Comprehensive Analysis of Alpha and Neutron Particle-induced Soft Errors in an Embedded Processor at Nanoscales

Mojtaba Ebrahimi[†], Adrian Evans[‡], Mehdi B. Tahoori[†], Razi Seyyedi[†], Enrico Costenaro[‡], Dan Alexandrescu[‡]

[†]Karlsruhe Institute of Technology, Karlsruhe, Germany {mojtaba.ebrahimi, mehdi.tahoori}@kit.edu, seyyedi@ira.uka.de

[‡]iROC Technologies, Grenoble, France {dan, adrian, enrico}@iroctech.com

Abstract—Radiation-induced soft errors have become a key challenge in advanced commercial electronic components and systems. We present results of Soft Error Rate (SER) analysis of an embedded processor. Our SER analysis platform accurately models all generation, propagation and masking effects starting from a technology response model derived using TCAD simulations at the device level all the way to application masking. The platform employs a combination of empirical models at the device level, analytical error propagation at logic level and fault emulation at the architecture/application level to provide the detailed contribution of each component (flip-flops, combinational gates, and SRAMs) to the overall SER. At each stage in the modeling hierarchy, an appropriate level of abstraction is used to propagate the effect of errors to the next higher level. Unlike previous studies which are based on very simple test chips, analyzing the entire processor gives more insight into the contributions of different components to the overall SER. The results of this analysis can assist circuit designers to adopt effective hardening techniques to reduce the overall SER while meeting required power and performance constraints.

I. INTRODUCTION

A gradual reduction in operating voltage combined with the exponentially growing device count per chip lead to an increase in the effective Soft Error Rate (SER) of complex designs over the past few years [1, 2]. In previous technology nodes, SRAMs and flip-flops were known to be the dominant contributors to the overall SER [2, 4]. In fact, despite the comparatively large number of combinational gates, due to high degree of electrical attenuation and low probability of latching in flip-flops, their overall SER contribution has been negligible [3, 4].

Existing work for investigating the contribution of sequential and combinational SER has been mostly based on the results of radiation testing which use simple test structures such as inverter chains, comparators, and shift-registers [5–10]. Such experiments reveal that the contribution of combinational logic is remarkable and has a linear relation with the circuit frequency. The major issue with these experiments is the simplicity of the test structures. Although such experiments give insight into the relative SER of different components, as we will show later, they are not representative of complex circuits. In fact, in complex circuits, due to the complexity of error propagation scenarios between memory units and logic components [11], identification of the error origin in radiation experiments is a challenging issue.

Another issue is that the results of simulations and radiation testing experiments performed in nanoscale technology nodes show that a single strike might affect multiple nodes in both memory units and logic blocks [12–14]. Additionally, as detailed in [15], in advanced technologies due to the shared diffusion and multiple node disruption, the SER of hardened flip-flops is comparable (30%-50%) with that of unprotected ones. Therefore, unlike previous technology nodes, using such kind of flip-flops in test chips cannot completely suppress the SER of flip-flops. An alternative solution exploited in [6, 9] is to use two types of shift-chains with and without intermediate combinational logic to compute logic SER by subtracting the SER of latter case from that of the former one. In this case, it is assumed that SERs of flip-flops are almost the same in both circuits while because of the difference in *temporal masking* of flip-flops in the two structures, the difference in observed error rate is not solely attributable to the combinational gates. As a conclusion, accurate isolation of the contribution of flip-flops and combinational logic during radiation-testing experiments is still a challenging issue.

Considering these issues, it is clear that radiation testing of simple test chips does not provide the full picture. The main objective of this work is to investigate the contribution of combinational, flip-flops, and SRAM elements to alpha and neutron particle-induced transients in an embedded processor designed in a nanoscale technology node operating in a terrestrial environment. This is achieved using an industrial SER analysis tool which combines limited TCAD simulation with cell-level SPICE simulation, an analytical circuit-level error propagation technique augmented with an emulation-based approach for fast and rigorous fault injection. This platform is able to perform full system SER analysis considering all masking factors in a hierarchical way and pass appropriate information among different parts of the platform. The accuracy of the industrial SER analysis platform has been verified by the radiation testing data using commercial process technologies as well as accurate statistical fault injection at SPICE-level.

The analysis of the OpenRISC 1200 (OR1200) processor shows that as with most digital circuits, after interleaving and *Single Error Correction* (SEC) code are applied to the SRAMs, the flip-flops become the major contributor to the overall SER. Additionally, it is shown that despite the claim of [5, 6] in which flip-flop SER is independent of clock frequency, it has an inverse linear relation with clock frequency which is due to the temporal masking of flip-flops. Since the decrease in flip-flop SER due to temporal masking is more significant than the increase in combinational SER due to reduced latch-window masking, the overall SER has an inverse linear relation with the clock frequency. Also, our analysis shows that the contribution of the clock-tree network to the overall SER is negligible whereas the reset-tree SER is considerable. Previous studies in [7, 8] reported that the clock-tree contribution could be as large as 20% of overall flip-flops SER, however, because of large size clock drivers, the probability of having soft errors in the clock-tree network is negligible. Additionally, the threshold frequency at which the combinational logic SER approaches the flip-flop SER is estimated to be in the range of the typical frequency of current commercial high performance processors.

The results presented in this paper are for one processor implemented using Nangate 45 nm library, running particular workloads. Under different conditions, the results may vary, however, we believe that most of the major trends and the conclusions hold across different designs and processes which can provide useful insights to designers for cost-effective protection of embedded processors.

The rest of this paper organized as follows: The employed SER analysis framework is detailed in Section II followed by experimental setup and results in Section III and IV, respectively. Finally, Section V summarizes the results.

II. SER ANALYSIS FRAMEWORK

The framework for SER analysis consists of three main steps as depicted in Figure 1. First, all the cells were analyzed using a commercial SER analysis tool to obtain the raw FIT (Fault in Time) rates. Then, by employing an accurate analytical model considering electrical, logical, and latching-window masking, errors are propagated from combinational logic towards flip-flops and memory units. Temporal masking of each flip-flop is determined using *Static Timing Analysis* (STA) based on the post-layout timing data. Then, an emulation-based tool is employed to inject errors at flip-flops and SRAM arrays. This tool is able to propagate errors and determine their effect (masked, latent, failure) with respect to an actual workload. Finally, the SERs of all components are computed based on the results obtained in these three steps.

A. FIT Rate Analysis Tool

The first step in the flow is to characterize the intrinsic FIT rate of each library cell under multiple load conditions. This is done using a commercial FIT rate analysis tool [16]. The tool uses a process response model which is generated based on a set of TCAD simulations and models the radiation sensitivity of the intrinsic process as well as the SPICE netlist and the layout of the cell. The accuracy of this tool for raw FIT rate analysis has been validated on several commercial processes.

This tool accepts the layout and SPICE netlist of the cell as well as information about the radiative environment as input. The tool also accesses a nuclear database which enables it to compute the interactions between neutrons and the atoms in the device, the secondary particles which are produced, and the distribution of generated current pulses. For each transistor, these current pulses are injected into a SPICE netlist of the cell in order to determine the effect. For sequential and SRAM cells, the probability of an upset is computed as the FIT rate.

For combinational cells, this tool provides the distribution of pulses of various widths. This distribution is dependent on the output load seen by the cell, thus each cell is characterized with multiple loads. During the full circuit analysis, for each gate, based on its load, an interpolation is done between the two closest load conditions that were simulated for the given cell type. In order to accelerate the analysis flow in the next phases, the pulse width distribution curve is discretized into a few pulse widths with distance of d (see Figure 1). Smaller values of d result in higher accuracy but larger runtime. Therefore, an appropriate value for this variable must be selected considering the trade-off between runtime and accuracy. The FIT rate of a combinational gate g with load l is denoted as $FIT^g(l, w)$ which represents the rate of having transient pulses with widths between $(w - \frac{d}{2}, w + \frac{d}{2})$.

In the case of flip-flops and latches, this tool reports the *Single Event Upset* (SEU) rate as FIT^{FF}. For single-port and multi-port SRAMs, the tool provides SEU and *Multiple Bit Upset* (MBU) rates as well as MBU patterns and their occurrence probability[1]. For each SRAM unit, the FIT rate for the pattern p is donated as $FIT^{SRAM}(p)$.

B. Single-cycle Analysis

Error propagation from combinational gates and flip-flops in the first cycle is dependent on the electrical and timing characteristics of the propagation paths and requires a detailed circuit-level analysis.

1) Single-cycle Error Propagation from Combinational Gates: A transient pulse in combinational gates is only latched in flip-flops if it is not masked by any of the logical (where another input of a gate is in controlling state), electrical (pulse electrically attenuated during propagation), and latching-window masking (pulse is not captured in the flip-flops as it is not arrived in the latching-window of receiving elements) factors [3].

Since the electrical and latching-window masking behavior of combinational logic is completely different in an emulation environment [17], SPICE-based fault injection is the only accurate approach for analyzing SER of these elements. However, as the number of combinational gates is much more than flip-flops and also the error propagation should be repeated for different pulse widths, reaching a reasonable level of accuracy with such fault injections is intractable even for a small circuit with a few thousand gates.

In this regards, we employed a combination of previously proposed analytical techniques to propagate the errors in the combinational logic in the first cycle. The techniques presented in [18], [19], and [20] have been used for modelling logical, electrical, and latching-window masking factors, respectively. Comparison of the employed analytical techniques with SPICE-based fault injection for 100 randomly selected combinational gates from the OR1200 processor shows average inaccuracy of 2.1±0.6% while offering five orders of

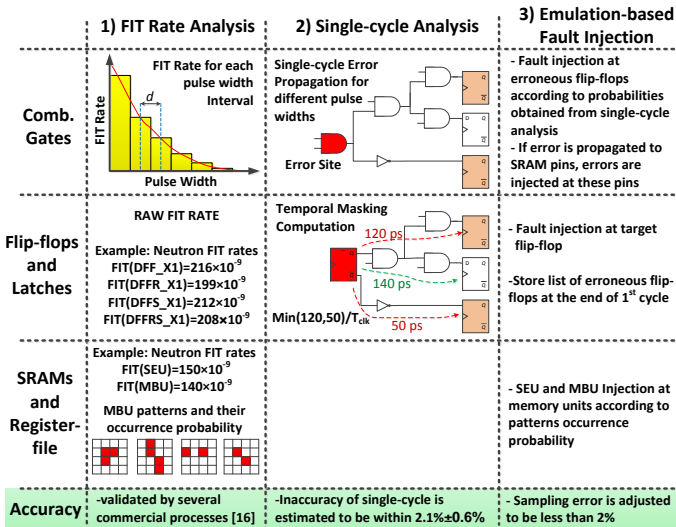Fig. 1. Overall Flow of Industrial SER Analysis Framework

---

[1]For sake of simplicity, in the rest of this paper, SEU in SRAMs is assumed as an MBU pattern with one bit-flip

magnitude speedup compared to SPICE simulations.

The single-cycle analysis should be repeated for different pulse widths. At the end of single-cycle analysis for pulse width w, we obtain $SCPP^g(w,s)$ which is the probability of propagation of the transient pulse from gate g to all flip-flops in set s.

2) Temporal Masking Computation for Flip-flops: Temporal masking in flip-flops happens when an SEU occurs sufficiently late in the clock cycle and is not able to propagate to the inputs of the downstream flip-flops before the clock edge [21]. A conventional approach for estimating the amount of temporal masking is to compute the delay of the shortest path to downstream flip-flops using STA. In order to be more realistic, we have computed the delay of the shortest path to erroneous downstream flip-flops rather than all flip-flops. In this regard, using STA, the shortest path delay from each flip-flop to its downstream flip-flops is computed. This information will be combined with the set of erroneous flip-flops extracted from the emulation-based error injection in the next step to compute the temporal masking of each flip-flop.

C. Emulation-based Error Propagation

Once we obtain how the errors propagate to flip-flops and SRAM cells, we perform an emulation-based fault injection on those cells to extract the effect of architecture and application masking. Error sites for evaluation includes SEU in each flip-flop, SEU/MBU in each SRAM unit, and MBU in flip-flop sets extracted from the single-cycle analysis during combinational logic evaluation.

A fast and flexible emulation-based platform similar to [22] is deployed to inject SEU and MBU errors in flip-flops and SRAMs to observe their effect on workload output stored in the main memory. This platform relies on Altera FPGAs debugging facilities for error injection. In our experiments, one error (SEU or MBU) is injected in a random clock cycle during a workload execution at the target error site(s). To do this, as depicted in Figure 2, after initializing memory units and flip-flops in the processor, the workload is emulated until the error injection time and then after injecting the error by flipping the value of the error site(s), emulation continues until the end of the workload. At the end, error classification is done by comparing the system state with that of the golden run. In case of error injection in flip-flops, at the end of the first error emulation cycle, the list of affected flip-flops is extracted. This is required for more accurate temporal masking computation as explained in the previous subsection.

By analyzing the results from this platform, we are able to extract the probability of error propagation from some set of flip-flops to the application output (denoted as $EPP^{FF}(s)$, s is the set of flip-flops) and also propagation probability of a certain SEU/MBU pattern to the workload output (denoted as $EPP^{SRAM}(p)$ where p is the pattern). In addition, we can compute the probability that error in a specific flip-flop f is propagated to w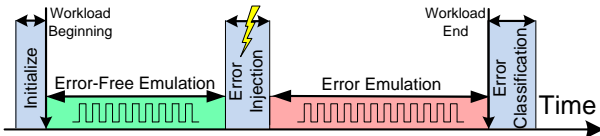orkload output while it affects set s of flip-flops at the end of the first cycle after the particle strike. This is denoted as $EPP^{FF}(ff \cap s)$ and is crucial for accurate temporal masking computation.

D. SER Computation

At the end of this analysis, we compute the SER using the results obtained from previous steps. SER of combinational gate g is computed according to:

$$ SER^g = \sum_{w \in PW} \left(FIT^g(l,w) \times \sum_{s \in FFS} \left(SCPP^g(w,s) \times EPP^{FF}(s) \right) \right) \quad (1) $$

where PW is the set of discretized pulse widths and FFS is the set of all possible erroneous flip-flop sets at the end of the first cycle. This means that for each combinational gate, several pulse widths are propagated and for each pulse width, errors in different sets of downstream flip-flops are emulated.

For each flip-flop, a set containing the possible groups of downstream flip-flops that could be affected is computed using emulation-based fault injection. Then, using STA, and assuming the time of error occurrence is random, the shortest path from the target flip-flop to the erroneous flip-flops group is determined and the temporal masking factor for this group is computed accordingly. The weighted sum of these probabilities gives the effective SER for each flip-flop, as follows:

$$ SER^{ff} = FIT^{ff} \times \sum_{s \in FFS} TMF(ff,s) \times EPP^{FF}(ff \cap s) \quad (2) $$

where $TMF(ff,s)$ is the temporal derating of flip-flop ff when the error is propagated to all flip-flops in set s. $EPP^{FF}(ff \cap s)$ is the probability that error is propagated to the workload output while it affects all flip-flops in set s.

SRAMs SER can be computed by considering the set of all possible MBU patterns (PAT), their respective occurrence rates, and their propagation probability to the workload output:

$$ SER^{SRAM} = (\# \ of \ SRAM \ cells) \times \sum_{p \in PAT} FIT^{SRAM}(p) \times EPP^{SRAM}(p) \quad (3) $$

III. EXPERIMENTAL SETUP

In this section, the experimental setup for analyzing the OpenRISC 1200 (OR1200) processor SER with respect to a 45 nm technology node is presented.

A. FIT Rate Analysis

The FIT rate analysis tool considers the effect of neutrons and alpha particles. The neutron energy distribution defined by JEDEC89a standard [23]. The tool uses a nuclear database (CEA, France) to determine the secondary reactions that occur when neutrons interact with the atoms in the CMOS structure. Separate simulations for alpha particles were performed, based on the assumption of *Ultra Low Alpha* (ULA) (0.001 alpha/cm^2/hour) packaging materials.

SER simulations were performed for all the cells in the Nangate 45 nm standard cell library. A generic response model for 45 nm CMOS process was used to model the underlying process sensitivity. For the combinational cells, separate simulations were performed with different output loads (0,1,3,6,10,20,40, and 100 fF) and pulse widths in steps of d=20ps. The Nangate library does not contain SRAM cells, therefore, representative FIT rates and MBU patterns for 45 nm SRAMs presented in [24] were used in this study.

Fig. 2. Important Phases During Error Injection on a Microprocessor

B. OR1200 Processor

OR1200 is an open source 32-bit processor which implements 5-stage pipeline with Harvard architecture, hardware divider and multiplier, and a floating point unit. Both data and instruction caches are write-through with direct-mapped policy and have size of 1024×32 bit and their tags are stored in 256×21 bit array. It also has a 32×32 bit register-file with one write and two read ports.

The processor was synthesized into the Nangate 45nm library using Synospys Design Compiler. Placement, routing, clock-tree synthesis and post-route optimization were performed using Cadence SoC Encounter. The final netlist consists of 2,694 flip-flops and 30,986 combinational gates. The maximum operating frequency, after all optimizations was 894 MHz. The clock tree that was built by the tools had three levels of inverter/buffer gates. There was one INVX32[2], one INVX32, and 50 BUFX8 in the first, second, and third level, respectively. The average fanout of the leaf-level of the tree is 52. The clock skew and sink transition time[3] for this tree are 13 ps and 181 ps, respectively, which are appropriate for a design with frequency of 894 MHz.

Four workloads from MiBench benchmark suite [25] were executed on the OR1200 during SER analysis experiments including *bitcounts*, *stringsearch*, *qsort*, and *crc32* with runtime of 8, 3, 57, and 202 million cycles, respectively.

IV. EXPERIMENTAL RESULTS

In order to reduce the number of error cases to be emulated, a merging technique is employed based on two observations. First, there are lots of sets of flip-flops with a very small probability of latching which do not significantly affect the final results. Second, there are many cases in the evaluation of combinational logic which lead to the same set of flip-flops at the end of the first cycle. In order to avoid unnecessary fault emulations for such cases, before fault emulation, by assuming that the probability of error propagation from all possible set of flip-flops to the workload output is 1, we compute the maximum SER due to error in each specific flip-flop set. We also merge repeated cases by summing up their maximum SERs. Then, the minimum number of sets that have 95% contributions is determined and fault emulations are only performed for those cases. For the remaining 5%, the propagation probability of the closest set(s)[4] is used instead of performing new emulation. Our detailed analysis for the *stringsearch* benchmark shows that this leads to an inaccuracy of 0.08% in the combinational logic SER while the fault emulations for more than half of the sets are skipped. After fault emulation, the computed propagation probabilities are used to compute the nominal SER.

The number of fault injections is selected dynamically based on the method outlined in [26] to achieve a maximum sampling error of 2%. It took approximately 33 hours to perform the fault emulation for the all four workloads studied in this work. The amount of time could be reduced at the expense of increased sampling error.

---

[2] Xn suffix on the gate name is the drive strength

[3] maximum delay from clock pin to flip-flops

[4] Closest set is the set which have maximum number of common flip-flops with the desired set. If there are few sets with the same number of common flip-flops, average propagation probability is used.
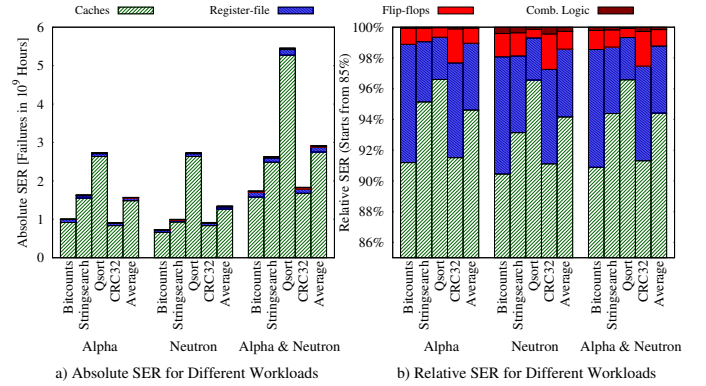

a) Absolute SER for Different Workloads   b) Relative SER for Different Workloads

Fig. 3. Contribution of Different Types of Components on the Overall SER

A. Overall SER

Figure 3 shows the alpha and neutron SER contribution from all three components for four workloads. As it can be seen, the alpha and neutron SER are similar in magnitude. At this point, where no ECC is applied to memory units, the SER is dominated by the memory (>90%) while the combinational contribution is well below 1%. Also, the figure clearly shows the dependency of the SER on the workload with QSort being nearly twice as sensitive as the others.

B. Memory Protection

It is well known that SRAMs are the largest contributor to the overall SER and in most designs with reliability targets, they are ECC protected. Our experiments show that the effectiveness of SEC codes can be maximized for mitigating MBU errors when used together with memory interleaving. Memory interleaving is primarily used to manage the aspect ratio and to improve the timing of memories. However, it can also significantly mitigate the effect of MBUs [27].

In order to protect the write-through instruction and data cache memory units, a parity bit scheme is implemented. Whenever an error is detected in these memory units, a cache miss is issued and error-free data is recovered from the main memory. For protecting the register-file, a hamming code with distance three is employed. Figure 4 shows the SER estimation results for different interleaving schemes. Our experiments show that for an interleaving distance of four (ID=4) the overall memory units SER is less than that of a single flip-flop.

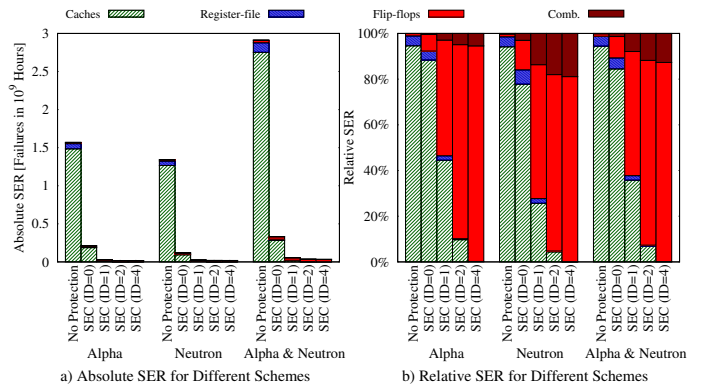a) Absolute SER for Different Schemes   b) Relative SER for Different Schemes

Fig. 4. Contribution of Different Types of Components on the Overall SER Where Caches and Register-file SRAMs Have Different Interleaving Distances (IDs)
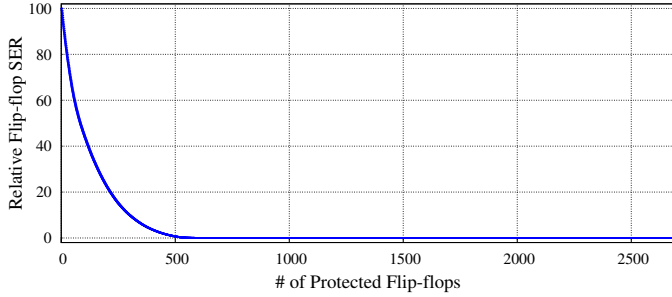
Fig. 5. The Effect of the Selective Protection of the Flip-flops on the Overall Flip-flops SER

Fig. 7. Frequency vs. SER

C. Flip-flop SER

After the memory is efficiently protected, flip-flops become the dominant contributor to the overall SER. Our analysis shows that the SER of flip-flops are non-uniform and over 75% of the flip-flops have a negligible contribution to the overall SER. Although these flip-flops have a considerable raw FIT rate, logical masking prevents errors in these flip-flops from propagating to the workload output. Figure 5 shows the effective flip-flop SER assuming the most critical flip-flops are selectively replaced with hardened flip-flops (e.g. DICE, TMR). As it can be seen, protecting 200 flip-flops (less than 10%) reduces the overall flip-flop SER by 80%. This clearly shows the importance of selective protection for flip-flops. Our analysis shows that these top 200 flip-flops are mostly located in pipeline stages.

In another experiment, we investigated the top 200 vulnerable flip-flops for four MiBench workloads to see whether the vulnerabilities of flip-flops are uniform among different workloads or not. The results of this investigation are summarized in Figure 6. The results reveal that 94 of these flip-flops are common in all workloads whereas 69 (46+23) and 69 (32+36+1) are common between three and two workloads, respectively. There are 79 flip-flops which appear in the top 200 list of only one workload. This indicates that on average only $\frac{79}{4} = 19.75$ flip-flops (less than 10% of 200) in each list differ from the others. Therefore, results from only a few workloads which cover all the functional units (i.e. memory, IO, ALU, floating-point) are sufficient to protect the processor for a broad class of workloads.

D. SER Dependence on Clock Frequency

Figure 7 shows the SER of combinational logic and flip-flops in OR1200 for neutron and alpha-induced particles. As expected, both neutron- and alpha-induced combinational logic
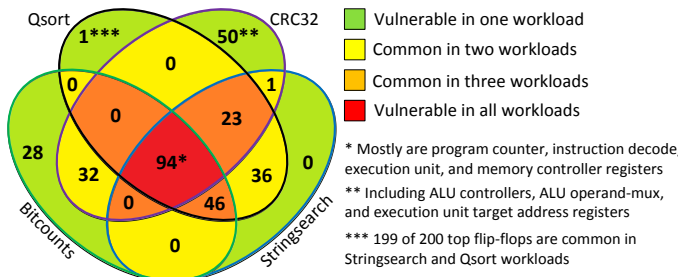

Fig. 6. Investigation of Top 200 Vulnerable Flip-flops of Four MiBench Workloads
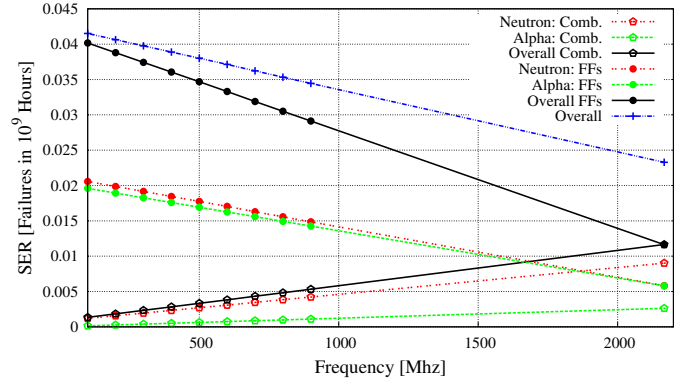
SER show a linear relation with clock frequency which is mainly due to decreasing of latching-window masking. In contrast, there is a reverse linear relationship between flip-flop SER and clock frequency because of the increase in the temporal masking factor of flip-flops. This observation is despite the common assumption that flip-flop SER is largely independent of clock frequency. The major shortcoming of radiation-based experiments in [5, 6] is that the test circuits are typically very simple (e.g. flip-flop chain, inverter chain, comparator) and have just one level of flip-flops. Such circuits cannot capture the effect of temporal masking of flip-flops. Our experiments show that such dependency is not only significant, but also dominates the increasing logic SER resulting in a decreasing trend in the overall SER.

For estimating the threshold frequency at which the logic SER would exceed that of flip-flops, data extracted from the experiments must be extrapolated. Although this is not an accurate method as the circuit structure would be different for a high-speed design, we believe this gives an indication of the frequency range where such an issue occurs. Using this method, it is seen that the threshold frequency is about 2.2 GHz which is a typical frequency of current high-end processors.

E. Clock and Reset Tree Contribution

A combinational gate in the data-path is only sensitive to soft-errors during a short time interval when the transient error could propagate to a flip-flop, whereas the gates on a reset or clock-tree are potentially always sensitive. The radiation-testing on test chips in [7, 8] shows that the clock tree is a significant contributor to the overall SER.

In order to investigate the validity of such observations for a complex design, we did an analysis to compute the contribution of the clock and reset trees to the combinational logic SER. Before discussion about the results of this analysis, it is very important to understand the effects of gate sizing and output load on the gate SER. Figure 8.a shows the total FIT-rate (pulse width>0) of different inverter/buffer cells for typical load capacitances. As it can be seen, larger gate size and load capacitance result in less FIT rate for both types of cells. However, unlike inverters, buffers with higher sizing ratio have considerable FIT rate in the presence of load capacitance larger than 40 fF. This can be interpreted with respect to the layout of such buffers. Figure 8.b shows the layout of BUFX8 cell from the Nangate library. Buffer cells consist of two inverters and as it can be seen, sizing is only applied to the second inverter. Consequently, the first inverter has a small load capacitance

(a) Dependency of FIT Rate on Cell Load   (b) BUFX8 Layout
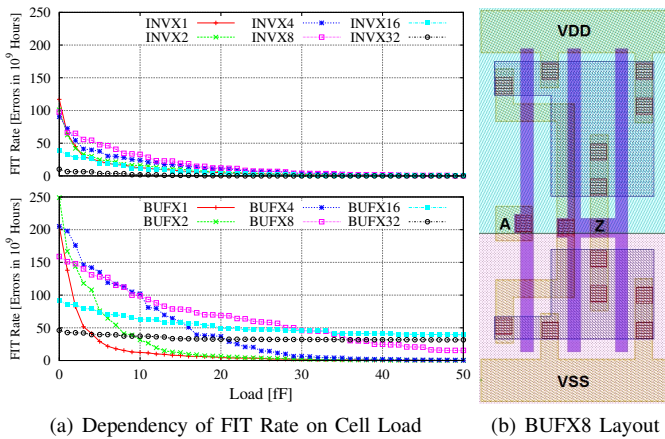
Fig. 8. FIT Rate vs. Load Capacitance (Extracted by FIT Rate Analysis Tool)

and the output load capacitance has no effect on the FIT rate of the first cell. However, by increasing the load capacitance of the cell, delay of the second inverter increases and it would be able to electrically mask more pulses from the first inverter. This explains why the FIT rate diagrams for buffers do not become constant after a while and have a slow decreasing rate.

This analysis showed a 1.8% and 7.2% contribution to the overall combinational SER for the clock and reset-tree, respectively. The small contribution of the clock tree in this design, is due to the fact that the buffers and inverters in the clock-tree were large and had relatively large output loads, thus making them relatively immune to SETs. In this design, Cadence SOC encounter chose large inverter/buffers to achieve the required slew rate targets and minimize the clock tree depth. Large rise/fall time may cause that data is not properly latched in the flip-flops and also it results in higher leakage power consumption. Therefore, it is common to have large size gates with considerable load capacitances.

Unlike the clock tree, the reset tree does not have very tight constraints. It only needs to obey the input to register timing constraints and has to have an acceptable slew rate as specified in the library. For the asynchronous reset pin of OR1200 which is connected to 1322 flip-flops, we assumed an input delay of 400 ps. Employed EDA tools designed the reset tree mostly with small size gates and hence it has a considerable FIT rate.

V. CONCLUSION

We presented a comprehensive SER analysis for a relatively complex design using an efficient hierarchical modeling technique that enabled us to accurately model the intrinsic SER as well as all the masking factors up to the application level. After ECC is employed, the flip-flops dominated the SER and we showed how, using selective mitigation, less than 10% of them need to be hardened to reduce the FF SER by 80%. Interestingly, we found that the overall SER increased with the clock frequency due to the reduced SEU temporal masking. Our results showed that the clock tree was not a significant contributor to the overall SER due to the large size of the clock buffers.

ACKNOWLEDGMENT

REFERENCES

[1] E. Ibe, H. Taniguchi, Y. Yahagi, K.i. Shimbo, and T. Toba. Impact of scaling on neutron-induced soft error in SRAMs from a 250 nm to a 22 nm design rule. *IEEE Transactions on Electron Devices*, 57(7):1527–1538, 2010.

[2] A. Dixit and A. Wood. The impact of new technology on soft error rates. In *International Reliability Physics Symposium*, pages 5B–4, 2011.

[3] V. Ferlet-Cavrois, L.W. Massengill, and P. Gouker. Single Event Transients in Digital CMOS: A Review. *IEEE Transactions on Nuclear Science*, 60(3):1767, 2013.

[4] R.C. Baumann. Radiation-induced soft errors in advanced semiconductor technologies. *IEEE Transactions on Device and Materials Reliability*, 5(3):305–316, 2005.

[5] B. Gill, N. Seifert, and V. Zia. Comparison of Alpha-particle and Neutron-induced combinational and sequential logic error rates at the 32nm technology node. In *International Reliability Physics Symposium*, pages 199–205, 2009.

[6] N.N. Mahatme, S. Jagannathan, T.D. Loveless, L.W. Massengill, B.L. Bhuva, S.-J. Wen, and R. Wong. Comparison of combinational and sequential error rates for a deep submicron process. *IEEE Transactions on Nuclear Science*, 58(6):2719–2725, 2011.

[7] N. Seifert, P. Shipleg, M.D. Pant, V. Ambrose, and B. Gill. Radiation-induced clock jitter and race. In *International Reliability Physics Symposium*, pages 215–222, 2005.

[8] M. Cabanas-Holmen, E.H. Cannon, A.J. Kleinosowski, J. Ballast, J. Killens, and J. Socha. Clock and reset transients in a 90 nm RHBD single-core Tilera processor. *IEEE Transactions on Nuclear Science*, 56(6):3505–3510, 2009.

[9] D.L. Hansen, E.J. Miller, A.j. Kleinosowski, K. Kohnen, A. Le, D. Wong, K. Amador, M. Baze, D. DeSalvo, M. Dooley, et al. Clock, flip-flop, and combinatorial logic contributions to the SEU cross section in 90 nm ASIC technology. *IEEE Transactions on Nuclear Science*, 56(6):3542–3550, 2009.

[10] B. Narasimham, M.J. Gadlage, B.L. Bhuva, R.D. Schrimpf, L.W. Massengill, W.T. Holman, A.F. Witulski, X. Zhu, A. Balasubramanian, and S.A. Wender. Neutron and alpha particle-induced transients in 90 nm technology. In *International Reliability Physics Symposium*, pages 478–481, 2008.

[11] M. Ebrahimi, L. Chen, H. Asadi, and M. B. Tahoori. Class: Combined logic and architectural soft error sensitivity analysis. In *Asia and South Pacific Design Automation Conference*, 2013.

[12] M. Ebrahimi, H. Asadi, and M.B Tahoori. A layout-based approach for multiple event transient analysis. In *Design Automation Conference*, pages 100–105, 2013.

[13] R. Harada, Y. Mitsuyama, M. Hashimoto, and T. Onoye. Neutron induced single event multiple transients with voltage scaling and body biasing. In *International Reliability Physics Symposium*, pages 3C–4, 2011.

[14] D. Rossi, M. Omana, F. Toma, and C. Metra. Multiple transient faults in logic: an issue for next generation ICs? In *International Symposium on Defect and Fault Tolerance in VLSI Systems*, pages 352–360, 2005.

[15] T.D. Loveless, S. Jagannathan, T. Reece, J. Chetia, B.L. Bhuva, M.W. McCurdy, L.W. Massengill, S.-J. Wen, R. Wong, and D. Rennie. Neutron-and proton-induced single event upsets for D-and DICE-flip/flop designs at a 40 nm technology node. *IEEE Transactions on Nuclear Science*, 58(3):1008–1014, 2011.

[16] E. Costenaro, D. Alexandrescu, K. Belhaddad, and M. Nicolaidis. A Practical Approach to Single Event Transient Analysis for Highly Complex Design. *Journal of Electronic Testing*, 2013.

[17] L. Entrena, M. Garcia-Valderas, R. Fernandez-Cardenal, A. Lindoso, M. Portela, and C. Lopez-Ongil. Soft error sensitivity evaluation of microprocessors by multilevel emulation-based fault injection. *IEEE Transactions on Computers*, 61(3):313–322, 2012.

[18] L. Chen, M. Ebrahimi, and M.B. Tahoori. CEP: Correlated Error Propagation for Hierarchical Soft Error Analysis. *Journal of Electronic Testing*, 2013.

[19] R. Rajaraman, J.S. Kim, N. Vijaykrishnan, Y. Xie, and M.J. Irwin. SEAT-LA: a soft error analysis tool for combinational logic. In *VLSI Design*, 2006.

[20] E. Costenaro, A. Evans, D. Alexandrescu, L. Chen, M. Tahoori, and M. Nicolaidis. Towards a hierarchical and scalable approach for modeling the effects of SETs. In *IEEE Workshop on Silicon Errors in Logic-System Effects (SELSE)*, 2013.

[21] N. Seifert and N. Tam. Timing Vulnerability Factors of Sequentials. *IEEE Transactions on Device and Materials Reliability*, 4(3):516–522, 2004.

[22] A. Mohammadi, M. Ebrahimi, A. Ejlali, and S.G. Miremadi. SCFIT: a FPGA-based fault injection technique for SEU fault model. In *Design, Automation and Test in Europe Conference*, pages 586–589, 2012.

[23] Jedec89c, http://www.jedec.org/standards-documents.

[24] D. Alexandrescu. A comprehensive soft error analysis methodology for SoCs/ASICs memory instances. In *International On-Line Testing Symposium*, pages 175–176, 2011.

[25] M.R. Guthaus, J.S. Ringenberg, D. Ernst, T.M. Austin, T. Mudge, and R.B. Brown. Mibench: A free, commercially representative embedded benchmark suite. In *IEEE International Workshop on Workload Characterization*, pages 3–14, 2001.

[26] R. Leveugle, A. Calvez, P. Maistri, and P. Vanhauwaert. Statistical fault injection: quantified error and confidence. In *Design, Automation and Test in Europe*, pages 502–506, 2009.

[27] S. Baeg, S. Wen, and R. Wong. SRAM interleaving distance selection with a soft error failure model. *IEEE Transactions on Nuclear Science*, 56(4):2111–2118, 2009.