A Highly-Efficient Technique for Reducing Soft Errors in Static CMOS Circuits

Srivathsan Krishnamohan and *Nihar R. Mahapatra*
E-mail: {krishn37, nrm}@egr.msu.edu
Department of Electrical & Computer Engineering, Michigan State University, East Lansing, MI 48824, USA

Abstract— Soft errors are functional failures resulting from the latching of single-event transients (transient voltage fluctuations at a logic node or SETs) caused by high-energy particle strikes or electrical noise. Traditionally, they have been deemed to be a problem in memory structures, for which effective techniques (such as error correcting codes) are well known. However, due to technology scaling and reduced supply voltages, they are expected to increase by several orders of magnitude in logic circuits. Existing circuit and architectural approaches to addressing soft errors in logic circuits have appreciable area/cost, performance, and/or energy overheads or are limited to particular types of circuits (combinational or sequential). We present a very efficient and systematic error masking technique that uses the same circuitry to cope with soft errors in combinational and sequential circuits. It prevents an SET pulse of width less than approximately half of the slack available in the propagation path from latching and turning into a soft error. The SET is masked without additional delay and within the clock cycle time in an area- and energy-efficient manner, which makes this technique attractive for commodity as well as reliability-critical applications. Our technique also tolerates soft errors in the overhead circuitry, which we minimize through clustering. Application of our technique to ISCAS85 benchmark circuits yields an average SER reduction of 70.93% with an average area overhead of only 11.98%.

1. Introduction
A. Background and motivation

In addition to traditional design metrics of performance, energy, and cost, technology scaling has added reliability and robustness too. *Reliability* is normally defined as the immunity to hard failures such as electromigration, hot carrier effects, or dielectric breakdowns. However, frequent occurrence of transient faults or soft errors due to crosstalk noise and radiation-induced upsets can also affect reliability of circuits. *Design robustness* is defined as the ability of a circuit to operate correctly under varying process, temperature, voltage, and noise conditions. Previously, reliability and robustness were issues to be considered during design of chips used in medicine, military, nuclear, or space applications. But nanometer CMOS technology has made reliability and robustness important issues in the design of commodity chips too.

Soft errors are functional failures resulting from the latching of single-event transients (transient voltage fluctuations at a logic node or SETs) caused by electrical noise or external radiation. In this paper, we are concerned with static CMOS circuit *soft errors*, which are transient functional failures due to electrical noise or external radiation. Although most of our discussion applies to soft errors due to either source,

our focus is on radiation-induced errors, particularly, those resulting from high-energy neutron strikes. For a SET to cause a soft error, it must propagate to a primary output (PO) gate and be finally captured by an output flip-flop (FF). However, a soft error will not occur if the SET is either: (1) *logically masked*—some other input of a gate in the SET propagation path determines its output instead of the SET; (2) *electrically masked*—the SET is attenuated sufficiently due to the electrical properties of gates in the propagation path; or (3) *latching-window masked*—the SET reaches an output FF, but not at the clock edge where the FF captures the value [1].

The smallest deposited charge required at a gate to create an SET pulse that results in a soft error, if it is not logically or latching-window masked, is called the *critical charge* Q_{crit} of the SET propagation path. The charge deposited is directly related to the energy of the striking particle and soft error rate (SER) increases exponentially with decrease in Q_{crit} [2]. Soft errors pose increased reliability problems in nanometer-scale circuits because: (1) smaller, faster transistors lower electrical masking effects [1], (2) reduced source/drain capacitances and supply voltages lower Q_{crit} [3], (3) and higher clock frequencies reduce latching-window masking probability [1]. Recent studies have shown that SER per chip of logic circuits will increase nine orders of magnitude when minimum feature size scales from 600 nm to 50 nm, becoming comparable to SER per chip of unprotected memory elements [1]. This necessitates an efficient design approach for static CMOS circuits that would make them soft-error resilient without adversely affecting other design considerations such as power, performance, and cost.

B. Related work

Traditional techniques to provide soft error tolerance rely on triple modular redundancy (TMR), in which the original circuit is triplicated and a majority voter used to determine the final output. However, this technique involves high overhead ($> 200\%$) in terms of area and cost, which limits its usage to reliability-critical applications. Various ideas for soft error tolerance based on time redundancy were presented in [4]. The time domain majority voter presented in [4] has a performance overhead since the sampling is started after the longest path in the circuit settles. Hence, an online error detection and retry procedure was considered better [5]. Online or concurrent error detection can be achieved by using self checking circuits [6], [7] or by exploiting temporal redundancy of signals [5]. Self checking circuits may require high hardware cost for arbitrary logic functions. Online error

detection and retry may affect performance (throughput) and cannot be used in real-time systems to overcome transient faults due to electrical noise or external radiation. Another technique called partial error masking, corrects errors with lower overhead than traditional TMR techniques by utilizing the difference in soft error vulnerabilities of gates. But, it masks SEU errors only in CLBs and has higher overhead compared to the technique presented in this work [8]. Prior efforts have also focused on latch design for mitigating soft errors [9], [10] and combinational logic design for preventing pulse spreading [11]. Our technique uses a delay line that is common to one or more combinational logic blocks (CLBs) as opposed to a delay line within each latch as done in [10]. The latch design in [9] requires resistor insertion to slow down the input stage, which incurs both performance and area penalty. Time redundancy based architectural approaches also have significant performance and power overheads and design time cost [12].

C. Scope and contributions of our work

In this paper, we present an efficient 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. It has a number of features that make it attractive compared to existing approaches: (1) It modifies only the flip-flops of a combinational logic block (CLB) for sampling PO values and thus has lower area and power overheads. (2) Further helping lower these overheads is the use of a common delay line for an entire CLB or even multiple CLBs to produce control signals used in the technique. (3) In CLBs that have sufficient slack at a significant fraction of the PO gates, which is quite common, SER can be reduced markedly without any performance overhead. Otherwise, SER can be reduced with some performance overhead. (4) The proposed design technique also masks soft errors in both CLB and the master stage of the flip-flop as compared to [13]. Sampling points closer are clustered and triggered with a single control signal which reduces area overhead for a small increase in SER as compared to [13].

The remainder of the paper is organized as follows. Sec. 2 explains our error-masking technique in detail, along with the circuits used to achieve this. Sec. 3 describes the simulation setup and presents results obtained with ISCAS85 circuits, and finally, Sec. 4 concludes.

2. Time Redundancy Based Error Masking
A. Exploiting timing slack

We first analyze the soft-error vulnerability of a CLB in the original circuit, and then, in the next paragraph, explain our technique conceptually and analyze how it exploits timing slack to reduce SER. All time instants in the following discussion are specified in terms of elapsed time after a cycle begins. Let T denote the cycle time. When an SET pulse is generated at the output of a static CMOS gate in a combinational circuit due to a high-energy particle strike, it may propagate through a path u and be captured by an output flip-flop (FF), and thus cause a soft error. At $t_3 = T - t_{\text{setup}}$, u's output (primary output) is sampled by an output FF, where
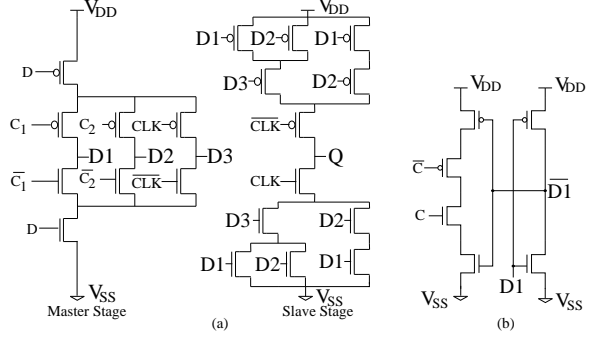

Fig. 1. (a) A modified C^2MOS flip-flop to sample and latch signal value at different time instances within a clock cycle. The slave stage contains a majority voter to vote among the different sampled values. (b) Output node D1 is dynamic after C and $\overline{C}$ become zero and one in (a). Output node D1 or D2 can be kept static by cross coupled inverter shown.

t_{setup} is the setup time of the FF. Assume that a maximum of single transient pulse is caused from a particle strike at some gate per cycle (this is referred to as a *single-event upset* or SEU), but that this may propagate to multiple FFs connected to the circuit output. Consider an SET pulse of width w that can begin at any time during a cycle with equal probability. The probability $P(w)$ that this pulse, will latch at an output FF and cause a soft error (i.e., it will overlap the sampling instant t_3) can be determined to be $P(w) = \frac{w}{T}$.[1]

Since the effect of an SET is only temporary, it is possible to prevent a soft error by exploiting timing slack available in the path u as follows. Let t_1 denote the worst-case propagation delay from the primary inputs to the output of u. The slack for u is then $t_s = t_3 - t_1$, i.e., in the absence of an SET, u's output will be stable at its correct value in the time interval $[t_1, t_3]$. If in addition to t_3, we sample u's output (in the connected flip-flop) at t_1 and t_2 too, where $t_1 < t_2 < t_3$, and we then perform majority voting among the three sampled values, we will be able to obtain the correct value of u's output whenever an SET pulse does not overlap more than one sampling instant. Let $t_{s12} = t_2 - t_1$ and $t_{s23} = t_3 - t_2$, and let $t_{s12} \leq t_{s23}$ without loss of generality. The probability $P(w)$ that an SET pulse of width w, after reaching u's output, will cause a soft error (i.e., it will overlap at least two sampling instants) can be verified to be as follows: (1) $P(w) = 0$ when $w < t_{s12}$; (2) $P(w) = \frac{w - t_{s12}}{T}$ when $t_{s12} \leq w < t_{s23}$; (3) $P(w) = \frac{2w - t_s}{T}$ when $t_{s23} \leq w < t_s$; and (4) $P(w) = \frac{\min(w, T)}{T}$ when $w \geq t_s$. The transient pulse and its overlap with different sampling points to cause a soft error is shown in Fig. 2. Thus, in the first three cases, our technique improves soft-error tolerance and has the same tolerance as the original circuit in the last case. In the first case, soft errors are always prevented. To maximize the pulse width that is guaranteed to be tolerated, we can choose $t_2 = \frac{t_1 + t_3}{2}$ or $t_{s12} = t_{s23}$, so that SET pulses of width less than half of slack at u are guaranteed to be tolerated.

We now move onto implementation issues. First, we

---

[1]More precisely, a soft error will be caused if the SET pulse overlaps the setup and hold time interval of the output FF.
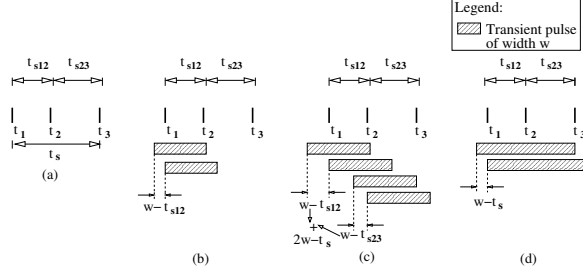
Fig. 2. Figures (b), (c), and (d) show different transient pulse widths and their starting and ending times when they overlap two sampling points to cause soft error. (a) Effective slack available in a path and the time when the FF samples: t_1, t_2 and t_3. Three different cases for transient pulse width: (b) Transient pulse width is greater than t_{s12} and covers both t_1 and t_2. (c) Transient pulse width is greater than t_{s12} and t_{s23}, hence can overlap both t_1 and t_2 or t_2 and t_3. (d) Transient pulse width is greater than t_s and completely covers the slack time t_s available.

Fig. 3. **Generation of control signals C and $\overline{C}$.** (i) $\overline{C_1}$ and $\overline{C_2}$ are generated by delaying CLK if they go low after $\frac{T}{2}$. DCLK shown is used as $\overline{C_1}$ or $\overline{C_2}$. (ii) $\overline{C_1}$ and $\overline{C_2}$ are generated by ANDing CLK and delayed $\overline{CLK}$ when they go low before $\frac{T}{2}$. C is generated by inverting $\overline{C}$ in both cases.

discuss circuits for sampling the path's output values and majority voting. Then we describe a delay chain technique used to generate the sampling control signals for the FF. In the above discussion, we exploited the complete slack from t_1 till t_3 to reduce SER. However, for implementation efficiency (as explained below), we may not exploit this slack completely, and so may sample at time instants t'_1, t'_2, and t_3 (the last sampling time remains unchanged), such that $t'_1 \leq t'_1 < t'_2 < t_3$. We define $t'_s = t_3 - t'_1$, $t'_{s12} = t'_2 - t'_1$ and $t'_{s23} = t_3 - t'_2$, and let $t'_{s12} \leq t'_{s23}$ without loss of generality.

B. Output sampling and majority voting

We apply our technique to only those paths that have some reasonable slack. The sampling is performed by adding two sets of n and p control transistors (corresponding to t'_1 and t'_2) to a FF as shown in Fig. 1(a). At sampling time, sampling control signal C ($\overline{C}$) goes high (low), which disconnects output node F from V_{DD} and GND, thus preventing any further transitions and completing the sampling. A majority voter embedded into the slave stage of the FF determines the final output value (see Fig. 1(a)). Since the load on the PO gate connected to the modified FF increases, the extra delay reduces the effective slack that can be exploited. To reduce the susceptibility of node D1 to particle strikes after sampling (when it is essentially a dynamic node), cross-coupled inverters are added to it to make it static (see Fig. 1(b)). Explicit switched-capacitor can also be added to node $\overline{D1}$ to harden the cross-coupled inverter against soft errors [14]. The capacitor addition should be done based on SER requirements and power and area overheads incurred.

An SET pulse generated in the CLB and reaching the modified FF will be tolerated as per our analysis in Sec. 2.1. An SET pulse generated only at D1, or D2, or D3 of the modified FF due to a particle strike (an SEU) can always be tolerated because of majority voting. However, a single-event multiple upset (SEMU), i.e., a single particle strike causing transient pulses to be generated at multiple data nodes, can be a problem as it can cause a wrong value to appear at the majority voter output. Since it is hard to characterize the charge required for an SEMU through simulation, we do not
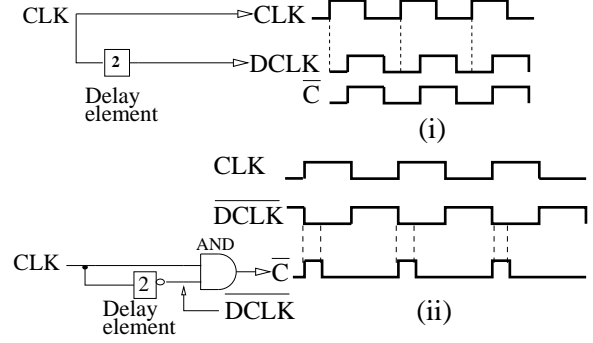
include soft error contribution of FFs to calculate original and final reduced SER (i.e., we present quantitative SER reduction results only for CLB). However, the data nodes D1, D2, and D3 in the modified FF can be spaced apart in the layout, by placing the cross coupled inverters and the layout of any explicit switched capacitances present between the data nodes. This would further reduce the chances of a SEMU occurring in the FF itself.

There are two cases when the soft errors are not masked. Error pulses generated at output of majority voter gate are not masked, while transient pulses with sufficient width to overlap setup and hold window of a flip-flop occurring in paths without reasonable slack are not masked. Delay faults can be handled by providing frequency guardband in the circuit [15]. Errors occurring at the output of a majority voter gate affect only the next stage in pipeline, which is corrected by using our technique in the subsequent pipeline stage. In case of reconvergent paths where transient pulse propagates through both paths, a single logical flip originating before reconvergent paths begin can affect more than one sampling point. An error can occur if the delay difference between the reconverging paths makes the same transient pulse overlap two sampling points. To protect the sampling points t'_{s12} and t'_{s23} should be made greater than the delay difference between reconverging paths plus the overlapping error pulse width, or delay difference between reconverging paths can be reduced by increasing the delay of faster path.

C. Delay chain

The control signals C and $\overline{C}$ are generated using the circuit shown in Figure 3. For ease of explanation we explain the generation using the NMOS control signal $\overline{C}$. The generation of control signals depend on when $\overline{C}$ and C go low. $\overline{C}$ is generated by delaying CLK if $\overline{C}$ goes low after $\frac{T}{2}$, while it is generated by ANDing CLK and delayed $\overline{CLK}$ when $\overline{C}$ goes low before $\frac{T}{2}$. C is generated by inverting $\overline{C}$ in both cases. Particle strikes in the control signal generation circuit can also cause soft errors due to wrong value being latched. The occurrence of such soft errors is determined by the sampling time t'_1, t'_2, and t_3 for a FF. Since sampling time t_3 always occurs at T-t_{setup}, we only consider the occurrence of t'_1 and

t_2' with respect to $\frac{T}{2}$ (CLK is symmetric and for simplicity $t_3 = T$ is used here). We do not consider particle strikes on the CLK signal itself due to high load on CLK signal.

1) $t_1' < \frac{T}{2}$ and $t_2' < \frac{T}{2}$: $0 \rightarrow 1$ logic flip occurring in the delay chain before t_1' and extending till t_2' will make both $\overline{C_1}$ and $\overline{C_2}$ low before t_1'. $\overline{C_2}$ remains low till t_2' which causes a wrong value to be latched in both D1 and D2. The corresponding waveforms are shown in Fig. 4(a).

2) $t_1' < \frac{T}{2}$ and $t_2' > \frac{T}{2}$: In this case t_{2d}', the time by which CLK signal has to be shifted to produce control signal $\overline{C_2}$ is

$$t_2' = t_1' + (\frac{T - t_1'}{2}) = \frac{t_1'}{2} + \frac{T}{2}$$
$$t_{2d}' = t_2' - \frac{T}{2} = \frac{t_1'}{2}$$

smaller than $t_{1d}' = t_1'$. The corresponding waveforms for $\overline{C_1}$ and $\overline{C_2}$ are shown in Figure 4(b). A $0 \rightarrow 1$ logic flip occurring in $\overline{C_2}$ as shown by the dotted line would cause $\overline{C_1}$ to go low earlier than t_1', which may cause a wrong value at D1 in the gate shown in Figure 1(b). However, as $\overline{C_2}$ and hence D2 are not affected, the majority value still remains correct. Hence, a $0 \rightarrow 1$ logic flip occurring in $\overline{C_2}$ does not cause a soft error. One to zero logic flip occurring in signal $\overline{C_2}$ before t_1', could cause an error in D2 if the error pulse width extends till t_2'. Since $\overline{C_1}$ only changes to one, D1 is not affected by this $1 \rightarrow 0$ error in $\overline{C_2}$, which gives a correct value at the majority voter output.

3) $t_1' > \frac{T}{2}$ and $t_2' > \frac{T}{2}$: The corresponding waveforms $\overline{C_1}$ and $\overline{C_2}$ are shown in Figure 4(c). A one to zero logic flip occurring in $\overline{C_1}$ before t_1' and extending till t_2' can cut-off both NMOS transistors controlled by $\overline{C_1}$ and $\overline{C_2}$, which could cause wrong values to be latched in both D1 and D2.

To avoid soft errors described in cases one and three, separate delay lines are used to generate control signals $\overline{C_1}$ and $\overline{C_2}$ (only in particular cases as described later). A voltage controlled current starved inverter shown in Fig. 5 is used as a delay element to form the delay lines [16], since the delay can be adjusted post-fabrication by changing the controlling voltage to counter static process variability. Due to discrete nature of delays produced by the delay elements sampling cannot happen exactly at the ideal t_1' and t_2' times, which are equal to worst case output settling time of the path and $\frac{t_1' + t_3}{2}$, respectively. This requires us to determine the nearest sampling time which can be used to reduce SER. The number of discrete control signals C and $\overline{C}$ to be generated can be reduced by *clustering* and using common control signals for flip-flops whose sampling time occur close together. This reduces the area overhead by using fewer delay elements to generate control signals and fewer wires to route. However, due to clustering of control signals sampling may be done at new time instants t_1'', t_2'', and t_3 (the last sampling time remains unchanged), such that $t_1' \le t_1'' < t_2'' < t_3$. We define $t_s'' = t_3 - t_1''$, $t_{s12}'' = t_2'' - t_1''$
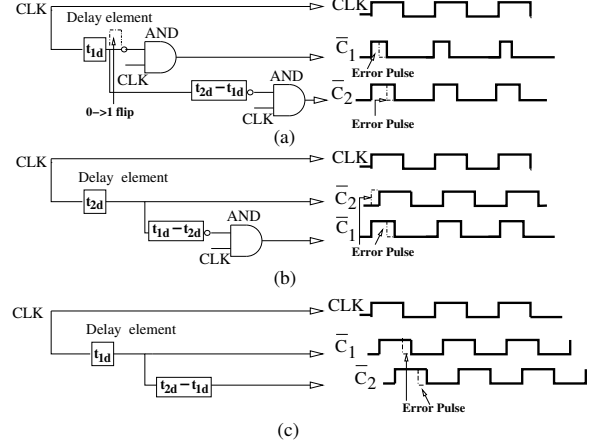

Fig. 4. (a) $t_1' < \frac{T}{2}$ and $t_2' < \frac{T}{2}$. Zero to one logic flip affects both $\overline{C_1}$ and $\overline{C_2}$. (b) $t_1' < \frac{T}{2}$ and $t_2' > \frac{T}{2}$. Zero to one logic flip affects only $\overline{C_1}$. (c) $t_1' > \frac{T}{2}$ and $t_2' > \frac{T}{2}$: Both $\overline{C_1}$ and $\overline{C_2}$ are affected.
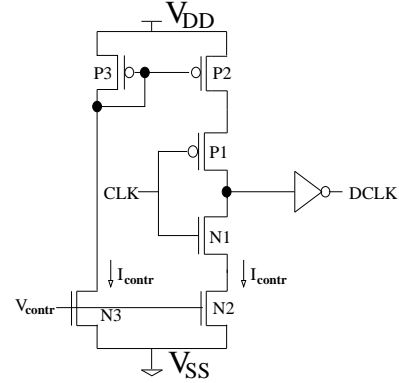

Fig. 5. A voltage controlled current starved inverter which delays the system CLK to produce DCLK. High to low propagation delay (t_{pHL}) is set by I_{contr} which is controlled by gate voltage of N2 V_{contr}. Low to high delay (t_{pLH}) is also controlled by V_{contr} through N3 and current mirror comprising transistors P3 and P2.

and $t_{s23}'' = t_3 - t_2''$. The new sampling time intervals t_{s12}'' and t_{s23}'' could reduce the effective error pulse width that can be tolerated. Therefore, the sampling times t_1'' and t_2'' have to be selected by minimizing the decrease in SER reduction obtained.

To cluster the control signals, we first determine t_{1d}' and t_{2d}', the time by which CLK has to be delayed to generate $\overline{C_1}$ and $\overline{C_2}$ for the flip-flops that are being controlled. The maximum of t_{1d}' and t_{2d}' over all points is always less than $\frac{T}{2}$, since control signals going low after $\frac{T}{2}$ are generated by delaying the CLK signal for a time less than $\frac{T}{2}$. Next, the time interval $t_{1\delta}'$ and $t_{2\delta}'$ over which t_1' and t_2' can be varied to get t_1'' and t_2'' are determined. The time intervals are set by the maximum width of the error pulse (w_{max}) that needs to be tolerated in the path, which can be provided by the user. The time intervals for paths where t_{s12}' and t_{s23}' are greater than w_{max} (for simplicity we use the maximum error pulse

width in the circuit) are given by:

$$t'_{1\delta} = (t'_{s12} - w_{max})/2$$
$$t'_{2\delta} = (t'_{s23} - w_{max}) \qquad (1)$$

We limit the value of t''_1 between t'_1 and $t'_1 + t'_{1\delta}$, and that of t''_2 between $t'_2 - (t'_{2\delta}/2)$ to $t'_2 + t'_{2\delta}$. In paths where t'_{s12} and t'_{s23} are not greater than w_{max} we use a threshold of 100 pS for clustering. We then bin the t'_{1d} and t'_{2d} values in regular intervals and then construct a delay line with delay taps closer to the mean of bins with one or greater items in them. Tapered buffers are used to distribute the control signals derived from the delay taps. Finally, we allocate t'_{1d} and t'_{2d} for the sampling points such that t''_1 and t''_2 do not exceed their respective boundaries determined before. We construct a separate delay line for control signals C_1 and $\overline{C_1}$ corresponding to sampling time t'_1, when $t'_1 > \frac{T}{2}$. CLK and $\overline{CLK}$ are used as control signals $\overline{C_2}$ and C_2 corresponding to sampling time t'_2, where $t'_2 < \frac{T}{2}$. This avoids the use of a separate delay line to prevent soft errors occurring due to a $0 \rightarrow 1$ logic flip, as described before.

3. Simulation Results

ISCAS85 circuits were synthesized in 0.18 micron technology using the standard cell library described in [17]. Original and reduced SER of the circuit are given by equation 2.

$$SER_{orig.} = \sum_{i=1}^{n} SER(g_{i,w_{orig}})$$
$$SER_{red.} = \sum_{i=1}^{n} SER(g_{i,w_{t''_s/2}})$$
$$SER(g_i) = \sum_{\forall j} \Big(\sum_{k=1}^{m} (SER(Q_{L_k}) - SER(Q_{R_k}))$$
$$\times P_{latch}(w_{Q_{L_k}}) \Big) \times P_j$$
$$(2)$$

$SER(g_{i,w_{orig}})$ and $SER(g_{i,w_{t''_s/2}})$ are the soft error contribution of gate g_i when the transient pulse width required to cause an error are w_{orig} and $w_{t''_s/2}$. $SER(Q_{crit}) = k \times F \times A \times e^{(-\frac{Q_{crit}}{Q_s})}$ [2], where F is the incident neutron flux (value of 0.00565 neutrons$*cm^{-2}s^{-1}$ was used), A is the area of the circuit sensitive to particle strikes, in cm^2, Q_{crit} is the smallest charge required to cause a logic upset, Q_s is the charge collection efficiency of the device in fC, k is a technology independent constant equal to 2.2×10^{-5}. $SER(Q_{crit})$ gives the soft error rate for charges equal to and greater than Q_{crit}. The soft error contribution of each gate g_i is calculated starting from Q_{crit} up to a charge of 3 pC, which can be approximated to be the maximum charge collected by a CMOS device on an epitaxial layer [18]. In order to calculate the SER of a gate for charges between Q_{crit} and 3 pC, we divide the charge values into m equal intervals of 50 fc. The soft error contribution of each interval is calculated by subtracting SER corresponding to right endpoint from the left [1]. The soft error contribution of each interval is

weighted by the latching window probability of a transient pulse produced by a charge Q_{L_k}, corresponding to the left endpoint in the interval. The latching probability is calculated differently for original and modified circuit as discussed in Sec. 2.1. The SER of each gate is calculated with respect to all latches in its fanout cone and weighted by the logical masking probability P_j through the path to latch j.

Q_{crit} of a gate depends on the fanout capacitance and electrical masking through the path to flip-flop. Q_{crit} of each gate was characterized through Spice simulation using TSMC 0.18 micron transistor models with V_{DD}=1.8V, for different values of fanout capacitance both for the original circuit and when the sampling is done. A representative path with the actual and modified flip-flop connected at the end, and using varying gate levels was used to take into account the electrical masking during Q_{crit} simulation. The SER of both the original and modified circuit depends on the probability that a gate is sensitized through a particular path P_j, called logical masking probability. As ISCAS85 circuits do not have specific input patterns to test them, the logical masking probability P_j is generated as a random number with uniform distribution between zero and one.

We calculate the original SER of each circuit using equation 2. The reduced SER is calculated for two cases: (1) When the sampling time t'_1 and t'_2 are ideal, (2) Sampling time t''_1 and t''_2 obtained after clustering of control signals are used. The original area of ISCAS85 circuits were obtained from the synthesis tool cadence physically knowledgeable synthesis (PKS), while the area overhead is equal to the sum of area occupied by delay line and the associated buffers, the modified FFs, and a five percent wiring overhead. The results are presented in Table I for ISCAS85 circuits, where N_{trig} represents the number of flip-flops which were modified as shown in Figure 1(a). Soft error rate reduction corresponding to cases (1) and (2) are presented as ideal SER reduction and SER reduction (Clust.), respectively. Latches in path with slack t''_s (=$t_3 - t''_1$), where Q_{crit} required for producing error pulse of width $t''_s/2$ is not greater than original Q_{crit} are not triggered.

Circuit	Circuit Features			Ideal SER Redn. %	SER Redn. (Clust.) %	Area Ovhd. %
	PIs	POs	N_{trig}			
c432	36	7	3	55.66	50.15	16.8
c1908	33	25	16	77.15	69.45	10.4
c2670	233	140	60	83.35	74.3	13.8
c3540	50	22	20	90.17	84.81	9.4
c7552	207	108	67	73.24	64.68	10.2
c5315	178	123	93	88.24	82.16	11.3
Avg.			43	77.97	70.93	11.98

TABLE I
SER REDUCTION FOR ISCAS85 CIRCUITS

The area overhead depends on the number of modified flip-flops, the number of distinct sampling times and the maximum sampling time which contribute to the delay element

overhead. If the number of sampling times are close together, then the delay element overhead can be reduced more (by clustering) without significant loss of SER reduction, as compared to circuits with sampling times wide apart. The delay lines can be shared across multiple modules which would further reduce their area as well as power overheads. The active energy consumed by a module (without leakage power) is equal to $C_{EFF} \times V_{DD}^2$, where C_{EFF} is the effective capacitance switched every clock cycle. For the ISCAS85 circuits it is hard to calculate the active energy consumed, since the switching activity is difficult to estimate without benchmark inputs. The extra capacitance switched in the error masked circuit is because of the control transistors and the majority voter added to FF, and the delay lines. Since the overhead is quite low, the extra energy would be small in comparison to the energy consumed by the original circuit, as well as TMR schemes used for SER reduction which have a greater than 200% energy overhead.

The results presented here are for zero delay overhead i.e., the critical path delay is not affected, excluding the increase in the CLK-Q delay of the modified flip-flop. C499/C1355 which have the same overall function are not selected due to the presence of balanced paths in the circuit. Balanced static CMOS circuits attenuate noise pulses within four stages [15], which reduces the SER of such circuits. However, if the ISCAS85 circuits used were synthesized with delay balanced paths the ratio of SER reduction to area and power overhead would have been much lower (overhead is greater) compared to the technique presented in this paper. This is because delay elements have to be inserted in each of the individual unbalanced paths. As technology scales, clock frequency is increasing which decreases the absolute value of slack in circuits. However, as the time constant for charge collection process of a device decreases exponentially with minimum gate length [1], current pulse width due to particle strike also decreases. The decrease in current pulse width coupled with decrease in gate output capacitance, leads to a decrease in the width of SET as technology scales. This should allow us to exploit the reduced slack available in a path to reduce SER using the technique discussed.

4. Conclusions

We presented an efficient time redundancy based design technique for error masking and recovery. This technique can be used to improve the reliability of a circuit, by reducing transient faults caused due to cross-talk or soft errors due to particle strikes within the slack available in a circuit. We control flip-flops only in paths with sufficient slack which ensures that the delay increase caused by the addition of majority voter and control transistors to the flip-flops does not affect the timing of the circuit. There are two cases when the soft errors are not masked. Error pulses generated at output of majority voter gate are not masked, while transient pulses in critical paths are not masked. Results show an average SER reduction of 70.93%, with an average area overhead of 11.98% and zero performance overhead, which is significantly better compared to any of the current techniques.

References

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

[2] P. Hazucha and C. Svensson, "Impact of CMOS technology scaling on the atmospheric neutron soft error rate," *IEEE Transactions on Nuclear Science*, vol. 47, no. 6, pp. 2586–2594, Dec. 2000.

[3] S. Hareland, J. Maiz, M. Alavi, K. Mistry, S. Walsta, and C. Dai, "Impact of CMOS process scaling and SOI on soft error rates of logical processes," in *Symposium on VLSI Technology Digest of Technical Papers*. IEEE, 2001, pp. 73–74.

[4] M. Nicolaidis, "Time redundancy based soft-error tolerance to rescue nanometer technologies," in *Proc. International VLSI Test Symposium*, 1999.

[5] L. Anghel and M. Nicolaidis, "Cost reduction and evaluation of a temporary faults detecting technique," in *Proc. Design Automation and Test Europe*, 2000.

[6] J. Lo, "A novel area-time efficient static CMOS totally self-checking comparator," *IEEE Journal of Solid-State Circuits*, vol. 28, pp. 165–168, Feb. 1993.

[7] C. Metra, M. Favalli, and B. Ricco, "Self-checking detection and diagnosis of transient, delay, and crosstalk faults affecting bus lines," *IEEE Transactions on Computers*, vol. 49, pp. 560–574, June 2000.

[8] K. Mohanram and N. A. Touba, "Partial error masking to reduce soft error failure rate in logic circuits," in *Proc. International Symposium on Defect and Fault Tolerance in VLSI Systems*, 2003, pp. 433–440.

[9] H. Cha and J. Patel, "Latch design for transient pulse tolerance," in *Proc. ACM International Conf. Computer Design (ICCD)*, Oct. 1994, pp. 385–388.

[10] K. Hass, J. Gambles, B. Walker, and M. Zampaglione, "Mitigating single event upsets from combinational logic," in *Proc. 7th NASA Symposium on VLSI Design*. NASA, 1998.

[11] M. Baze and S. Buchner, "Attenuation of single event induced pulses in CMOS combinational logic," *IEEE Transactions on Nuclear Science*, vol. 44, pp. 2217–2223, Dec. 1997.

[12] S. Mukherjee, C. Weaver, J. Emer, S. Reinhardt, and T. Austin, "A systematic methodology to compute the architectural vulnerability factors for a high-performanc e microprocessor," in *International Symposium on Microarchitecture*, Dec. 2003.

[13] S. Krishnamohan and N. Mahapatra, "An efficient error masking technique for improving the soft-error robustness of static CMOS circuits," in *Proc. IEEE International System on Chip Conference*, Sept. 2004.

[14] T. Karnik, S. Vangal, V. Veeramachaneni, P. Hazucha, V. Erraguntla, and S. Borkar, "Selective node engineering for chip-level soft error rate improvement," in *Symposium on VLSI Circuits Digest of Technical Papers*, June 2002, pp. 204–205.

[15] K. Bernstein, "High speed CMOS logic responses to radiation-induced upsets," in *The Designing Robust Circuits and Systems with Unreliable Components Workshop*, 2002.

[16] J. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital integrated circuits*, 1st ed. Prentice Hall, 1996.

[17] J. Grad and J. E. Stine, "A standard cell library for student projects," in *International Conference on Microelectronic Systems Education*, 2003, pp. 98–99.

[18] K. Hass and J. Gambles, "Single event transients in deep submicron CMOS," in *Proc. Midwest Symposium on Circuits and Systems*, 1999.