha), & \text{in case (3)} \end{cases} \quad (12)$$

where $P_{pass}(n)$ is the passing logic probability for struck node n (i.e. logic-0 (logic-1) for PMOS (NMOS)) and α is the logic switching probability.

Algorithm 1 summarized the proposed framework. Each functions in this algorithm can be refereed to the equations described in this section. First, a particle with collected charge q is selected for SER estimation and this particle will be injected into each node n of CUT. Next, the sensitized probability P_{sen} of node n could be calculated by (5). Under each case, the generation and latching probability, P_{gen} and P_{lat}, are thus calculated by (9) to (12), respectively. Finally, the SER_{CUT} can be computed by particle frequency, P_{gen}, P_{lat} and P_{sen}.

4. EXPERIMENTAL RESULT

This section performs the proposed framework considering the *striking-time* and *multi-cycle* effects to estimate SERs of the CUT. The proposed framework was implemented in C/C++ on a Linux machine with an Intel Core i7 processor and 16GB RAM. The technology used was 45nm Nangate Open Cell Library [17] and the clock frequency was set as 2GHz. Experiments were conducted on a set of standard benchmarks from ISCAS85, a series of multipliers (mul_4 to mul_32), a CAN-bus ECU design for automotive electronics and five industrial benchmark circuits (bench1 to bench5) from the Industrial Technology Research Institute of Taiwan (ITRI) [18]. The switching activity of each node is set independent.

Table 1 lists SER results and information of each benchmark circuits. Columns 1 to 4 show the information about each benchmark circuits, including the name of each circuit (circuits), the number of nodes (#node), the number of primary inputs (#PI), and the number of primary outputs (#PO), respectively. Next column (original) indicates SER estimation without incorporating the *striking-time* (denoted by s.t.) and *multi-cycle*(denoted by m.c.) effects, respectively. The other conditions for SER computation and difference between the result without s.t and m.c. are listed in the following six columns. The last column (time) shows the required run times of our proposed framework.

Compared with the SER estimation with and without s.t. and m.c., respectively, the SER differences are 26.95% and 11.49% in average. As we incorporte both s.t. and m.c., the discrepancy further goes up to 38.45% in average. This result, again, proves that both s.t. and m.c. effects are necessary for SER estimation; otherwise, the result will be
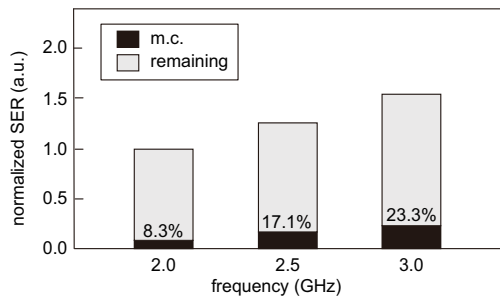
Figure 7: Impact of increasing operating frequency

Figure 8: Impact of switching activity α

greatly *underestimated.* Moreover, all runtime from our proposed framework are less than 1 hour, even for the largest circuit($\sim$ million gate counts), demonstrating the efficiency of our framework to be scaled.

Next, the impact of increasing operating frequency for SER estimation is shown in Fig. 7. Here SERs in each frequency are normalized by SER in 2 GHz. m.c. represents the part from the *multi-cycle* effect in total SER. The result shows that the m.c. in total SER grows from 8.3% to 23.3% as the frequency rises from 2 GHz to 3 GHz. In other words, the m.c. becomes more critical for SER analysis when the operating frequency increases to fulfill the demand of high performance.

Moreover, the switching activities are not independent among all nodes in the cirucit. Hence, the impact of switching activity for SER estimation is illustrated in Fig. 8, where the SERs are all normalized by the SER when switching activity is 0.1. As we can see, the normalized SERs decrease as the switching activity of each node goes larger. It means that the largest SERs may occur when the circuit operates at the low-power mode. This observation also suggests that the low-power algorithm should be designed more carefully to prevent the reliability problem.

5. CONCLUSION

As the IC technology continues scaling, the reliability issue (especially on soft error rate) becomes increasingly critical for CMOS designs. In this work, the *striking-time* and *multi-cycle* effects on SER estimation have been thoroughly investigated. Thus, an accurate and efficient SER analysis framework considering both effects can be proposed. Experimental results demonstrate that both *striking-time* and *multi-cycle* effects need to be considered in SER analysis for avoiding underestimation of SER. Particularly, high-performance or low-power designs may cause rapid increase on SER and thus need different treatment in the future. Other directions related to radiation-induced soft errors include (1) applying hardening techniques for design optimization and (2) developing the SER-aware low-power algorithm.

6. REFERENCES

[1] V. Ferlet-Cavrois, L. W. Massengill, and P. Gouker, "Single Event Transients in Digital CMOS - A Review," *IEEE Trans. Nuclear Science,* vol. 60, no. 3, pp. 1767-1790, Jun. 2013.

[2] 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. Intâ̆Źl Conf. Dependable Systems and Networks,* pp. 389-398, 2002.

[3] Automotive Electronics Council, "Failure Mechanism Based Stress Test Qualiïïn̆Acation for Integrated Circuits," AEC-Q100-Rev.G, May 14, 2007.
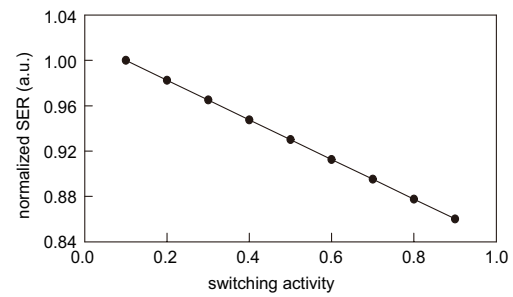
[4] R. R. Rao, K. Chopra, D. T. Blaauw, and D. M. Sylvester, "Computing the Soft Error Rate of a Combinational Logic Circuit Using Parameterized Descriptors," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems,* vol. 26, no. 3, pp. 468-479, Mar. 2007.

[5] R. Rajaraman, J. S. Kim, N. Vijaykrishnan, Y. Xie, and M. J. Irwin, "SEAT-LA: A Soft Error Ananlysis tool for Combinational Logic," *Proc. Intâ̆Źl Conf. VLSI Design,* pp. 499-502, 2006.

[6] S. Krishnaswamy, S. M. Plaza, I. L. Markov, and J. P. Hayes, "Signature-Based SER Analysis and Design of Logic Circuits," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems,* vol. 28, no. 1, pp. 74-86, Jan. 2009.

[7] M. Zhang and N. Shanbhag, "A Soft Error Rate Analysis (SERA) Methodology," *Proc. Intâ̆Źl Conf. Computer Aided Design,* pp. 111-118, 2004.

[8] R. Garg, C. Nagpal, and S. P. Khatri, "A fast, analytical estimator for the SEU-induced pulse width in combinational designs," *Proc. Design Automation Conf.,* pp. 918-923, 2008.

[9] D. Alexandrescu, E. Costenaro, and M. Nicolaidis, "A Practical Approach to Single Event Transients Analysis for Highly Complex Designs," *Proc. Intâ̆Źl Smyp. Defect and Fault Tolerance in VLSI and Nanotechnology Systems,* pp. 155-163, 2011.

[10] N. Miskov-Zivanov and D. Marculescu, "Mars-C: Modeling and Reduction of Soft Errors in Combinational Circuits," *Proc. Design Automation Conf.,* pp. 767-772, 2006.

[11] B. Zhang, W.-S. Wang, and M. Orshansky, "Faser: Fast Analysis of Soft Error Susceptibility for Cell-based Designs," *Proc. Intl Smyp. Quality Electronic Design,* pp. 755-760, 2006.

[12] H.-M. Huang and C.H.-P. Wen, "Fast-Yet-Accurate Statistical Soft-Error-Rate Analysis Considering Full-Spectrum Charge Collection," IEEE Design & Test, vol. 30, no. 2, Apr. 2013, pp. 77-86.

[13] V. Ferlet-Cavrois, P. Paillet, D. McMorrow, N. Fel, J. Baggio, S. Girard, O. Duhamel, J.S. Melinger, M. Gaillardin, J. R. Schwank, P. E. Dodd, M. R. Shaneyfelt, and J. A. Felix, "New Insights Into Single Event Transient Propagation in Chains of Inverters-Evidence for Propagation-Induced Pulse Broadening," *IEEE Trans. Nuclear Science,* vol. 54, no. 6, pp. 2338-2346, Dec. 2007.

[14] P. E. Dodd and L. W. Massengill, "Basic mechanisms and modeling of single-event upset in digital microelectronics," *IEEE Trans. Nuclear Science,* vol. 50, no. 3, pp. 583-602, Jun. 2003.

[15] P. E. Dodd, M. R. Shaneyfelt, J. A. Felix and J. R. Schwank, "Production and propagation of single-event transients in high-speed digital logic ICs," *IEEE Trans. Nuclear Science,* vol. 51, no. 6, pp. 3278-3284, Dec. 2004.

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

[17] Nangate 45nm Open Library, Nangate Inc., http://www.nangate.com/, 2009.

[18] "Industrial Technology Research Institute," http://www.itri.org.tw/chi/

[19] JEDEC JESD89. Measurement and Reporting of Alpha Particles an Terrestrial Cosmic Ray-Induced Soft Errors in Semiconductor Devices, Joint Electron Device Engineering Council, Soild State Technology Association, Aug. 2001.