Cost-Effective Approach for Reducing Soft Error Failure Rate in Logic Circuits

Kartik Mohanram* and Nur A. Touba
Computer Engineering Research Center
Department of Electrical and Computer Engineering
University of Texas at Austin, Austin, TX 78712-1084
E-mail: {kmram, touba}@ece.utexas.edu

Abstract

In this paper, a new paradigm for designing logic circuits with concurrent error detection (CED) is described. The key idea is to exploit the asymmetric soft error susceptibility of nodes in a logic circuit. Rather than target all modeled faults, CED is targeted towards the nodes that have the highest soft error susceptibility to achieve cost-effective tradeoffs between overhead and reduction in the soft error failure rate. Under this new paradigm, we present one particular approach that is based on partial duplication and show that it is capable of reducing the soft error failure rate significantly with a fraction of the overhead required for full duplication. A procedure for characterizing the soft error susceptibility of nodes in a logic circuit, and a heuristic procedure for selecting the set of nodes for partial duplication are described. A full set of experimental results demonstrate the cost-effective tradeoffs that can be achieved.

1. Introduction

When high-energy neutrons (present in terrestrial cosmic radiation) and alpha particles (that originate from impurities in the packaging materials) strike a sensitive region in a semiconductor device, they generate a dense local track of electron-hole pairs. This may be collected by a p-n junction resulting in a current pulse of very short duration termed a *single-event upset (SEU)* in the signal value. A SEU may cause a bit flip in some latch or memory element thereby altering the state of the system resulting in a *soft error*. Additionally, a SEU may occur in an internal node of combinational logic and subsequently propagate to and be captured in a latch. Soft errors in memories (both static and dynamic) have traditionally been a much greater concern than soft errors in logic circuits (for the same minimum feature size) since memories contain by far the largest number and density of bits susceptible to particle strikes. 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 SEUs and that this will result in unacceptable soft error failure rates even in mainstream commercial applications [Ziegler 96], [Cohen 99]. In a recent study, it has been projected that by 2011, the soft error rate in logic circuits will be comparable to that of unprotected memory elements [Shivakumar 02].

A system or component is said to *fail* if it does not correctly perform its intended function. Whether or not a soft error causes a component or system to fail depends on its fault tolerance features. If a soft error is not detected, then it can result in a failure. The *failure rate* for a component or system is the number of failures that occur per unit time. It is generally measured in units of FIT (1 failure in 10^9 hours of operation). Note that there may be other sources of failures in a system besides soft errors (e.g., permanent faults), however, this paper just focuses on the soft error failure rate (which dominates). All sources of failures are additive, so they can be considered independently.

One way to detect soft errors is to use concurrent error detection (CED) circuitry that monitors the outputs of a circuit for the occurrence of an error [Gössel 93], [Nicolaidis 98]. If an error is detected, then the system can recover thereby preventing a failure. The use of CED depends on the soft error failure rate requirements of the application. Consider two classes of applications: mission critical applications (e.g., traffic control, banking, medical, etc.) and mainstream applications. In mission critical applications, the primary objective is to achieve very high reliability with cost and performance as secondary concerns. In mainstream applications, cost and performance are the primary objectives. Traditionally for mainstream applications, soft error failure rates have been tolerable even without the use of CED. However, as the soft error failure rate for memories has increased, the use of CED for memories has become more common, e.g., parity and error correcting codes (ECC). CED for logic circuits in mainstream applications has seen limited use for two reasons:

1) Very high overhead (power, area, timing, etc.) – Unlike memories, logic circuits do not have a regular structure thereby making CED more complex.

[*] Kartik Mohanram is now with the Department of Electrical and Computer Engineering, Rice University, TX

0-7803-8106-8/03 $17.00 Copyright 2003 IEEE

Conventional techniques for CED were developed for mission critical applications where the goal is to detect all errors under a fault model. Some synthesis techniques have been developed for reducing the overhead without compromising on the error detection capability [Jha 93], [De 94], [Bolchini 97], [Touba 97], [Saposhnikov 98], [Das 99], [Zeng 99]. Some techniques based on multiple sampling of outputs have also been developed [Franco 94], [Metra 98], [Nicolaidis 99], [Favalli 02].

2) Lower susceptibility to soft errors – Logic circuits have a natural barrier to propagating SEUs to their output [Lidén 94]. When a particle strike occurs at an internal node of a logic circuit, there are three masking factors – *logical, electrical,* and *latching-window* – that may prevent it from being latched and resulting in a soft error (this is described in detail in Sec. 2). For larger process technologies, the soft error failure rate for logic circuits is much lower than for memories.

In the next decade, the soft error failure rate for logic circuits is projected to increase dramatically. Technology trends are causing the barriers for propagating SEUs to diminish significantly. These include smaller feature sizes, lower voltage levels, higher operating frequencies, reduced logic depth between latches, etc. (the effects of these will be described in detail in Sec. 2). As a result, reducing the soft error failure rate in logic circuits in mainstream applications is an important challenge for the future.

One approach for reducing the soft error failure rate in mainstream applications is to simply apply the CED techniques that have been developed for mission critical applications. However, these techniques are geared towards very high reliability and thus may be overkill. In the highly cost sensitive environment of mainstream applications, the goal is to reduce the soft error failure rate to acceptable levels at minimum cost. Thus, there is a need for a new class of CED techniques that span the middle ground between no protection/no overhead and very high protection/very high overhead. Whereas traditional CED techniques for mission critical applications target all modeled faults, the CED techniques for mainstream applications need to satisfy soft error failure rate reductions in a cost-effective manner.

In this paper, a new paradigm for designing logic circuits with CED is described. Rather than target all modeled faults, CED is targeted towards the nodes that have the highest soft error susceptibility, i.e., the nodes that contribute the most to the soft error failure rate of the logic circuit. This allows cost-effective tradeoffs between overhead and soft error failure rate reduction. Such techniques can be used in cost-sensitive mainstream applications to satisfy soft error failure rate requirements at minimum cost. Under this new paradigm, we present one particular approach that is based on partial duplication and show that it is capable of reducing the soft error failure rate significantly with a fraction of the overhead required for full duplication. The proposed technique scales very well for large circuits and is highly compatible with synthesis flows. While some results have shown parity prediction can achieve significant reductions in overhead versus duplication for arithmetic logic units (ALUs) or small circuits with few levels of logic, it does not scale well for large multilevel circuits where the parity functions are very hard to optimize with synthesis tools.

We address the issue of characterization of the soft error susceptibility of nodes in logic circuits and use the results of such a characterization to develop our low-cost CED methodology to reduce the soft error failure rate. We show that in the presence of the masking factors in logic circuits, SEUs at some internal nodes in logic circuits can have orders of magnitude greater probability of being latched and causing an error than at other nodes. By focusing CED towards the nodes that are most susceptible to SEUs, the soft error failure rate in logic circuits can be significantly reduced at a fraction of the cost of existing techniques that try to guarantee coverage of all nodes. We present an algorithm for the synthesis of CED circuitry for logic circuits based on partial duplication. The proposed algorithm achieves a very high reduction in the estimated soft error failure rate within the specified overhead constraints. We present experimental results that demonstrate the effectiveness of the proposed algorithm.

The rest of the paper is organized as follows. In Sec. 2, we describe the problem addressed in this paper in greater detail and review some previous research in this area. In Sec. 3, we present the methodology for estimating the soft error susceptibility of nodes in logic circuits. In Sec. 4, we present a methodology for the synthesis of low-cost CED circuitry for logic circuits based on partial duplication. In Sec. 5, we present experimental results. Section 6 is a conclusion.

2. Motivation and Previous Work

In this section, we discuss the factors that contribute to an increase in the soft error failure rate in logic circuits in greater detail. Specifically, we look at the masking factors and how present-day design trends are diminishing their significance leading to higher soft error failure rates. We then discuss the key idea in our work which is based upon exploiting the asymmetric soft error susceptibility of nodes in logic circuits. Lastly, we review previous work in the area of soft error failure rate estimation for integrated circuits.

2.1 Reasons for Increase in Soft Error Failure Rate of Logic Circuits

Soft errors in memories have traditionally been a much greater concern than soft errors in logic circuits. Memories are very susceptible to soft errors because of their small cell size and the fact that a SEU in a memory cell can immediately result in a soft error provided it exceeds a certain minimum critical charge required to flip the value stored in the cell [May 79], [Kirkpatrick 79].

Logic circuits have been much less susceptible to soft errors than memories. If a SEU occurs at an internal node of a logic circuit, there are three factors that may prevent it from being latched and resulting in a soft error:

1) There needs to be a functionally sensitized path from the location of the SEU to a latch. This will depend on what input vector is being applied at the time of the SEU. If there is not a sensitized path, then the SEU will be logically masked.

2) The SEU must create a pulse of significant duration and amplitude to propagate through each stage of logic until it reaches a latch. The pulse will be attenuated due to the electrical properties of each gate that it passes through, so the farther away from a latch the SEU occurs, the stronger it must be to make it to the latch. If a pulse is attenuated before it propagates (along a sensitized path) and reaches a latch, then the SEU will be electrically masked.

3) The timing of the SEU must be such that it causes a pulse that arrives at a latch just as the clock transitions so that the latch captures its value. If the SEU occurs at a time outside of the "latching-window", then it will not be captured in the latch and the SEU will be latching-window masked.

These three factors present a natural barrier to soft errors in logic circuits and have prevented soft errors in logic circuits from being a major concern [Lidén 94]. However, technology trends are causing these barriers to diminish significantly. The trend towards reduced logic depth between latches means that (1) there is less attenuation when propagating SEUs and (2) there is an increased number of sensitized paths. In addition, smaller feature sizes and lower voltage levels result in a reduction in the charged stored at a node. This allows lower energy particles to cause SEUs capable of being latched. Particles of lower energy occur much more frequently than particles of higher energy. An order of magnitude difference in energy can correspond to more than an order of magnitude larger flux for the lower energy particles. Faster gates allow SEUs of smaller pulse width to propagate through the circuit with minimum attenuation to the outputs. High operating frequencies mean that there are more latching-windows per unit time thereby increasing the probability of a SEU being latched.

Thus, as technology continues to scale, logic circuits are becoming much more susceptible to soft errors. Projections in [Shivakumar 02] indicate that for microprocessors that use ECC to reduce the soft error failure rate for memories, logic will become the dominant source of soft error failures. Thus, reducing the soft error failure rate for logic circuits is expected to emerge as a very important problem in the future.

2.2 Soft Error Susceptibility of Nodes in Logic Circuits

A key idea in this paper is to exploit the asymmetric soft error susceptibility of internal nodes in a logic circuit. While radiation bombards a chip fairly uniformly in space and time, the probability that a SEU is latched varies greatly depending on which node it occurs at in the logic circuit. The reasons for this include the following:

1) The percentage of time that each node is functionally sensitized to a latch depends on the logic function being implemented and the distribution of input vectors that are applied while the circuit operates. Often a small subset of the input vectors can be applied for a large percentage of the clock cycles resulting in certain nodes being sensitized to a latch much more frequently than others.

2) The size of the gate driving a node and the amount of capacitance at the node affects how much particle energy is required to create a SEU of sufficient strength to be latched. Particles with lower energy have much greater flux than those of higher energy, thus this can greatly skew the probability of a SEU of sufficient strength occurring at certain nodes.

3) The logic depth of a node from a latch affects how many gates a SEU has to propagate through to reach a latch and therefore how much attenuation will occur. The farther away a node is from a latch, the more particle energy is required to create a pulse of sufficient strength to be latched and thus the less likely it is to occur.

As a result of these factors, the soft error susceptibility of internal nodes in a logic circuit can vary by at least an order of magnitude. This provides an opportunity to significantly reduce the soft error failure rate at a reduced cost, since CED techniques can be targeted towards the nodes with high soft error susceptibility, while those with very low soft error susceptibility can essentially be ignored. This can be used to achieve a significant reduction in the soft error failure rate in a cost-effective manner.

Figure 1 shows the normalized soft error susceptibility distribution profile for six benchmark circuits calculated using the methodology that will be explained in Sec. 3. The soft error susceptibility of each node in the circuit is normalized with respect to the node with the largest soft error susceptibility. The x-axis shows increasing amount of soft error susceptibility from left to right, and the y-axis shows the number of nodes with that amount of susceptibility. These profiles illustrate the fact that certain nodes of the circuit have greater soft error susceptibility than others, and that a large number of nodes have negligible soft error susceptibility and can be effectively ignored when inserting fault tolerance features. These profiles were obtained assuming random input vectors, however, if real input traces from normal system workloads were used, the profiles would be even more skewed. Conventional approaches focus on protecting every node in the circuit, but more cost-effective approaches can be achieved by focusing only on the nodes with high soft error susceptibility.

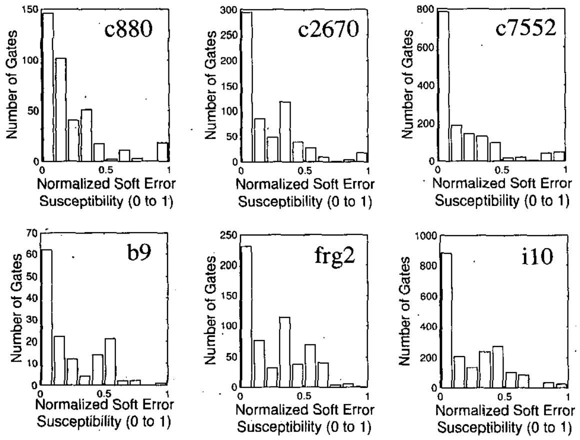

Figure 1. Soft error susceptibility profile for 6 benchmark circuits

2.3 Soft Error Modeling and Estimation Techniques

A key component to the proposed methodology is the capability to estimate the soft error susceptibility of nodes in a logic circuit and for the logic circuit as a whole. A lot of previous work has been done in this area. A comprehensive modeling program to predict the sensitivity of circuits to alpha-particles that was developed in [Sai-Halasz 82] formed the basis for many cosmic ray soft error rate modeling programs such as the Soft Error Monte Carlo Modeling Program (SEMM) [Murley 96]. A methodology to characterize the SEU rate in CMOS SRAM circuits (based on a verified empirical model for a 0.60 micron process technology) was presented in [Hazucha 00a]. A methodology to scale the results of this study to other feature sizes was presented in [Hazucha 00b]. There has been some previous work in the area of soft error susceptibility analysis of combinational and sequential circuits. A VHDL simulator that can inject and analyze soft faults in synthesized VHDL descriptions of synchronous logic circuits and the results of a study on a custom-designed bit-slice processor were presented in [Massengill 98], [Massengill 00]. An efficient transient fault injection and simulation technique that can be used to evaluate the soft error susceptibility of a design has been described in [Alexandrescu 02]. The effect of technology scaling and high-performance microprocessor design trends on the soft error rate in CMOS memory and logic circuits was presented in [Shivakumar 02]. The reader is referred to [Ziegler 96] for a comprehensive survey of the history of the study of soft errors in integrated circuits.

3. Proposed Soft Error Failure Rate Estimation Methodology

In this paper, we build on some of the previous research and develop a model that can be applied efficiently on a gate-level synthesized netlist of the design. After a design has been mapped to a standard cell library, each of the nodes (gates) in the netlist can be characterized individually to determine their soft error susceptibility. By then analyzing the interconnection of nodes (gates) in the netlist, the overall soft error susceptibility and the soft error failure rate of the design can be determined. Computing the soft error susceptibility for a node *n* with respect to latch *l* requires calculating three factors: (1) $R_{SEU}(n)$ – the rate at which a SEU of sufficient strength to change the logic value occurs at node *n*, (2) $P_{sensitized}(n,l)$ – the probability that node *n* is functionally sensitized to latch *l*, and (3) $P_{latched}(n,l)$ – the probability that the SEU at node *n* is captured in latch *l*. The soft error susceptibility for node *n* with respect to latch *l* (the rate at which soft errors are generated at latch *l* due to SEUs at node *n*) is the product of these three factors:

$$\text{Soft error susceptibility of node } n \text{ with respect to latch } l = R_{SEU}(n) \bullet P_{sensitized}(n,l) \bullet P_{latched}(n,l)$$

The calculation of each of these three factors is discussed below:

1) $R_{SEU}(n)$, the rate at which a SEU of sufficient strength to change the logic value occurs at node *n:*

This depends on the device characteristics of the gate driving node *n*, the amount of capacitance at node *n*, as well as the sensitive area of node *n*. Two methods of calculating $R_{SEU}(n)$ in avionics were presented in [Normand 96]. Both methods are directly applicable to SEU rate calculations under terrestrial conditions when the variation in neutron flux at ground-level is taken into consideration. The first method, called the neutron cross-section (NCS) method, uses the neutron/proton SEU cross-section measured for the device, while the second method uses heavy-ion SEU data via the burst-generation-rate method proposed in [Ziegler 79] to calculate the SEU rate for the device. We follow the NCS method in this paper. The neutron SEU cross-section is defined as the probability that a neutron of energy E_x can produce an upset in a device in units of cm^2/device. The SEU rate for a node $R_{SEU}(n)$ using the NCS method is given by:

$$R_{SEU}(n) = \int_{E_{\min}}^{\infty} \sigma_{nSEU}(E_x)\left(\frac{dN}{dE_x}\right) dE_x$$

where $\sigma_{nSEU}(E_x)$ is the neutron SEU cross-section of the device and (dN/dE_x) is the differential neutron flux (note that the flux varies depending on altitude, latitude, etc.). For a logic circuit, it is possible to obtain $\sigma_{nSEU}(E_x)$ for node n by characterizing the cell library. The integration is then done for all particle energies greater than the minimum particle energy E_{min} needed to create a voltage pulse of sufficient strength to change the logic value.

The minimum energy, E_{min}, can be determined using the charge to voltage pulse model developed in [Freeman 96]. The accuracy of the calculation of E_{min} will depend on the amount of design information available. If layout information is available, then a very accurate measure of E_{min} can be obtained from SPICE simulations where waveforms corresponding to increasing particle energies are injected at node n until it is sufficient to change the logic value. If only a technology mapped netlist is available, then the library cell that is used in the netlist can be characterized in SPICE. We use the following approximation to arrive at E_{min}. The critical charge Q_c at a node is determined using SPICE simulations. Increasing amounts of charge – modeled by current pulses of increasing magnitude and duration – are inserted till the output of the node changes. Once the critical charge is determined in this manner, we assume that all the energy deposited by the particle was used to generate the charge. Thus the critical charge Q_c is formed by deposition in the critical volume of energy E_{min} given by

$$E_{min} = \frac{3.6\text{eV} \bullet Q_c}{1.6 \bullet 10^{-19}} \text{C}$$

where 3.6eV is the energy required to generate an electron-hole pair in silicon.

For the differential neutron flux (dN/dE_x), we use the analytic approximation of the differential neutron flux in New York City over the 1MeV to 10000MeV range [Bradley 98]. We then use a piecewise summation of the above integral formulation of the NCS method in intervals of 10MeV starting from E_{min} to obtain $R_{SEU}(n)$.

2) $P_{sensitized}(n,l)$, the probability that node n is functionally sensitized to latch l:

Whether or not node n is sensitized to latch l depends on the input pattern being applied. Thus, the probability that node n is sensitized to latch l depends on the probability of each input pattern being applied to the circuit while it is operating. A fast and efficient way to calculate $P_{sensitized}(n,l)$ is to simply simulate the system with a typical workload for some number of clock cycles. For each clock cycle, critical path tracing [Abramovici 83] can be performed starting from each latch to identify all of the nodes that are sensitized to it (alternatively fault injection and simulation can be used for the same purpose). Note that there can be some nodes in the circuit which are only sensitized to a latch for very few input patterns and hence may not get sensitized at all during the simulation. However, these nodes will have a negligible affect on the overall soft error rate (since their probability of being sensitized is extremely low) and hence can be ignored. A less accurate alternative to simulating the system with a typical workload would be to just apply random patterns at the primary inputs to get a rough estimate.

3) $P_{latched}(n,l)$, the probability that the SEU is captured in latch l:

In order to be captured in latch l, the pulse created by the SEU must arrive at the latch during the latching-window in time. The probability of the pulse being present during the latching-window depends on the width of the pulse relative to the clock period. The width of the pulse depends on the amount of particle energy. By taking attenuation through the propagation path into consideration (through a characterization of the library cells for different voltage pulse inputs), the width of the pulse as it reaches the latch can be determined for different particle energies. By comparing this with the total clock period, the probability of the pulse being latched can be computed.

4. Concurrent Error Detection

Conventional schemes to design circuits with CED based on error-detecting codes such as parity, duplication and compare, etc. employ checkers to monitor the outputs for the occurrence of an error. Figure 2 shows the structure of a circuit that has CED capability. Based upon the scheme chosen for CED, the check symbol generator can be a copy of the original circuit (duplication and compare), parity prediction logic, codeword generator (e.g., for Berger or Bose-Lin codes), etc. The check symbol generator generates check bits and the checker determines if they form a codeword.

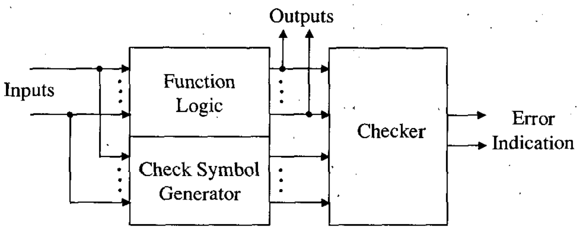

Figure 2. Block diagram for conventional CED

4.1 Intuition for the Partial Duplication Method

The proposed method is based upon the observation that in the presence of the three masking factors described in Sec. 2.2, the soft error susceptibility of certain nodes in the logic circuit can be orders of magnitude higher than that of the other nodes in the design. The second observation is that these nodes tend to be located closer to the primary outputs. Thus, nodes that are several levels of logic from the primary outputs have a comparatively lower soft error susceptibility than nodes close to the primary outputs. Thus, the proposed heuristic involves selecting a cluster of nodes (henceforth the "cutset") near the primary outputs whose logic is duplicated as shown in Fig. 3.

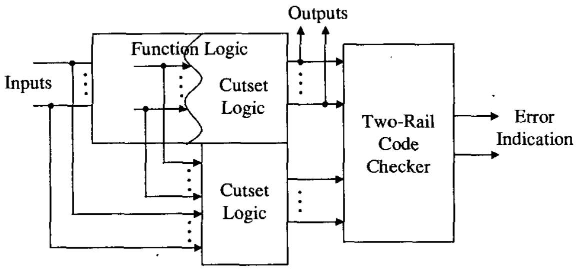

Figure 3. CED with partial duplication

The inputs to the duplicated logic of the cutset are taken from the design. Thus, if a particle strike occurs in the non-duplicated portion of the design and is of sufficient strength, the resulting SEU will (in the presence of a sensitized path) propagate to the outputs of both the cutset and the function logic and go undetected. However, any SEU from a particle strike that occurs at a node in the partially duplicated portion of the circuit (in either the function logic or the cutset) will be detected at the outputs of the checker. Note that if the path from the node where the SEU occurs to the outputs contains nodes that are not duplicated, the SEU propagates to both sets of outputs and is hence not detected (an example is provided in Sec. 4.3). By carefully selecting the cutset, the nodes with the highest soft error susceptibility will be in the partially duplicated portion of the circuit thereby giving a very cost-effective reduction in the soft error failure rate.

4.2 Algorithm for Partial Duplication

A heuristic algorithm for partial duplication that generates a cutset with specified area overhead is described below. The basic idea of the heuristic is to traverse the circuit from the primary outputs to the primary inputs in a greedy manner to generate the cutset. A priority queue (indexed by soft error susceptibility) of nodes that can be added to the current cutset of nodes in a consistent manner is maintained. At each iteration of the heuristic, the node *n* with the highest soft error susceptibility is removed from the head of the priority queue and added to the cutset. The current area overhead is updated to reflect the latest addition to the cutset and all gates that are inputs to node *n* are added to the priority queue. This process terminates when the size of the cutset equals (or just exceeds) the specified area overhead. We describe the steps of the heuristic using the following example. In Fig. 4, we present a screenshot of a small circuit where the gates that have been selected for partial duplication by the proposed algorithm have been highlighted. The soft error susceptibility for each of the gates in the design (to 2 significant digits) is also provided. The only constraint to the proposed algorithm is the overhead that is allowed for partial duplication. The following are the steps of the algorithm.

Step 1 – A priority queue *gateQ* that is indexed by the soft error susceptibility of nodes is initialized. All the primary outputs of the circuit (G_1 to G_7 in Fig. 5) are inserted into *gateQ*. The heuristic processes gates starting from the primary outputs for two reasons. The first is based upon the observation that the primary outputs have a very high soft error susceptibility since they are always sensitized. The second is that checking is performed on the primary outputs. As a result, the cutset has to always be "consistent" – all gates selected for partial duplication in the cutset must have at least one path to the primary outputs that is entirely contained within the cutset. Thus for a gate to be added to the cutset, at least one of the gates that it fans out to must already be a part of the cutset (and hence, by induction, have at least one path entirely contained in the cutset to the primary outputs). Adding all primary outputs to *gateQ* during initialization ensures that the cutset is generated in a consistent manner. For the example, at the end of the initialization procedure, node G_7 with a normalized soft error susceptibility of 10 is at the head of the priority queue.

Step 2 – The pseudo-code for the iterative process of growing the cutset is described in Fig. 5. The first two passes of the iterative phase of the heuristic are as follows. Gate G_7 is popped and marked as a node that belongs to the cutset. The cost of the cutset at this point is equal to the area of the gate G_7. Since G_7 is driven only by the primary inputs, there are no gates in its immediate fanin that need to be processed.

```
/* netlist – technology mapped design with soft error susceptibility data
   overhead – area overhead constraint ( overhead < area ( netlist ) )
   gateQ – priority queue initialized with all primary outputs */

while ( ( is_not_empty ( gateQ ) ) || ( current_cost < overhead ) ) {
    node = top ( gateQ ) ;
    pop ( gateQ ) ;
    mark ( node ) ;
    current_cost += area ( node ) ;
    for_each_fanin ( node , fanin ) {
        if_not_marked ( fanin ) {
            insert ( gateQ , fanin , soft_error_susceptibility ( fanin ) ) ;
        }
    }
}
```

Figure 5. Pseudo-code for iterative phase

The next gate that is popped from *gateQ* is G_6. Note that if there are several gates with the same soft error

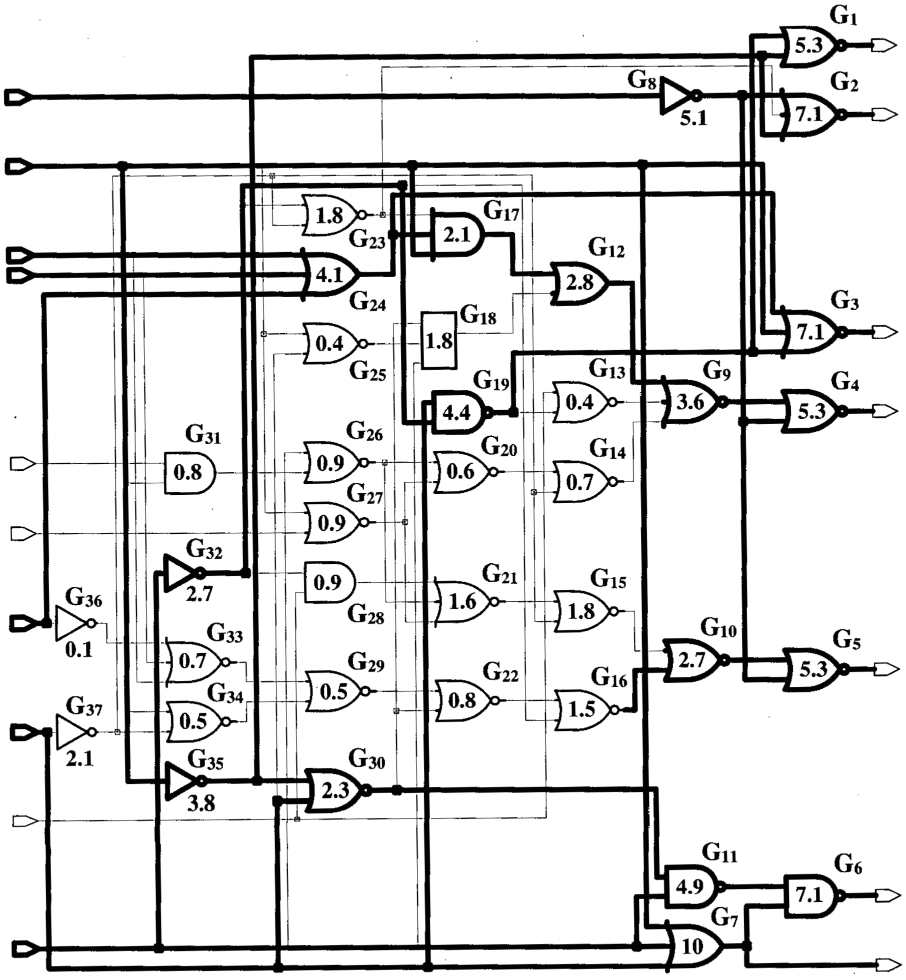

Figure 4. Screen shot with gates selected for partial duplication

susceptibility, the ones with the smallest area are processed first. Of the gates in G_6's immediate fanin list, G_7 has already been added to the cutset. G_{10} is inserted into *gateQ*. In this manner, the cutset grows until the cost of all the gates added to the cutset exceeds the specified area overhead constraint.

Coverage estimation is run to see if the reduction in the soft error failure rate meets requirements. Care has to be taken during soft error failure rate estimation to ensure that a SEU that is propagated along a path that does not entirely lie within the cutset is accounted for. An example of such a case is a SEU that propagates along the path ($G_{19} \rightarrow G_{13} \rightarrow G_9 \rightarrow G_4$).

It may also be the case that a buffer or an inverter is driven by a node of considerably higher soft error susceptibility. In order to ensure that this is taken into consideration whenever a node's immediate fanin is inserted into *gateQ*, a buffer (or inverter) is indexed by the sum of the soft error susceptibilities of the buffer (or inverter) and the node that drives the buffer.

5. Experimental Results

The synthesis tool used for all technology mapping and optimization in this paper was Synopsys' Design Analyzer. The technology library used is the 0.25 micron library distributed by Virginia Tech [Sulistyo 02]. The combinational benchmark circuits were chosen from the LGSynth91 suite [Yang 91]. A framework for the soft error failure rate estimation methodology described in Sec. 3 was implemented in C++.

Table 1 presents the reductions in the soft error failure rate that we achieved using the proposed partial duplication scheme. Under the first major heading, we provide details about the circuits that were chosen – name, number of primary inputs, and number of primary outputs.

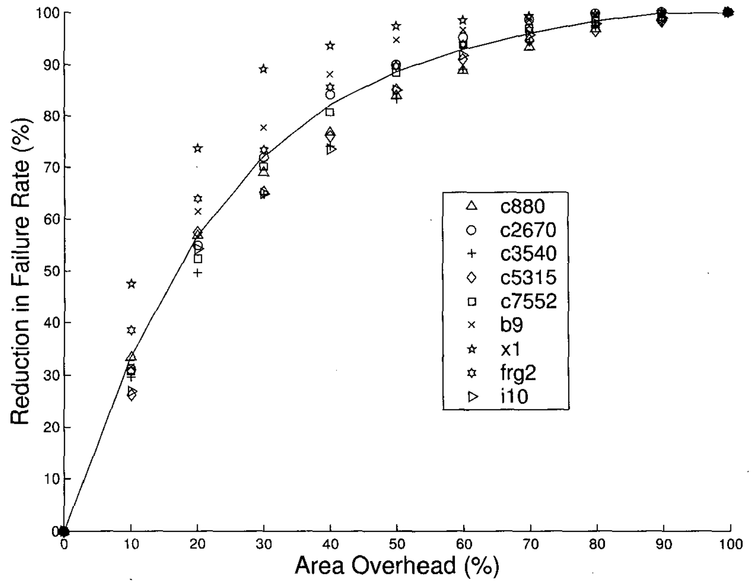

Figure 6. Reduction in soft error failure rate (%) for 9 benchmark circuits

Under the second major heading, we present the reduction in the soft error failure rate that was observed when the area overhead constraint for CED is 20%, 33%, and 50% respectively. The soft error failure rate reduction percentage was computed as:

$$\left(\frac{\text{Original Failure Rate - Reduced Failure Rate}}{\text{Original Failure Rate}}\right)\times 100\ \%$$

Table 1. Soft error failure rate reduction using partial duplication

Sof Error Failure Rate Reduction (%)					
Circuit			Area Overhead		
Name	No. PIs	No. POs	20%	33%	50%
C2670	233	140	53.9	77.7	89.5
C3540	50	22	49.9	68.6	83.1
C5315	178	123	57.3	68.5	83.8
C7552	207	108	52.3	73.6	88.7
x1	51	35	73.9	90.8	95.5
c880	60	26	56.1	73.3	83.6
b9	41	21	59.8	87.9	95.0
i10	257	224	54.3	68.8	85.1
frg2	143	139	65.4	78.0	90.5
Average Reduction			58.1	76.3	88.3

The last row presents the average reduction in the soft error failure rate that is observed using the proposed scheme. Note that an order of magnitude reduction in the soft error failure rate can generally be achieved with a 50-60% overhead. A factor of 4 reduction can generally be achieved with 33% overhead, and more than a factor of 2 reduction can be achieved with 20% overhead.

In Fig. 6, we present a graph of the reduction in soft error failure rate that is achieved versus the allowed area overhead for all 9 benchmark circuits. The average of the reductions over all the benchmark circuits versus area overhead is provided by a continuous curve in the figure. Depending on the soft error failure rate requirements for a particular application, the appropriate point on this curve can be selected.

6. Conclusions

In the future, as the soft error failure rate of logic circuits becomes unacceptably high even for mainstream applications, CED will become necessary for logic circuits. This paper described a promising new paradigm for designing logic circuits with cost-effective CED by exploiting the asymmetric soft error susceptibilities of nodes. The partial duplication approach described here is one particular approach for accomplishing this. An area for future research is to investigate other techniques that can be selectively targeted towards the most susceptible nodes.

Acknowledgements

The authors would like to acknowledge the contributions to this work by Michael Gössel and Egor S. Sogomonyan from the University of Potsdam. This research was supported by a grant from the Hewlett-Packard Company.

References

[Abramovici 83] M. Abromovici, P. R. Menon, and D. T. Miller, "Critical Path Tracing – An Alternative to Fault Simulation", *Proc. Design Automation Conference*, pp. 214-220, 1983.

[Alexandrescu 02] D. Alexandrescu, L. Anghel, and M. Nicolaidis, "New Methods for Evaluating the Impact of Single Event Transients in VDSM ICs," *Proc. Defect and Fault Tolerance Symposium*, pp. 99-107, 2002.

[Bolchini 97] C. Bolchini, F. Salice, and D. Sciuto, "A Novel Methodology for Designing TSC Networks based on the Parity Bit Code," *Proc. European Design and Test Conference*, pp. 440-444, 1997.

[Bradley 98] P. D. Bradley and E. Normand, "Single Even Upsets in Implantable Cardioverter Defibrillators," *IEEE Trans. Nuclear Science*, Vol. 45, No. 6, pp. 2929-2940, Dec. 1998.

[Cohen 99] N. Cohen, *et al.*, "Soft Error Considerations for Deep-Submicron CMOS Circuit Applications," *International Electron Devices Meeting*, 1999.

[Das 99] D. Das and N. A. Touba, "Synthesis of Circuits with Low-Cost Concurrent Error Detection Based on Bose-Lin Codes," *Journal of Electronic Testing: Theory and Applications*, Vol. 15, Nos. 1/2, pp. 145-155, Aug. 1999.

[De 94] K. De, *et al.*, "RSYN: A System for Automated Synthesis of Reliable Multilevel Circuits," *IEEE Trans. VLSI Systems*, pp. 186-195, Jun. 1994.

[Favalli 02] M. Favalli and C. Metra, "Online Testing Approach for Very Deep-Submicron ICs," *IEEE Design and Test of Computers*, Vol. 19, No. 2, pp. 16-23, Mar. 2002.

[Franco 94] P. Franco and E. J. McCluskey, "On Line Delay Testing of Digital Circuits," *Proc. VLSI Test Symposium*, pp. 167-173, 1994.

[Freeman 96] L. B. Freeman, "Critical Charge Calculations for a Bipolar SRAM Array," *IBM Journal Research and Development*, Vol. 40, No. 1, pp. 119-129, Jan. 1996.

[Gössel 93] M. Gössel and S. Graf, *Error Detection Circuits*, McGraw-Hill Book Company, London, 1993.

[Hazucha 00a] P. Hazucha and C. Svensson, "Cosmic Ray Neutron Multiple-Upset Measurements in a 0.6-/spl mu/m CMOS Process," *IEEE Trans. Nuclear Science*, Vol. 47, No. 6, pp. 2595-2602, Jul. 2000.

[Hazucha 00b] P. Hazucha and C. Svensson, "Impact of CMOS Technology Scaling on the Atmospheric Neutron Soft Error Rate," *IEEE Trans. Nuclear Science*, Vol. 47, No. 6, pp. 2586-2594, Dec. 2000.

[Jha 93] N. K. Jha and S. Wang, "Design and Synthesis of Self-Checking VLSI Circuits," *IEEE Trans. Computer-Aided Design*, Vol. 12, No. 6, pp. 878-887, Jun. 1993.

[Kirkpatrick 79] S. Kirkpatrick, "Modeling Diffusion and Collection of Charge from Ionizing Radiation in Silicon Devices," *IEEE Trans. Electron Devices*, No. 11, pp. 1742-1753, Nov. 1979.

[Lidén 94] P. Lidén, *et al.*, "On Latching Probability of Particle Induced Transients in Combinational Networks," *Proc. Fault-Tolerant Computing Symposium*, pp. 340-349, 1994.

[Massengill 98] L. W. Massengill, *et al.*, "Single-Event Upset Cross-Section Modeling in Combinational CMOS Logic Circuits," *Journal of Radiation Effects, Research, and Engineering*, Vol. 16, No. 1, Jan. 1998.

[Massengill 00] L. W. Massengill, *et al.*, "Analysis of Single-Event Effects in Combinational Logic-Simulation of the AM2901 Bitslice Processor", *IEEE Trans. Nuclear Science*, Vol. 47, No. 6, pp. 2609-2615, Dec. 2000

[May 79] T. C. May and M. H. Woods, "Alpha-Particle-Induced Soft Errors in Dynamic Memories," *IEEE Trans. Electron Devices*, Vol. 26, No. 1, pp. 2-9, Jan. 1979.

[Metra 98] C. Metra, M. Favalli, and B. Ricco, "Online Detection of Logic Errors Due to Crosstalk, Delay and Transient Faults," *Proc. International Test Conference*, pp. 524-533, 1998.

[Murley 96] P. C. Murley and G. R. Srinivasan, "Soft-Error Monte Carlo Modeling Program, SEMM," *IBM Journal of Research and Development*, Vol. 40, No. 1, pp. 109-118, 1996.

[Nicolaidis 98] M. Nicolaidis and Y. Zorian, "Online Testing for VLSI – A Compendium of Approaches," *Journal of Electronic Testing: Theory and Applications*, Vol. 12, Nos. 1/2, pp. 7-20, Feb. 1998.

[Nicolaidis 99] M. Nicolaidis, "Time Redundancy Based Soft-Error Tolerance to Rescue Nanometer Technologies," *Proc. of VLSI Test Symposium*, pp. 86-94, 1999.

[Normand 96] E. Normand, "Single-Event Effects in Avionics," *IEEE Trans. Nuclear Science*, Vol. 43, No. 2, pp. 461-474, Apr. 1996.

[Sai-Halasz 82] G. A. Sai-Halasz, M. R. Wordeman, and R. H. Dennard, "Alpha-Particle-Induced Soft Error Rate in VLSI Circuits," *IEEE Journal of Solid-State Circuits*, Vol. 17, No. 2, pp. 355-361, Apr. 1982.

[Saposhnikov 98] V. V. Saposhnikov, *et al.*, "A New Design Methodology for Self-Checking Unidirectional Combinational Circuits," *Journal on Electronic Testing: Theory and Applications*, Vol. 12, Nos. 1/2, pp. 41-53, Feb. 1998.

[Shivakumar 02] P. Shivakumar, *et al.*, "Modeling the Effect of Technology Trends on the Soft Error Rate of Combinational Logic," *Proc. International Conference on Dependable Systems and Networks*, pp. 389-398, 2002.

[Sulistyo 02] J. B. Sulistyo and D. S. Ha, "Developing Standard Cells for TSMC 0.25μ Technology under MOSIS DEEP Rules", Department of Electrical and Computer Engineering, Virginia Tech, TR VISC-2002-01, Jan. 2002.

[Touba 97] N. A. Touba and E. J. McCluskey, "Logic Synthesis of Multilevel Circuits with Concurrent Error Detection," *IEEE Trans. on Computer-Aided Design*, Vol. 16, No. 7, pp. 783-789, Jul. 1997.

[Yang 91] S. Yang, "Logic Synthesis and Optimization Benchmarks User Guide," Technical Report 1991-IWLS-UG-Saeyang, MCNC, Research Triangle Park, NC, Jan. 1991.

[Zeng 99] C. Zeng and E. J. McCluskey, "Finite State Machine Synthesis with Concurrent Error Detection", *Proc. International Test Conference*, pp. 672-679, 1999.

[Ziegler 79] J. F. Ziegler and W. A. Lanford, "Effect of Cosmic Rays on Computer Memories," *Science*, Vol. 206, pp. 776-788, Nov. 1979.

[Ziegler 96] J. F. Ziegler, *et al.*, "IBM Experiments in Soft Fails in Computer Electronics (1978-1994)," *IBM Journal of Research and Development*, Vol. 40, pp. 3-18, 1996.