Single-Event Upset Analysis and Protection in High Speed Circuits

Single-Event Upset Analysis and Protection in High Speed Circuits

Authors: Hosseinabady M., Lofti-Kamran P., Di Natale G., Di Carlo S., Benso A., Prinetto P.,

Published in the Proceedings of the IEEE 11th European Test Symposium (ETS), 21-24 May 2006, SouthAmpton, UK.

DOI: 10.1109/ETS.2006.41

Single-Event Upset Analysis and Protection in High Speed Circuits

Mohammad Hosseinabady†, Pejman Lotfi-Kamran†, Giorgio Di Natale‡, Stefano Di Carlo‡, Alfredo Benso‡, Paolo Prinetto‡
† University of Tehran, Iran. ‡Politecnico di Torino, Italy.
{Mohammad, plotfi}@cad.ece.ut.ac.ir, {giorgio.dinatale, stefano.dicarlo, alfredo.benso , Paolo.Prinetto}@polito.it
Politecnico di Torino, Dip. di Automatica e Informatica, Corso Duca degli Abruzzi 24, I-10129 Torino TO, Italy
Phone: +39-011-564.7080, Fax:+39-011-564-7099

Abstract: The effect of Single-Event Transients (SETs) (at a combinational node of a design) on the system reliability is becoming a big concern for ICs manufactured using advanced technologies. An SET at a node of combinational part may cause a transient pulse at the input of a flip-flop and consequently is latched in the flip-flop and generates a soft-error. When an SET conjoined with a transition at a node along a critical path of the combinational part of a design, a transient delay fault may occur at the input of a flip-flop. On the other hand, increasing pipeline depth and using low power techniques such as multi-level power supply, and multi-threshold transistor convert almost all paths in a circuit to critical ones. Thus, studying the behavior of the SET in these kinds of circuits needs a special attention. This paper studies the dynamic behavior of a circuit with massive critical paths in the presence of an SET. We also propose a novel flip-flop architecture to mitigate the effects of such SETs in combinational circuits. Furthermore, the proposed architecture can tolerant a Single Event Upset (SEU) caused by particle strike on the internal nodes of the flip-flop.

1. Introduction

Radiation-induced soft errors pose a major challenge to the design of memories and logic circuits in nanometer technologies. Neutron radiations from cosmic rays or alpha particles from packaging materials are common causes of soft errors in the nodes of a circuit. These radiations generate concentrated bursts of excess charges at random locations in a semiconductor substrate. These charges may be collected by a p-n junction resulting in a current pulse of very short duration in the signal value, usually termed *Single-Event Upset* (SEU). An SEU occurs in the hold state of a memory cell or in a flip-flop and causes a *soft error* when the content of the storage element is flipped. Furthermore, an SEU may occur in an internal node of a combinational circuit and subsequently be propagated to a storage element and be latched there. In this case it is usually called *Single Event Transient* (SET). Combinational circuits have a natural barrier against the propagation of SETs to their outputs. When an SET occurs at an internal node of a logic circuit, there are three masking factors that have impact on the SET [1].

1. **Logical masking**: If an SET reaches an input of a NAND (NOR) gate, but one of the other inputs is in the controlling state (*0* for NAND and *1* for NOR gates), the SET will be completely masked and the output will be unchanged. In other words, this SET will not cause a soft error. In order to causing a soft error, it is necessary to have a sensitized path from the particle strike node to the input of a latch.
2. **Temporal masking**: As an SET propagates towards a sequential element, e.g., a latch, the noise on a node of the combinational circuit may be outside the latching window of all the latches in the subsequent combinational paths. Hence, the error will not be latched, and there will be no soft error. This is called temporal masking.
3. **Electrical masking**: since all CMOS circuits have limited bandwidth, transients with bandwidth higher than the cutoff frequency will be attenuated. The pulse amplitude may reduce, the rise and fall time increase, and, eventually, the pulse may be filtered out completely.

In spite of these three masking mechanisms, an SET with enough amplitude may appear in the sampling window of a flip-flop in the circuit and can be latched in the flip-flop. To eliminate erroneous results due to this erroneously latched data, latches should protect themselves against these errors. As process technology scales below 100 nanometers, studies indicate high-density, low-cost, high-performance integrated circuits, characterized by high operating frequencies, low voltage levels, and small noise margins will be increasingly susceptible to SETs and this will result in unacceptable soft error failure rates even in mainstream commercial applications [1], [2].

Several researches study the soft-error caused by particle strike in the combinational and sequential parts of a circuit. Some works propose algorithms to estimate circuit vulnerability to an SEU/SET whereas, the other work propose device and circuit techniques to protect circuits against the SEU/SET.

Mohanram [3] proposes a comprehensive technique for simulation of transients caused by SETs in combinational logic circuits. Based upon linear RC models of gates, the proposed technique integrates a closed-form model for computation of the SET-induced transient at the site of a particle strike with propagation models for the transients along a functionally sensitized path. Gill, et al. [4] introduce an approach for computing soft error susceptibility of nodes in large CMOS circuits at the transistor level. The developed technique computes the electrical masking of nodes using characterization tables for every logic cell of the library using Spice simulations for a 100nm process technology. They also describe a technique to compute the logic masking of the transistor nodes using an automatic test pattern generation tool. Zhao, et al. [7] propose a noise impact analysis methodology based on a Noise Probability Density Function (NPDF) transformation technique to evaluate the circuit vulnerability to SEU.

Naseer, et al. [8] describe the delay filtered dual interlocked storage cell which immune to single event transients on any input

and single event upsets within the storage cell. Krishnamohan, et al., [5] propose an error-masking design technique for static CMOS combinational circuits that exploits the inherent temporal redundancy (timing slack) of logic signals to increase soft-error robustness. Because logic signals on the critical paths do not have a reasonable timing slack, this method is not applicable to latches in critical paths of a circuit whose behavior in the event of an SET has a great impact on the functionality of the circuit. Zhang, et al., [6] propose a technique to mitigate the deleterious effects of SETs. This technique combines a dual-sampling flip-flop (DSFF) and skewed CMOS (SCMOS) combinational circuit to mitigate the impact of SETs. The DSFF eliminates any 1-0-1 SETs and the SCMOS can be tuned to eliminate 0-1-0 SETs.

Almost all of these works study the SET and its effects in steady state voltage levels. However, dynamic behavior of a signal in the presence of an SET should be studied. When an SET is conjoined with a transition (dynamic behavior) on a value of a node along a critical path of the combinational part of a design, a *transient delay fault* may occur at the input of a flip-flop. In the high speed circuit in which the SET pulse width is comparable with clock period, this situation will be worse.

This paper studies the dynamic behavior of signals in a circuit with massive critical paths in the presence of an SET. Note that, examining the histogram of the critical-path delays for a typical digital block reveals that only a few paths are critical or near critical and that many path have much shorter delays [10]. But, using some high speed and low power techniques increases the number of critical paths in the circuit. The pipeline depth is increasing to 15 or 20 in order to accommodate the speed increase. Today 10 levels of logic in the critical path is more common and this number is expected to be decreasing further [11]. This decreasing numbers of gates in the pipeline stages results in an increasing number of critical paths in the circuit. On the other hand, using multiple voltage supply [10], Dynamic Voltage Scaling (DVS) [12], and multiple threshold voltage transistor [10], some of the major low power techniques, convert almost all paths in the combinational part of the circuit to critical ones. When a transition at the internal node along a critical path synchronizes with an SET caused by particle strike, a transient delay fault may be generated. When this transient delay fault appears at the input of a storage cell, it can be latched in the storage element as a soft error.

In this paper, we study the effect of SET in the critical paths of a circuit. We show that a particle strike at a node on a critical path may appear as an erroneous value at the input of a flip-flop in two shapes: a *transient pulse voltage*, or a *transient delay fault*. It should be noted that, electrical masking mechanisms which can attenuate a transient pulse has a very low effect on transient delay. Furthermore, we propose a new flip-flop architecture based on the clock gating techniques to detect and correct the SET and SEU in a circuit.

The rest of this paper is organized as follows. The next section introduces the transient fault model that is used in this paper. Section 3 describes the effect of an SET on voltage level of signals. The effect of an SET on critical path is explained in Section 4. Section 5 explains a protection mechanism and new flip-flop architecture to detect and correct the SET and SEU in a digital circuit. Section 6 demonstrates experimental results. Finally, Section 7 points to some future work and conclusions are appeared in the last section.

2. Transient Fault Model

When high-energy neutrons (presented in terrestrial cosmic radiations) or alpha particles (that originated from impurities in the packaging materials) strike a sensitive node in the CMOS circuit, they generate a dense local track of electron-hole pairs in the substrate (Figure 1-I). In the case of CMOS circuits, a sensitive node in the semiconductor is the drain of the OFF-transistors [4].

In presence of an electric field (e.g., depleted junction), electron-hole pairs are separated by drift and a "funnel" shaped potential distortion is generated. Then, this additional charge is collected and a current spike appears. Finally, the funnel is collapsed and a diffusion effect occurs. This phenomena is usually represented with a triangular or a double-exponential current spike [9] (Figure 1-II).

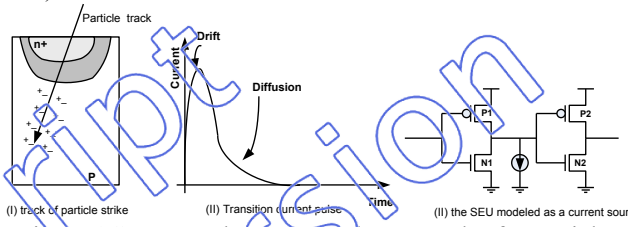

Figure 1Current pulse generated as a result of a particle strike

The current spike can be represented at the device level by a current source (Figure 1-III). Messenger [9] models this transient current as a double exponential injection current

$$I_{inj}(t) = \frac{Q}{(\tau_1 - \tau_2)}\left(e^{-\frac{t}{\tau_1}} - e^{-\frac{t}{\tau_2}}\right)$$

Where Q is the charge (positive or negative) deposited as a result of the particle strike, τ_1 is the collection time-constant of the junction, and τ_2 is the ion-track establishment time-constant. In the rest of the paper, we will use this current model.

In this paper, using the piecewise linear capability of modeling signals in HSPICE, we model the transient pulse current with a piecewise linear signal to generate some experimental results. Karnik, et al. [1] show that an SEU lasts about 100ps for 0.6um technology. In this paper, the maximum width of this transient current pulse is shown by τ_{max}.

Depending on the signals value along the propagation path from the upset node of a flip-flop (Figure 2), the generated pulse current due to particle strike may be manifested in different shapes in signals voltage and propagate to the data input of a flip-flop. The next section discusses this phenomenon.

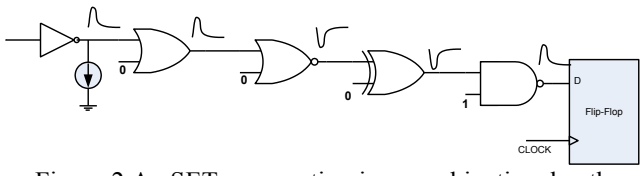

Figure 2 An SET propagation in a combinational path

3. Propagating an SET along a path

Consider a 2-input NAND gate. The effect of a particle strike on a NAND gate is shown in Figure 3. When inputs A and B are at logic values '1' and '0', respectively, transistors *P1* and *N2* are in their OFF-state, so their drains (i.e., nodes *p* and *n*) are susceptible to a particle trick. The current source I_{inj} of Figure 3-II models the effect of the particle that strikes the sensitive node *n*.

Figure 3-III and 3-IV show two different effects on the output voltage. If the two inputs *A* and *B* are stable at logic values '1' and '0', respectively, then a transient pulse will appear on the output node. If the input *B* changes during the particle strike, then an *early edge* will occur at the output node. An early edge may cause a soft-error in the downstream storage cells that are on a shortest path with propagation delay less than τ_{max}. Thus, if the propagation delay of the shortest path is greater than τ_{max} the early edge cannot generate a soft error.

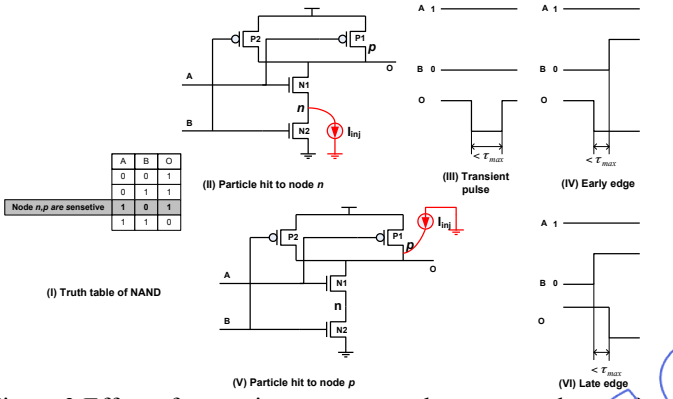

Figure 3 Effect of a transient current on the output voltage of a NAND gate

Figure 3-V and 3-VI show the case of a particle that hits the sensitive node *p*. In this case, extra charges in the node *p* can increase the delay of the NAND gate while the input *B* changes during the particle strike (i.e., a *late edge* is occurred). If such a delay occurs on a critical path of the design, it may cause a soft-error in the circuit. A late edge is also called *transition delay*.

On the other hand, a transient pulse caused by a particle strike may be changed when it propagates along a path in the circuit. A propagating transient pulse along a path may be *masked*, *attenuated, propagated*, converted to an *early edge* or a *late edge,* or even converted to a *dynamic hazard*. Figure 4 shows the five different effects of a transient pulse voltage generated at a node along its propagation path.

If a transient pulse reaches an input of a gate (e.g., a 2-input OR gate), but the other input is in the controlling state (e.g., 1 for OR), the transient pulse will be completely masked and the output will be unchanged. Therefore, this SET will not cause a soft error (Figure 4-I). If a transient pulse reaches an input of a gate (e.g., a 2-input OR gate), but the other input is in the non-controlling state (e.g., 0 for OR gate), because of the bandwidth limitation of the gate, an attenuated transient pulse will appear at the output of the gate (Figure 4-II). If a transient pulse reaches an input of a gate (e.g., OR), while the other input has a transition, the transient pulse may be attenuated (Figure 4-III), converted to an early edge (Figure 4-IV), converted to a late edge (i.e., delay, Figure 4-V), or converted to a dynamic hazard (Figure 4-VI). Dynamic hazard conversion attenuates the transient pulse width and increase the chance of the electrical masking. Thus, we do not consider this effect in the rest of the paper.

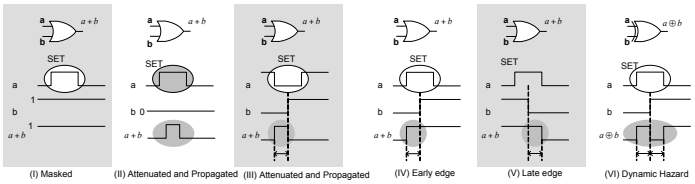

Figure 4 SET propagation mechanisms

The effect of propagated transient pulse has been studied in several works [3][4]. The early and late edge effects may have erroneous effects on shortest and critical paths of the circuit, respectively. And example of these effects is a delay fault, which will be analized in the next sessions.

4. Transient delay sensitive paths

In this section, we determine the paths in which early and late edges may lead to a soft error. First, we define some terminologies that are useful to determine these sensitive paths.

Definition 1: a Sampling window (t_{sw}) is the time that is bounded by the setup time (t_{up}) and hold time (t_h) around the active clock edge of a flip-flop (Figure 5-I).

Lemma 1: an SET results in a soft error if it appears in the sampling window of a flip-flop.

Definition 2: an Early edge sensitive path is a path in which an early edge caused by an SET may results in a soft error.

Definition 3: a Transient delay (late edge) sensitive path is a path in which a transient delay may lead to a soft error. Otherwise, the path is called transient delay insensitive. In other words, a transient delay never causes a soft error in a transient delay insensitive path.

Definition 4: the SET-setup time (t_{SETs}) is the time that the data input of a storage cell must be valid before the sampling window so that any transient delay (late edge) on the input of the storage cell cannot be latched in the storage cell (Figure 5-I).

Lemma 2: the SET-setup time is equal to τ_{max} (the maximum width of SET).

Definition 5: the SET-hold time (t_{SETh}) is the time that the data input of a storage cell must remain stable after the sampling window so that any early edge on the input of the storage cell cannot be latched in the storage cell (Figure 5).

Lemma 3: A path with propagation delay less than $\tau_{max} + t_h$ is an early edge sensitive path.

Lemma 4: A path is transient delay sensitive if its propagation delay (t_d) is greater than $T - (t_{SETs} + t_{su})$, where *T* is the period of the clock (i.e., $t_d > T - (t_{SETs} + t_{su})$).

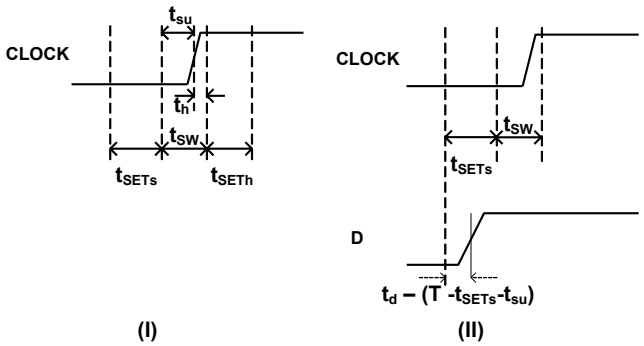

Figure 5 SET-setup time and SET-hold time

In the next section, we propose protection mechanisms and corresponding flip-flop architectures to detect and correct the transient pulse, early edge, and late edge (transient delay) that lead to a soft error in the flip-flop.

5. Protection Mechanism

To protect a circuit against the erroneous early edge, it is enough to increase the propagation of the shortest path of the circuit to $\tau_{max} + t_h$. This minimum-path delay can be realized by adding buffers to shortest paths during logic synthesis. Therefore,

this process introduces a certain amount of power and area overhead. However, in some design methodologies multi-level voltage supply or multi threshold voltage logic can be used to guaranty this minimum path delay such that the power consumption decreases.

In the sequel, we will analyze a sampling mechanism to protect a flip-flop against a transient pulse and transient delay in the combinational parts of the circuit. For this purpose, we first investigate the possible faulty signals at the data input of the flip-flop that may cause a soft error.

Figure 6 shows all the possible faulty signals at the input of a flip-flop that can create an erroneous data in the flip-flop. Signals of Figure 6-a and -b may occur at the input of all types of flip-flops in the design (those at the end of a transient delay sensitive path and those at the end of other paths). Signals of Figure 6-c and -d may occur only at the input of flip-flops that are at the end of a transient delay sensitive path.

A protection mechanism should detect these erroneous signals and correct the latching value in the flip-flops with minimum area, time, and power overhead on the normal operation of the circuit.

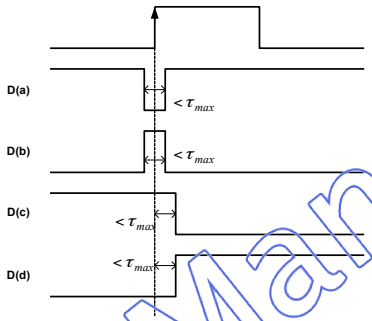

Figure 6 Possible erroneous signals caused by SET at the input of a flip-flop

In the sequel, we propose two SET-tolerant flip-flops for transient delay sensitive paths and transient delay insensitive paths of the design. The proposed architectures detect and correct the transient pulse and transient delay fault at the input of the flip-flop. Furthermore, for completeness of the protection the proposed architectures can also protect flip-flops against a possible SEU in the internal nodes of the flip-flops.

Multiple sampling protection method

Three sampling is a conventional approach to detect the erroneous pulse at the input of a flip-flop ([5]and [13]). Figure 7 shows a three sampling scheme to detect a transient pulse. CLK and D are the clock and data inputs of the flip-flop, respectively. Using three samples *a*, *b*, and *c*, a three sampling method detects and corrects a possible transient pulse on D. To guarantee the correctness of this algorithm, the time interval between each two consecutive samples should be greater than the maximum width of the transient pulse (i.e., $\Delta \geq \tau_{max}$). The first sample is latched at $\Delta > \tau_{max}$ time before the rising edge of the clock. The second sample is latched at the rising edge of the clock. Finally, the third sample is latched at $\Delta(> \tau_{max})$ after the rising edge of the clock. In this scheme, *b* will be selected as the default output. If there is a discrepancy between the first two samples, the third sample (i.e., *c*) will be selected as the output. The first sample is called *voter* sample, the second sample is called *main* sample, and the third sample is called *arbiter* sample. The maximum time penalty of this method in the presence of a transient pulse is Δ.

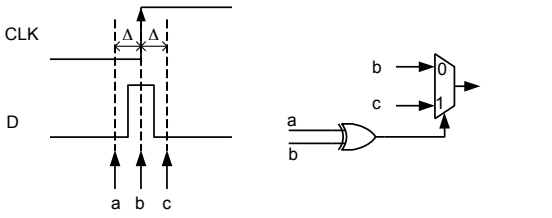

Figure 7 Three sampling to detect and correct a transient pulse

Figure 8 shows the sampling method to detect and correct a transient pulse and delay. Using the three samples, Figure 8-III shows the logic to detect and correct the SEU at the input of a flip-flop. Although, this three sampling method detect and correct the transient pulse and delay, it is sensitive to a transient pulse that may occur while the third value is sampled.

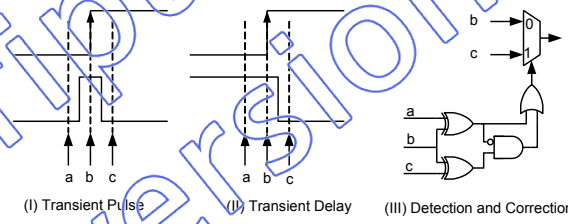

Figure 8 Three sampling to detect and correct a transient pulse and transient delay fault

Figure 9-I shows the failure case of the three sampling method. Based on the detector and corrector logic of Figure 8-III, instead of the correct logic value '1' the erroneous logic value '0' is latched in the victim flip-flop. In this case, a forth sample with $\Delta > \tau_{max}$ delay after the third sample can solve the issue (Figure 9-II). Figure 9-III shows the detector and corrector logic. In this circuit, the default value is the *main* sample (i.e., *b* sample). If a transient pulse or delay is detected during the sampling window of the flip-flop, the third sample is selected and latched in the flip-flop. The maximum time penalty of this method is $2 \times \Delta$.

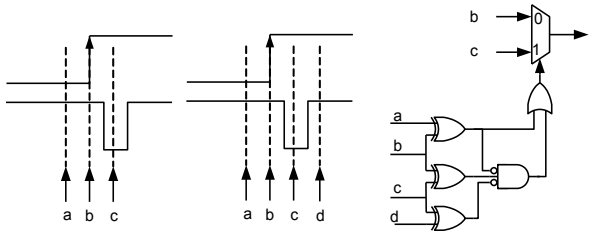

Figure 9 Four sampling method to detect and correct a transient pulse and transient delay fault

Note that, in this case, a new correct rising edge caused by a shortest path to this victim flip-flop should not interfere with the transient delay. Otherwise, the fourth sample may cause an incorrect value latched in the flip-flop; so, in this case, SET-hold time should be greater than $2 \times \Delta$ (i.e., $t_{SETh} > 2 \times \Delta$).

In the next section, we propose two architectures to implement the proposed sampling methodologies.

Proposed Structure

An architectural or circuit technique for implementing the proposed sampling methods should consider the following design issues:

1. **SET and SEU tolerant**: It should implement the three or four sampling method to eliminate all possible SET or SEU in the combinational and sequential parts.
2. **Power, time, and area overhead**: because the rare occurrence of SEU/SET, the proposed techniques and structures show introduce a low power, time and area overhead.
3. **Parameter variations**: Parameter variations (caused by local or global process variation, or environmental effects) in a deep-submicron design may uncertain the delay in a design. Furthermore, these variations may be data dependent. In other words, these variations may reveal their worst-case impact on circuit performance only under certain data sequence. Thus, finding the SET sensitive paths became difficult under these uncertainly. Furthermore, as a result of the process and of the environmental variations, the clock signal may have both *skew* (spatial variation) and *jitter* (temporal variation). The correctness of the proposed structure in present of these issues should be guaranteed.

For the sake of briefly and clarity, this paper focuses of the first two issues. However, some short solutions are proposed for the other issues.

SET/SEU tolerant flip-flop

Reusing the present test structures (e.g., scan flip-flops) in a circuit to cope with SET and SEU issue may be a promising technique to propose an optimum (low power, time, and area overhead) SET/SEU tolerant structure.

Using scan latches in parallel with system latches is becoming an efficient way to handle different problems during test and debug of a circuit ([14] and [15]). Reference [14] proposes a selective trigger scan architecture made of two parts (system part and test part) to reduce the test data volume and test dynamic power consumption. [15] proposes a microprocessor full hold-scan architecture that comprises two distinct circuits: a system flip-flop and a scan portion. This architecture is implemented in the 90nm Intel® Pentium® 4 processor.

Using the scan portion of these types of flip-flops, we implement the proposed sampling methods to obtain a soft-error tolerant flip-flop.

Figure 10-I shows our proposed architecture to detect a transient pulse at the input of the flip-flop. The flip-flop architecture consists of three parts: system, scan, and protection portions. Protection portion consists of three gates (an XOR, an AND, and an inviter) and a delay generator. The clocking scheme of the proposed architecture is based on the pulse-flip-flops [10] and the clocking signals are shown in Figure 11-I. Using a delay generator, the proposed architecture samples the first two samples of Figure 7-I, simultaneously (Figure 11-II). If there is a discrepancy between samples *a* and *b* the third sample (i.e., *c*) is latched as the output of the flip-flop.

When there is not any SET at the input of the flip-flop, this flip-flop can also tolerate an SEU in its internal nodes during its hold time, if the three samples *a*, *b*, and *c* are identical. To guarantee this equality, t_{SETh} should be grater than τ_{max}. Because one node may be upset, any bit-flip on *a* or *b* is corrected by node *c*. Furthermore, any bit-flip on node *c* does not change the output of the flip-flop. This flip-flop can be used on the transient delay insensitive paths.

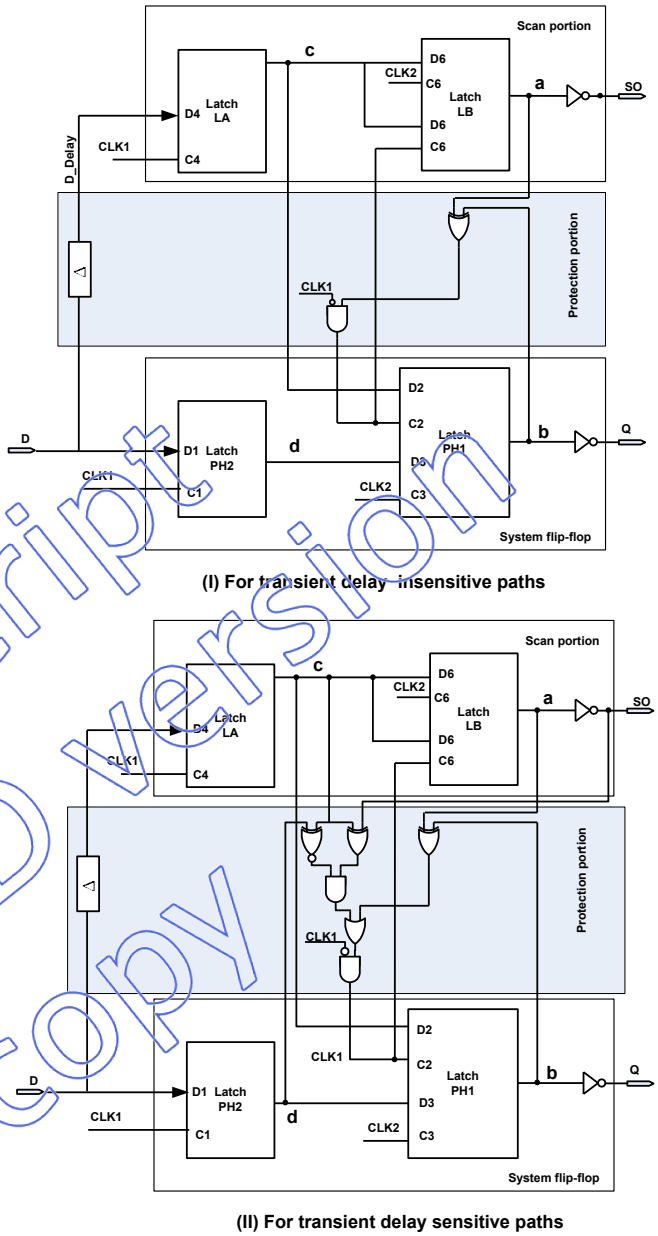

Figure 10 SEU-Tolerant flip-flop for non-critical paths

Figure 10-II shows our proposed architecture to detect a transient pulse and delay at the input of the flip-flop on the transient delay sensitive path. The flip-flop architecture consists of three parts: system, scan, and protection portions. Protection portion consists of seven gates and a delay generator. The clocking scheme of the proposed architecture is based on the pulse-flip-flops [10] and clocking signals are shown in Figure 11-II, -III. Using a delay generator, the proposed architecture samples the first two samples of Figure 9-II, simultaneously (Figure 11-II, -III)). If there is a discrepancy between samples *a* and *b* the third sample (i.e., *c*) is latched as the output of the flip-flop. In addition, sample c is also latched as the final output if there is a transient delay at the data input. This architecture can also tolerate an SEU at its internal node if $t_{SETh} > 2 \times \Delta$.

The condition $t_{SETh} > 2 \times \Delta$, which guarantees the SET detection, is compassed by considering a minimum-path length constraint during the design process. This minimum-path length can be

realized by adding buffers to the shortest path during logic synthesis. Therefore, this process introduces a certain amount of power and area overhead.

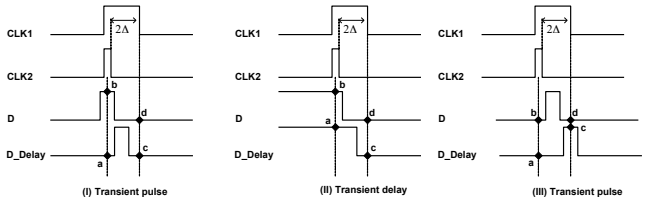

Figure 11 Clock and data signals

Clock skew and clock jitter may have negative impact on the proposed clocking scheme of Figure 11. A technique that can minimize this negative effect is to locally generate the CKL1 and CLK2 from the main system clock. However, this locally signal generation increases the power and area overhead.

6. Experimental Results

We have implemented a C++ program to detect number of flip-flops that are fed by transient delay (late edge) sensitive paths and early edge sensitive paths.

Table 1 shows some experimental results obtained by running the program on seven ISCAS89 benchmark circuits. We have simulated these circuits by using 100,000 inputs. The delay of paths is computed based on these inputs. The power consumption is obtained by computing the number of transition in the circuits during simulation.

The second column shows the parentage of flip-flops that are fed by transient fault sensitive paths. Column 3 shows the percentage of flip-flops that are simultaneously fed by transient fault sensitive and early edge sensitive paths. Column 4 shows the percentage of all the flip-flops that are fed by early edge sensitive paths. The area overhead due to applying the proposed flip-flop architecture is shown in Column six. Finally, an finally the last column shows the power overhead caused by using flip-flop architecture in the normal operation.

Table 1 Overheads of the proposed flip-flop

Circuit	# LESP FF*	# LESP_EESP FF**	Area Overhead	Power Overhead
s298	21.4%	0	21.3%	0.01 %
s344	53.3%	50%	23.5%	1.66%
s349	33.3%	80%	20.7%	1.45%
s526	19.0%	0	19.8%	0.01%
s1196	5.5%	0	8.3%	0.003%
s5378	6.1%	0	13.1%	0.09%
s35932	15.7%	100%	20.6%	0.04%

* LESP FF = Flip-Flops on Late Edge Sensitive Paths
** LESP_EESP FF = LESP Flip-Flops on Early Edge Sensitive Paths

The flip-flop architecture of Figure 10-(II) can detect types of delay (that are less than τ_{max}) faults. Thus the proposed architecture can be used in an online delay testing scenario. This multipurpose testing of the proposed architecture can justify its area overhead on some benchmark circuits.

7. Future work

The time penalty due to the protection mechanism may generate a transient delay fault for the downstream flip-flops. Thus, some architectural techniques are needed to handle this issue. Inserting a stall in the next level flip-flops in a pipeline datapath is a solution. Other techniques are disabling the clock for one cycle or borrowing some time from next clock cycle. These ideas need more research and the authors are working on them as their future work. Proposing new SET-tolerant flip-flops for next generation of high speed microprocessor whose clock cycles will be comparable with the width of SET are also future work of authors.

8. Conclusions

This paper considers logic circuits with many critical paths; and studies the effect of single event transient (SET) caused by particle strike on the nodes along the critical paths. This paper shows three different erroneous effects of a SET at the input of a flip-flip: a transient pulse, an early edge and late edge (transient delay). The paper also proposes two flip-flop architectures to detect and correct these erroneous effects.

References:

[1] T. Karnik, P. Hazucha, and J. Patel, "Characterization of soft errors caused by single event upsets in CMOS processes," *IEEE Trans. Dependability and Secure Computing*, Vol. 1, No. 2, pp. 128-143, 2004.

[2] S. Mitra, T. Karnik, N. Seifert, and M. Zhang, "Logic soft errors in sub-65nm technologies design and CAD challenges," 42nd *Design Automation Conference, DAC'05*, pp. 2-4, 2005.

[3] K. Mohanram, "Simulation of transients caused by single-event upsets in combinational logic," *International Test Conference* (ITC'05), Nov. 2005.

[4] B. S. Gill, C. Papachristou, F. G. Wolff, and N. Seifert, "Node sensitivity analysis for soft errors in CMOS logic," *International Test Conference* (ITC'05), 2005.

[5] S. Krishnamohan, and N. R. Mahapatra, "A high-efficiency technique for reducing soft errors in static CMOS circuits," *Proc. IEEE Inter. Conf. Computer Design* (ICCD'04), pp.126-131, Oct. 2004.

[6] M. Zhang, and N. R. Shanbhag, "An energy-efficient circuit technique for single event transition noise-tolerance," *IEEE international Symposium on Circuits and Systems* (ISCAS'05), pp.636-639, May 2005.

[7] C. Zhao, X. Bai, and S. Dey, "A static noise impact analysis methodology for evaluating transient error effects in digital VLSI circuits," *International Test Conference* (ITC'05), pp. ***, Nov. 2005.

[8] R. Naseer, J. Draper, "The DF-DICE storage element for immunity to soft errors," *48th Inter. Midwest Symp. On Circuit and Systems*, (MWSCAS'05), 2005.

[9] G. C. Messenger, "Collection of charge on junction nodes from ion tracks," IEEE Trans. Nucl. Sci., pp. 2024-2031, 1982.

[10] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, "*Digital Integrated Circuits*," Pearson Education, Inc. Upper Saddle River, New Jersey 07458, 2003.

[11] V. G. Oklobdzija, "Clocking in Multi-GHz Environment," in *Proc. 23rd Inter. Conf. on Microelectronics* (MIEL'02), pp.561-568, May 2002.

[12] T. Burd, T. Pering, A. Stratakos, and R. Brodersen, "A dynamic voltage scaled microprocessor system," in Proc. *IEEE Int. Solid-State Circuits Conf.,* pp. 294–295, 2000.

[13] D. G. Mavis, and P. H. Eaton, "Soft error rate mitigation techniques for modern microcircuits," *IEEE Proc. Annual Reliability Physics Sym*. Pp. 216-225, 2004.

[14] S. Sharifi, M. Hosseinabady, Zainalabedin Navabi, "Reducing Power, Time and Data Volume in SoC Testing Using Selective Trigger Scan Architecture", *International Symposium on Defect and Fault Tolerance in VLSI Systems* (DFT 2003), pp. 352-360, 2003.

[15] R. Kuppuswamy, P. DesRosier, D. Feltham, R. Sheikh, and P. Thadikaran, "Full hold-scan system in microprocessors: Cost/Benefit analysis," *Intel Technology Journal,* Vol. 18, No. 1, Feb. 2004; http://developer.intel.com/technology/itj/2004/volume08isssue01/.