

On-Line Characterization and Reconfiguration for Single Event Upset Variations

Kenneth M. Zick and John P. Hayes
{kzick, jhayes}@umich.edu

Advanced Computer Architecture Laboratory, University of Michigan
Ann Arbor, MI 48109 USA

Abstract- The amount of physical variation among electronic components on a die is increasing rapidly. There is a need for a better understanding of variations in transient fault susceptibility, and for methods of on-line adaptation to such variations. We address three key research questions in this area. First, we investigate accelerated characterization of individual latch susceptibilities. We find that on the order of 10 upsets per latch must be observed for variations to be adequately characterized. Second, we propose a method of on-line hardware reconfiguration using incremental place-and-route on FPGAs. Surprisingly, we find that highly localized place-and-route changes (e.g. restricted to groups of 8 flip-flops) are sufficient for realizing most of the possible benefits. Lastly, we quantify potential improvements in system-level soft error rates via Monte Carlo simulation experiments. The study highlights both what is required for and what can be gained by on-line adaptation.

I. INTRODUCTION

AS semiconductor scaling continues to advance, the amount of physical variation across components is increasing rapidly. For instance the 3σ spread in threshold voltages (V_t) is currently estimated to be 42%, and is projected to soon double due to random dopant fluctuations and other effects (see Table 1). Variation in transient fault susceptibility is not as well understood, but it is clear that variations in V_t and transistor gate length are impacting the minimum amounts of charge (Q_{crit}) that can upset logical states [5][7]. Most alarming of all, transient fault rates tend to have an exponential dependence on Q_{crit} ; one model indicates that a 70% difference in Q_{crit} can correspond to a 15 \times difference in fault rates [17]. Variation is already being implicated in a puzzling mismatch between fault simulations and radiation tests [19]. One can expect transient fault variations to play an even more prominent role in upcoming nanoelectronic technologies subject to thermal noise and quantum mechanical effects. The phenomenon of variation presents an opportunity for novel forms of hardware adaptation that can complement existing methods of soft error mitigation.

The main transient faults of interest in this work are *single event upsets* (SEUs), in which a radiated particle causes the state of a storage cell to be changed. SEUs can occur in sequential logic storage elements such as latches (and

TABLE 1

PROJECTED 3σ VARIATION IN THRESHOLD VOLTAGES (V_t), ITRS 2008 [13]

2009	2010	2011	2012	2013	2014
42%	42%	42%	58%	58%	81%

associated flip-flops), as well as in memory cells. We are specifically interested in latches and their inherent relative susceptibilities to transient faults, in other words, their individual tendencies to incur faults. These component level differences are a form of *intra-die variation* (within a single die), as distinct from inter-die (across different dies). Intra-die variation can include both stochastic variation, which is random from component to component, and correlated variation affecting components in a certain spatial region.

Physical variation is especially of interest in the domain of reconfigurable computing, where systems can potentially adapt to variations. For instance, with field-programmable gate arrays (FPGAs) the placement of design elements can be altered to account for non-uniform characteristics in the underlying platform. If the amount of variation is significant, then placement decisions can have a significant impact on system-level behavior.

Three key research questions need to be addressed regarding adaptation to intra-die variations. First, how can variations be efficiently and adequately characterized? Second, how can hardware be reconfigured to account for the variations? Third, what improvements in system-level soft error rates are possible? This work addresses all three questions, and presents new results from Monte Carlo simulation experiments.

The remainder of this paper is structured as follows. We summarize related work in Section II, then propose methods for characterization and reconfiguration in Section III. We present experimental results in Section IV and conclusions in Section V.

II. RELATED WORK

Research into variation-aware adaptation is currently being conducted in multiple contexts, including timing [21][14], leakage power [4], and temperature [15]. To date there has been little work on adapting to variations in fault susceptibility. Some attempts have been made at on-line characterization in the context of single event transients [18][20].

This work was supported in part by a NASA Langley Research Center GSRP Fellowship and by the National Science Foundation under grant CCF-0702276.

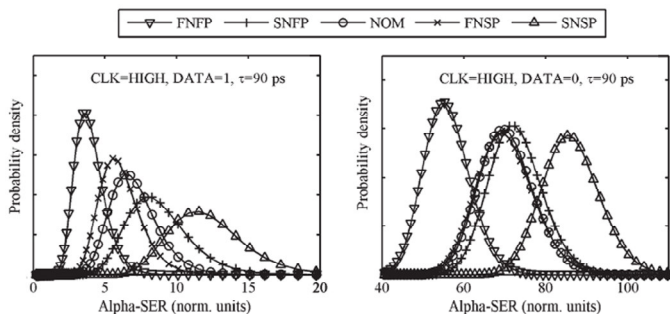


Fig. 1: Analytical model of variation in flip-flop SEU rate [11]. Note the spread within each of the five process corners, suggesting intra-die variations.

Very little experimental data has been published on variations in transient fault susceptibilities. Collecting statistically significant data for individual components can be difficult, given that transient faults are very improbable events. There may also be proprietary reasons why such data tends not to be divulged. A few radiation test reports do include data for individual flip-flop instances [6], allowing at least a glimmer of the variations to be observed. A recent analytical model of 90-nm latches by Heijmen *et al.* [11] suggests that the amount of variation in SEU susceptibility may already be quite significant; see Fig. 1. Variations in Q_{crit} have been studied, often in the context of SRAM cells [12].

Variation-aware adaptation requires a mechanism for reconfiguration; examples include adaptive body bias and FPGA place-and-route. Place-and-route methods that account for FPGA configuration upsets (firm errors) have been proposed [24][27][28]. A method of on-line incremental FPGA place and route is considered in [23].

III. PROPOSED APPROACH

We first define our models and assumptions, and then propose approaches for on-line characterization and on-line reconfiguration.

A. Preliminaries

The transient faults of interest here are single event upsets which alter the state of a latch. (We do not model single event transients or FPGA configuration upsets here.) A latch SEU can occur while a latch is closed. We define an SEU involving a $0 \rightarrow 1$ transition as a *rising-SEU*, and one with a $1 \rightarrow 0$ transition as a *falling-SEU*. We distinguish between a latch's rising-SEU rate and falling-SEU rate, which apply when the latch state is 0 and 1, respectively.

We next describe our model of SEU variation. At the lowest level of abstraction, we define a hardware platform consisting of N latches. An individual latch i has *relative susceptibilities* $\alpha_R(i)$ and $\alpha_F(i)$, which represent the ratios of rising- and falling-SEUs rates at latch i versus the mean rising- and falling-SEU rates across all latches. For example, $\alpha_R(i) = 1.2$ indicates that latch i 's rising-SEU rate is 20% higher than the mean. The mean rates across all latches are denoted λ_R and λ_F , and are a function of the technology, latch design, and the fault environment (e.g. radiation flux, particle mix, distribution of particle energies). The rates for each latch i are

determined by $\lambda_R(i) = \alpha_R(i) \times \lambda_R$ and $\lambda_F(i) = \alpha_F(i) \times \lambda_F$. Note that the $\alpha_R(i)$ and $\alpha_F(i)$ terms are convenient for characterizing the latch variation since the SEU rates themselves may be dynamic or unknown. Both stochastic and correlated variation can be captured by the $\alpha(i)$ terms, similar to models of variation used in other contexts such as timing [21].

Complementing the underlying hardware platform, we define at a higher level of abstraction a system architecture having M state bits. Each state bit j has weights $w_0(j)$ and $w_1(j)$ indicating the fraction of time spent in the 0 and 1 states during a given workload. The state bits are placed at physical locations such that each state bit j is mapped onto two latches composing a master-slave flip-flop. A pairing of a latch i and a state bit j will be denoted ij .

Only a fraction of SEUs occurring at state bit j cause a system-level error, due to timing masking, logic masking, and complex interactions between the two. These two inter-related masking phenomena are combined here into a single term, the vulnerability fraction $VF(j)$ [29]. In the most general case, the vulnerability fraction may be data-dependent and thus will be denoted $VF_0(j)$ and $VF_1(j)$.

The system-level soft error rate contribution associated with pair ij for a computation C is given by:

$$SER_{ij}(C) = \alpha_R(i) \times \lambda_R \times w_0(j) \times VF_0(j) + \alpha_F(i) \times \lambda_F \times w_1(j) \times VF_1(j) \quad (1)$$

Under a single fault assumption, the total system-level soft error rate can be estimated by:

$$SER(C) = \sum_{ij} SER_{ij}(C) \quad (2)$$

B. On-line Characterization

Variation-aware adaptation requires a feasible method of characterizing the relative fault susceptibilities. One question is whether latch fault susceptibilities must be characterized directly or whether they can be inferred from other latch parameters. Ideally, the susceptibilities could be inferred at least in part from the amount of leakage current, propagation delay, or even the tendency towards the 0 or 1 state upon power-up [16]. No such low-cost, indirect method of fault susceptibility characterization has yet been established.

Here we consider direct characterization via detection and counting of latch SEUs. We specifically propose using latch SEU counts as maximum likelihood estimators of the actual susceptibilities. For example, a latch incurring twice as many SEUs as another will be estimated to have twice the susceptibility. The number of SEUs that must be observed to make this approach sufficiently accurate will be determined in Section IV.

One option for characterization is off-line radiation testing, which is quite expensive. The costs are especially high for fine-grained characterization since high fluence tests are required. Off-line characterization may become feasible for emerging technologies prone to high rates of transient faults. Currently however, it is usually impractical to perform fine-grained susceptibility characterization of all field units before they are deployed.

In some cases *on-line* characterization is feasible. One advantage is that it allows the characterization to be performed over extended periods, and in the actual radiation environments. Furthermore, it may capture lifetime shifts in component parameters. This approach is only feasible if latch SEUs can be made to occur often enough. Several methods of SEU acceleration will be considered next.

Latches under test can be placed in a configuration that maximizes the rate of detectable SEUs. The latch clock can be turned off in order to maximize the window of vulnerability (WoV), which is the fraction of time that a latch is prone to an SEU that propagates beyond the current clock cycle. In normal operation this window is often open only 10% of the time due to timing masking [22]; here we increase the window to 100%. This un-clocked configuration also minimizes clock power.

An example of a simple on-line test procedure is shown in Fig. 2. Latches under test are first initialized to the desired state, and then left idle for an extended period to act as SEU detectors. Any SEU that occurs leaves a latch in an error state indefinitely (Fig. 2b). The latch states are regularly monitored by a controller, and evidence of SEU events is recorded in individual SEU counts maintained in SEU-protected memory. The latches are then re-initialized and the process is repeated.

Note that this procedure can typically be implemented on a reconfigurable platform without requiring any special circuitry. For instance, with the Virtex-family, latches can be initialized with either an asynchronous set/reset or by reloading the bitstream. The latch states can be read out in the background during system operation via the Virtex capture and readback feature.

One on-line test scenario is to characterize all unused latches opportunistically during system operation. For instance, a design may require 70% utilization of latches, allowing the remaining 30% to be characterized without impacting system availability. If necessary, user logic could be swapped from location to location such that all latches would eventually undergo testing; similar schemes have been devised to test for permanent faults [9][25].

An alternative scenario is to perform SEU testing of all latches simultaneously during system down time. This nonconcurrent approach is sometimes the simplest and quickest way to characterize the entire platform. A particular advantage is the greater freedom to set the operating conditions. By lowering the operating voltage, the Q_{crit} values can be decreased and thus the SEU rates can be significantly accelerated. (The supply voltage must not be so low as to significantly shift the relative susceptibilities.) In [11], lowering the supply voltage to 0.4V is shown to increase the latch SEU rates by 1 to 2 orders of magnitude.

Testing may be further accelerated by leveraging periods of high radiation. For instance, spacecraft encountering solar flares or passing through the South Atlantic Anomaly can experience upset rates 100x higher than normal [8]. Such periods may be unsafe for normal system operation, but they could provide a useful source of SEUs for accelerated testing.

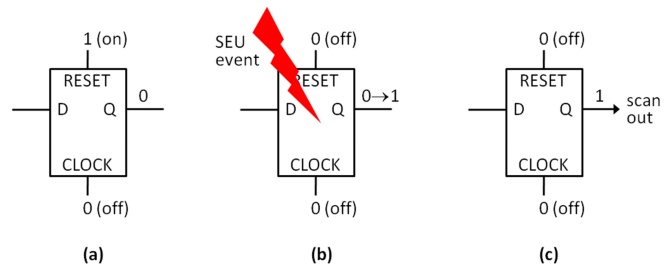


Fig. 2: Example of on-line SEU characterization of level-sensitive latches. a) Latch under test is initialized to the desired state. b) The latch is left idle to act as an SEU detector. c) Latch states are periodically read out so that any SEU events can be recorded.

Even with the above methods of acceleration, there is the complication of characterizing latches in all four permutations of data and clock states. The master and slave latches cannot necessarily be tested simultaneously; some reconfigurable platforms do not support both clocks being turned off at the same time, and thus only one type of latch is susceptible to SEUs at any time. This implies that at least four separate test campaigns (two for master latches and two for slave latches) would be needed in order to fully characterize all latches. Most problematically, the SEU rate in some of these four configurations may be much lower than in others, forcing much longer test times.

We now consider whether SEU characterization can be made more practical by focusing on a subset of the four states. To investigate this possibility, we calculated the relative SEU contributions of the four states using some typical values for the various parameters. First we note that latch SEU rates are often heavily dependent on the data state of the latch. The latches in the Actel ProASIC FPGA are 10x more susceptible when holding a 1 bit than when holding a 0 [3]. Note also that latches are prone to upsets only for particles above a certain energy (measured in linear energy transfer), and that this energy threshold may be different for rising- and falling-SEUs. In the case of the Xilinx Virtex-4QV, the rising-SEU threshold is 1.5 MeV-cm²/mg, but the falling threshold is only 0.5 MeV-cm²/mg [2]. Radiation flux tends to be dramatically higher at lower energies, so even slight differences in threshold can lead to large differences in fault rates. In any case, we used the flip-flop study performed in [11] as a guide and assumed a 0-1 bias of 4x.

Next, we modeled the difference in raw SEU rates between master and slave latches. Often master latches have higher raw SEU rates, for instance due to having lower drive strengths and lower Q_{crit} . Four of the five flip-flops designs studied in [11] have a significantly higher rate for the master latch; the differences are in the range of 3 to 10x, and we assume here a master-slave bias of 4x. Lastly, we chose typical values for the window of vulnerability. The WoV tends to be significantly higher for master latches; slave latches are susceptible to SEUs only during the second phase of a clock cycle when it may be too late for the SEU to be captured by downstream logic. We set the master WoV to 0.25 and the slave WoV to 0.04, based on data from [22].

TABLE 2
EXAMPLE OF TYPICAL SEU CONTRIBUTIONS

Scenario	Normalized SEU rate	Mean WoV	Rate of propagated SEUs	Fraction of propagated SEUs
Master latch in state <i>A</i>	1.0	0.25	0.25	77%
Master latch in state <i>B</i>	0.25	0.25	0.0625	19%
Slave latch in state <i>A</i>	0.25	0.04	0.01	3%
Slave latch in state <i>B</i>	0.0625	0.04	0.0025	1%

Combining the above factors, we determined the relative contributions of the four SEU types to the total number of SEUs that propagate to the next clock cycle. We found that with these parameter values, a full 77% of all propagated SEUs originate from a master latch in the most susceptible data state *A*. (State *A* can be a 0 or a 1 depending on the technology and latch design.) Some 19% come from a master latch in the least susceptible data state *B*. Only 3% of all propagated SEUs come from a slave latch in state *A*, and a mere 1% from a slave latch in state *B*. These estimates are summarized in Table 2. The rate of propagating SEUs shown in the fourth column is simply the raw SEU rate times the mean WoV.

Given their overwhelming importance, we propose a simplified test approach focused on master latches in state *A*. This reduced focus greatly improves the test time; instead of four tests of varying lengths (lower SEU rates require longer test times), only a single test campaign is performed and at the highest fault rate. This also reduces the amount of characterization data that must be maintained, and enables a simplified mitigation strategy that will be discussed next.

C. On-line Reconfiguration

Given estimates for the latch susceptibilities, a system can be reconfigured with a more effective mapping between logical state bits and physical latch instances. Performing a full variation-aware place and route cycle on-line would be quite costly in terms of computational and memory resources, especially if attempted on an embedded system. Instead we investigate the potential for *incremental* place and route.

We propose decomposing the logic array into a collection of virtual neighborhoods, and then optimizing the placement of state bits only within each neighborhood. This reduces the computational and memory burden significantly, and avoids most of the difficulty of routing and timing closure. Logic that has special placement constraints or that resides on a critical path can be made ineligible for re-placement. A simple example of such incremental place-and-route is illustrated in Fig. 3.

Along with relative fault susceptibilities, we will need estimates of the other factors in Eq. (1) in order to evaluate different system configurations. The mean SEU rates λ_R and λ_F can be found through off-line radiation testing combined with either a radiation model or runtime monitoring of a system’s actual radiation environment. The state weights $w_0(j)$ and $w_1(j)$ can be found by sampling the latch states on a representative workload; with some FPGAs this type of sampling can be performed transparently to a computation via

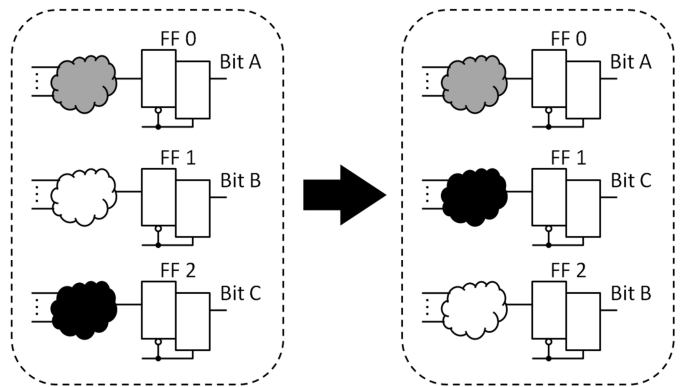


Fig. 3: Example of reconfiguration. A virtual neighborhood containing 3 flip-flops is shown before (left) and after (right) reconfiguration. The mapping of state bits (*A,B,C*) to physical latch/flip-flop sites (0,1,2) is modified in order to reduce system-level SER.

a non-intrusive scan. The *VF* factors can be estimated through methods such as statistical fault injection [29] and a careful study of the timing windows of vulnerability [10].

We suggest the following low-cost heuristic for finding improved placements. Physical flip-flop instances are ranked according to a single variable – the relative SEU susceptibility of the master latch in state *A*. Logical state bits in the design are ranked by their weighted vulnerability fraction in state *A* (weight w times *VF*). The worst-case unassigned state bit is paired with the best-case unassigned latch instance. Essentially this causes the threats at the physical and logical levels to be negatively correlated, in order to minimize the system-level soft error rate contribution. The process is repeated until all eligible state bits have been placed.

The main questions are whether such a local approach can be effective enough in reducing the soft error threat and, if so, what is a proper neighborhood size? We address these questions in experiments described in the next section.

IV. EXPERIMENTAL RESULTS

We created a statistical model of a computational system and conducted Monte Carlo experiments to determine the efficacy of the proposed methods. The model contains 100,000 master-slave latch pairs, each latch having stochastic variation in its rising- and falling-SEU susceptibility. The susceptibilities are normally distributed, with a standard deviation equal to 20% of the mean (unless otherwise noted), consistent with various estimates given in [5][11][13]. Note that the variation in SEU susceptibility can be greater than the variation in Q_{crit} . Only stochastic variation is modeled here; correlated variation is assumed to be negligible within the small optimization neighborhoods considered. The w and *VF* variables for state bits were given uniform random distributions from 0.0 to 1.0. Unless otherwise stated, the remaining parameters were set to the values described in Section III.B, with a neighborhood size of 8 flip-flops and a logic utilization of 50%. For each experiment, we simulated latch susceptibility characterization followed by variation-aware reconfiguration, and measured the impact on SER.

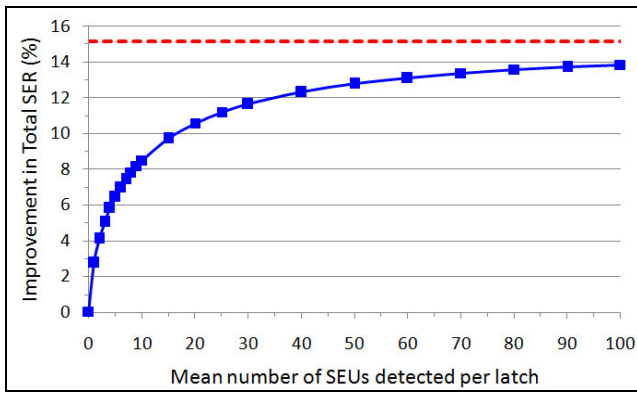


Fig. 4: Improvement in SER vs. amount of SEU characterization. Relatively few SEUs are needed per latch in order to realize most of the possible SER improvement.

In the first experiment, we addressed the question of how much SEU sampling is required to adequately characterize relative latch susceptibilities. One published guideline suggests that 100 SEU events are needed to characterize *average* susceptibility [1]; we were interested to find a similar empirical guideline for relative susceptibility of individual latches. We simulated successive amounts of SEU data collection followed by reconfiguration based on the susceptibility estimates. Thirty Monte Carlo trials were performed at each data point. The results are illustrated in Fig. 4. Note that the error bars at each data point in this figure (and the following figures) are too small to be visible. We found that SER could be improved by up to 15%. Surprisingly, we found that most of the possible SER improvement can be realized even after a small number of SEUs are observed per latch. For instance, after an average 8 SEUs, the improvement in SER is over half of what could be achieved with perfect knowledge of the relative susceptibilities.

We also studied the sensitivity of two parameters: latch utilization, and 0-1 bias in the SEU rates (i.e. skew between the rising- and falling-SEU rates). We tested a range of latch utilization from 10 to 100%, and four representative values of 0-1 bias: 1x, 4x, 10x, and 50x. We limited the amount of SEU characterization to an average of 10 SEUs per master latch, based on the findings from the previous experiment. The results are presented in Fig. 5. Since the proposed heuristic uses only the higher of the two SEU rates, it performs best when this rate is dominant (high 0-1 bias). There is little difference in performance for biases in the range of 4-50x. Understandably, the method works also best at lower levels of utilization, since in those cases more of the unreliable latches can be left unused. However, even at 100% utilization some SER improvement is possible.

Lastly, we measured the potential for SER improvements as a function of the amount of variation and as a function of the optimization neighborhood size. We simulated susceptibility characterization (again limited to an average of 10 SEUs per master latch), for three different amounts of variation: standard deviations equal to 10, 20, and 30% of the mean. Neighborhood sizes of 2, 4, 8, 16, and 32 flip-flops were

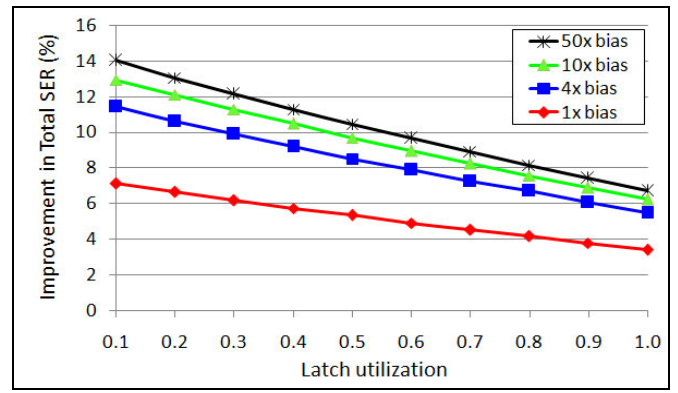


Fig. 5: Improvement in SER vs. latch utilization for four different values of 0-1 bias in SEU rates.

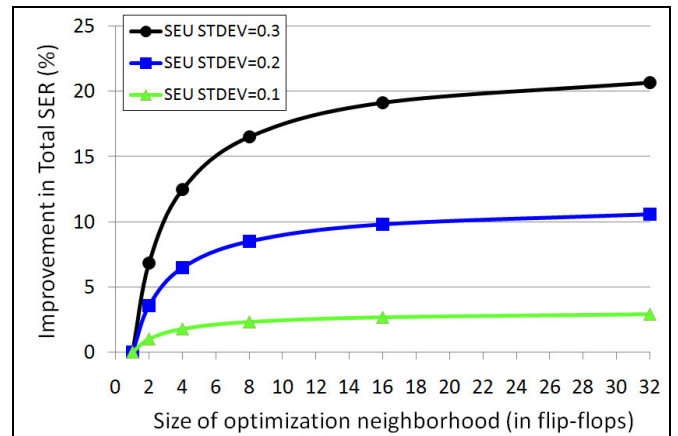


Fig. 6: Improvement in SER vs. neighborhood size, for 3 different amounts of SEU variation.

simulated, with 30 Monte Carlo trials performed for each size. The results are shown in Fig. 6. First note that the amount of SER improvement is much higher for higher amounts of variation. Surprisingly, the SER improvement levels off very quickly with neighborhood size, meaning most of the possible improvement can be realized with very small and tractable sizes. For instance, the amount of improvement with neighborhoods of just 8 flip-flops is 75% of the amount associated with a maximum size neighborhood (100,000). This holds true for all three values of SEU variation that were simulated.

V. CONCLUSIONS

Perhaps the most surprising result from this study is the effectiveness of logic reconfiguration within very small neighborhoods. Optimization within each neighborhood of just 8 flip-flops can produce significant reductions in the soft error rate. This is encouraging since incremental place and route is far more practical when the neighborhood size is small. In fact, modern reconfigurable platforms tend to naturally employ small neighborhoods in their logic arrays,

such as the Virtex-6 FPGA which is organized into “slices” containing 8 flip-flops [26].

We found that the mean number of faults that must be observed per latch to enable an effective reconfiguration is on the order of 10. This compares favorably to an existing rule of thumb [1] suggesting that 100 SEU events are generally required for accurate estimation. The advantage of the proposed approach is that accurate estimates of susceptibility are not required; all that is needed is enough relative data to identify most of the outliers (components that are particularly susceptible or unsusceptible). Whether 10 faults per latch can be generated in a timely fashion depends heavily on the radiation environment and the ability to lower Q_{crit} . In some applications SEUs are too infrequent even with accelerated testing, and thus relative susceptibilities cannot be readily characterized. Direct characterization (either on-line or off-line) of fault susceptibilities will likely be most applicable to emerging technologies with inherently high fault rates.

An additional finding of this study is that, under some typical parameter settings, the vast majority of system-level soft errors can be traced back to a single latch type (master latches) and a single data state. This allows the characterization and reconfiguration processes to be greatly simplified. In fact, due to differences in Q_{crit} and windows of vulnerability, 96% of all system-level errors in our study originate from master latches. For applications with a large imbalance of this kind, the SER can be lowered by optimizing the master latch case. In some systems this can be accomplished simply by lowering the clock duty cycle.

In summary, this study has demonstrated the desirability and feasibility of on-line adaptation to intra-die variations, in the context of transient faults. As the amount of variation in transient fault behavior continues to increase, the potential for adaptation rises as well, as seen in Fig. 6. Consequently, we expect that approaches such as those proposed here will become increasingly important as technology advances.

REFERENCES

- [1] P. Adell and G. Allen, “Assessing and mitigating radiation effects in Xilinx FPGAs,” JPL Publication, Feb. 2008.
- [2] G. Allen *et al.*, “Virtex-4QV static SEU characterization summary,” JPL Publ. 08-16, April 2008.
- [3] G.R. Allen and G.M. Swift, “Single event effects test results for advanced field programmable gate arrays,” *Radiation Effects Data Workshop*, pp. 115-120, 2006.
- [4] M. Ashouei *et al.*, “Probabilistic self-adaptation of nanoscale CMOS circuits: yield maximization under increased intra-die variations,” *Proc. Conf. VLSI Design*, pp. 711-716, 2007.
- [5] A. Balasubramanian *et al.*, “Effects of random dopant fluctuations (RDF) on the single event vulnerability of 90 and 65 nm CMOS technologies,” *IEEE Trans. Nuclear Science*, vol. 54, no. 6, pp. 2400-2406, Dec. 2007.
- [6] X. Cano *et al.*, “Heavy ion test results in a CMOS triple voting register for a high-energy physics experiment,” *Proc. Int’l On-Line Testing Symp.*, pp. 183-184, July 2007.
- [7] X. Fu, T. Li and J. Fortes, “Soft error vulnerability aware process variation mitigation,” *Proc. Symp. High-Performance Computer Arch.*, Feb. 2009.
- [8] J. Gal-Edd and C.C. Fatig, “L2-James Webb Space Telescope operationally friendly environment?” *Proc. Aerospace Conf.*, vol. 1, March 2004.
- [9] M.G. Gericota *et al.*, “DRAFT: an on-line fault detection method for dynamic and partially reconfigurable FPGAs,” *Proc. Int’l On-Line Testing Workshop*, pp. 34-36, 2001.
- [10] T. Heijmen, “Soft-error vulnerability of sub-100-nm flip-flops,” *Proc. Int’l On-Line Testing Symp.*, pp. 247-252, 2008.
- [11] T. Heijmen *et al.*, “A comprehensive study on the soft-error rate of flip-flops from 90-nm production libraries,” *IEEE Trans. Device and Materials Reliability*, vol. 7, no. 1, pp. 84-96, March 2007.
- [12] T. Heijmen *et al.*, “Factors that impact the critical charge of memory elements,” *Proc. Int’l On-Line Testing Symp.*, pp. 57-62, 2006.
- [13] International Technology Roadmap for Semiconductors, 2008 Update.
- [14] K. Katsuki *et al.*, “A yield and speed enhancement scheme under within-die variations on 90nm LUT array,” *Proc. Custom Integrated Circuits Conf.*, pp. 601-604, Sept. 2005.
- [15] E. Kursun and C.Y. Cher, “Variation-aware thermal characterization and management of multi-core architectures,” *Proc. Int’l. Conf. Computer Design*, pp. 280-285, Oct. 2008.
- [16] R. Maes, P. Tuyls and I. Verbauwhede, “Intrinsic PUFs from flip-flops on reconfigurable devices,” *Benelux Workshop on Information and System Security*, 2008.
- [17] D.C. Ness, C.J. Hescott and D.J. Lilja, “Improving nanoelectronic designs using a stat. approach to identify key parameters in circuit level SEU simulations,” *Proc. Sym. Nanoscale Arch.*, pp. 46-53, Oct. 2007.
- [18] I. Polian *et al.*, “Transient fault characterization in dynamic noisy environments,” *Proc. Int’l Test Conf.*, 2005.
- [19] B. Pratt *et al.*, “Fine-grain SEU mitigation for FPGAs using partial TMR,” *IEEE Trans. Nuclear Science*, vol. 55, no. 4, pp. 2274-2280, Aug. 2008.
- [20] A. Sanyal *et al.*, “A built-in self-test scheme for soft error rate characterization,” *Proc. Int’l On-Line Testing Symp.*, pp. 65-70, 2008.
- [21] P. Sedcole and P.Y.K. Cheung, “Within-die delay variability in 90nm FPGAs and beyond,” *Proc. Field Programmable Technology*, pp. 97-104, Dec. 2006.
- [22] N. Seifert and N. Tam, “Timing vulnerability factors of sequentials,” *IEEE Trans. Device and Materials Reliability*, pp. 516-522, 2004.
- [23] N. J. Steiner, “Autonomous computing systems,” PhD thesis, Virginia Tech, March 2008.
- [24] L. Sterpone and M. Violante, “A new reliability-oriented place and route algorithm for SRAM-based FPGAs,” *IEEE Trans. Computers*, vol. 55, no. 6, pp. 732-744, June 2006.
- [25] C. Stroud *et al.*, “On-line BIST and diagnosis of FPGA interconnect using roving STARS,” *Proc. Int’l On-Line Testing Workshop*, pp. 27-33, 2001.
- [26] Xilinx Inc., <http://www.xilinx.com>.
- [27] H.R. Zarandi *et al.*, “SEU-mitigation placement and routing algorithms and their impact in SRAM-Based FPGAs,” *Proc. Symp. Quality Electronic Design*, pp. 380-385, 2007.
- [28] H.R. Zarandi *et al.*, “CAD-directed SEU susceptibility reduction in FPGA circuits designs,” *Proc. Symp. Circuits and Systems*, pp. 3675-3678, 2007.
- [29] K.M. Zick and J.P. Hayes, “High-level vulnerability over space and time to insidious soft errors,” *Proc. High-Level Design, Validation and Test Workshop*, pp. 161-168, Nov. 2008.