

# A New Reliability Evaluation Methodology with Application to Lifetime Oriented Circuit Design

Hamed Sajjadi-Kia, *Student Member, IEEE* and Cristinel Ababei, *Member, IEEE*

**Abstract**—We propose a new circuit level vulnerability and reliability evaluation methodology and utilize it to develop a lifetime aware floorplanning strategy. Our work is motivated by increasingly adverse aging failure mechanisms, which have made reliability a growing fundamental challenge in the design of integrated circuits. Because the proposed methodology is based on a divide and conquer approach, it enjoys the benefits of transistor level accuracy and of block level efficiency. At the core of the lifetime estimation engine lies a Monte Carlo algorithm which works with failure times modeled as Weibull and lognormal distributions for several aging mechanisms including time dependent dielectric breakdown, negative bias temperature instability, electromigration, thermal cycling, and stress migration. To demonstrate the value of the proposed reliability evaluation methodology and floorplanning strategy, we apply them to a Network-on-Chip router design example. The new floorplanning approach is able to find floorplans with up to 15% difference in the lifetime of the router design. In addition, the proposed reliability evaluation methodology identifies the routing computation and virtual channel allocation units as the most vulnerable subblocks of the design. Such information can be very useful to designers to predict circuit and system mean time to failure and to focus on cost effective design techniques targeted at specific parts of the design to improve its lifetime.

**Index Terms**—Reliability estimation, Vulnerability analysis, Aging mechanisms, Floorplanning, Network-on-Chip router.

## I. INTRODUCTION

**R**ELIABILITY has become a growing fundamental challenge in the design of integrated circuits due to increasingly adverse aging failure mechanisms that can cause performance degradation and eventual device and system failure [1]. To maintain downscaling benefits, increasingly complex integrated circuits must be designed with built-in resilience techniques [2]–[4]. To achieve that, one of the main difficulties is to evaluate reliability. Evaluation of reliability is a challenging task because reliability is affected by numerous factors including aging or wearout mechanisms [5] (e.g., time-dependent dielectric breakdown (TDDB) [6], negative bias temperature instability (NBTI) [7]–[9], electromigration (EM) [10], [11], thermal cycling (TC), and stress migration (SM) [12]), process variations, dynamic power and thermal management, workload conditions, and system architecture and configuration.

H. Sajjadi-Kia is with the Department of Electrical and Computer Engineering, North Dakota State University, Fargo, ND 58102.  
E-mail: hamed.sajjadikia@ndsu.edu.

C. Ababei is with the Department of Electrical Engineering, State University of New York at Buffalo, Buffalo, NY 14260. E-mail: cababei@buffalo.edu.  
Copyright ©2012 IEEE

## A. Related Work

1) *Reliability Evaluation Techniques*: While there has been significant work carried out to estimate reliability [14]–[24], we discuss next two approaches that are related to our work. An extensive review of previous reliability simulation tools can be found in [25].

The RAMP approach [15] models the mean time to failure (MTTF) of a processor microarchitecture as a function of temperature related failure rates of individual structures on chip. It divides the processor into several discrete structures (e.g., ALU, register files, etc) and applies analytical models to each structure. Then, it combines the structure level MTTFs to compute the overall MTTF of the entire processor assumed as a series failure system. Because the lifetime distributions of failure mechanisms are assumed to be exponential [16], the reliability is calculated by applying the sum-of-failure-rates (SOFR) model. This approach is not realistic because failure rates of units increase with time due to aging. To address this limitation of the SOFR model, RAMP 2.0 [17], [26] uses lognormal distributions, which are harder to deal with analytically. One of the main limitations of the RAMP approach as an architecture level approach is its accuracy. In addition, it may estimate equal MTTFs for blocks of different sizes but with activity factors that cancel out the area proportionality factor.

Another more recent class of simulation based reliability evaluation approaches are based on Spice simulations. Failure rate based Spice (FaRBS) [27] and Maryland circuit reliability oriented (MaCRO) [29], [30] are circuit level reliability simulation methods. Both of these methods utilize degradation models for TDDB, NBTI, and hot carrier degradation (HCD). They are based on a series of accelerated lifetime models and failure equivalent circuit models for these wearout mechanisms [25], [32]. They employ Spice to calculate electrical parameters of fresh and degraded devices and to predict their degradation or failure from these parameters [27]. The main advantage of this class of simulation methods is the device level granularity that enables reliability analysis at transistor level to identify the most vulnerable transistors. There are some issues related to the Spice based reliability simulation. These approaches do not consider the layout of the system and simulations are done under worst case temperature scenarios, which is not realistic. Besides, Spice circuit simulations tend to take long time especially when done for large circuits. In addition, both methods (FaRBS and MaCRO) are developed under the assumption that failure rate is constant. As discussed above this assumption is inaccurate.

2) *Floorplanning*: Floorplanning is an important step during the design of integrated circuits. Because the relative locations of different subblocks is decided during floorplanning, the overall temperature profile of the chip is directly affected by the quality of the floorplanning step. As such, there has been significant work done on the problem of thermal aware floorplanning [33]–[39]. Even though reliability is directly related to temperature, it has been significantly less investigated. Nevertheless, a reliability aware voltage island partitioning and floorplanning algorithm for SoC is reported in [40]. The algorithm considers the sensitivity of the SoC to soft errors and does not address aging mechanisms. The authors of [41] define reliability in terms of supply voltage noise margin and propose a floorplanning algorithm that distributes the thermal profile evenly and reduces the power supply noise. The effect of temperature on the probability of errors in SRAM memories is discussed in [42], where a leakage aware floorplanner is introduced. Currently, there is no aging failure mechanisms aware floorplanning method reported in the literature.

### B. Contribution

To address the limitations of previous reliability evaluation methods, we propose a new circuit level reliability evaluation methodology. To this end, our main contribution is as follows: (1) We propose and implement a new divide and conquer based reliability evaluation methodology. Its core engine employs a Monte Carlo algorithm, which works with failure times modeled realistically as Weibull and lognormal distributions for five different aging failure mechanisms: TDDB, NBTI, EM, TC, and SM. Hence, our results are more accurate and realistic compared to previous works that are based on the assumption that lifetime distributions are exponential. (2) We utilize the proposed reliability evaluation methodology to develop a new lifetime aware floorplanning strategy that is capable of identifying the most reliable floorplan for a given design. We consider this an essential step towards a design approach where reliability is also a primary objective. To demonstrate the usefulness of the proposed algorithms, we apply them to an Network-on-Chip router as a design example. We analyze its reliability, identify its most vulnerable subblocks, and generate the most reliable floorplan for it.

## II. LIFETIME FAILURE MODELS

### A. Importance of Lifetime Distribution of Failure Mechanisms

Many proposed lifetime reliability models assume a uniform device density on the chip and an identical vulnerability of devices to failure mechanisms [14]. The lifetime distributions of failure mechanisms are usually assumed to be exponential [15], [16], [18], [26], [43]. As discussed in the previous section, this allows system-level reliability to be calculated by applying the sum-of-failure-rates (SOFR) model. However, this approach is not realistic because failure rates of units increase with time due to aging. To address this issue and to develop an accurate reliability model, more general lifetime distributions (e.g., Weibull and lognormal) must be utilized. On the other hand, when Weibull or lognormal distributions are utilized the prediction of reliability becomes more difficult

and therefore Monte Carlo simulations must be employed [16], [26], [43]. In this paper, we adopt Weibull distribution modeling for TDDB, NBTI, TC, and SM and lognormal distribution modeling for EM because these distributions have been found to best fit the corresponding wearout mechanisms [12].

### B. Time Dependent Dielectric Breakdown (TDDB)

Time dependent dielectric breakdown is caused by formation of a conducting path through the gate oxide to substrate due to electron tunneling current. TDDB has become increasingly severe as the thickness of the gate oxide decreased due to continuous technology downscaling. Under the same stress conditions, devices can feature directly hard breakdown or several soft breakdown events before the final hard breakdown [31]. While in this paper we utilize a recently proposed model [32], the proposed reliability evaluation methodology is flexible and can be changed by replacing equation 1 with different models as they are discovered.

1) *TDDB Lifetime Model*: The model for  $MTTF_{TDDB}$  is described by the following expression [32]:

$$MTTF_{TDDB} \propto \left(\frac{1}{A}\right)^{\frac{1}{\beta}} (F)^{\frac{1}{\beta}} V_{gs}^{a+bT} e^{\left(\frac{c}{T} + \frac{d}{T^2}\right)} \quad (1)$$

where  $A$  is the transistor's gate oxide area,  $\beta$  is the Weibull slope parameter,  $F$  is cumulative failure percentile,  $T$  is temperature, and  $V_{gs}$  is gate source voltage of the MOSFET. Model fitting parameters  $a$ ,  $b$ ,  $c$ ,  $d$ ,  $\beta$ , and  $F$  are determined from experimental data. In this paper, we utilize typical values of these parameters [32]:  $\beta = 1.2$ ,  $F = 0.01\%$ ,  $a = -78$ ,  $b = 0.081$ ,  $c = 8.81 \times 10^3$ , and  $d = -7.75 \times 10^5$ .

### C. Negative Bias Temperature Instability (NBTI)

Negative bias temperature instability mainly affects PFETs, when they are stressed at large negative gate voltages and high temperatures. NBTI manifests as a gradual increase in the threshold voltage and consequent decrease in drain current and transconductance. The degradation exhibits logarithmic dependence on time. This effect has become more severe with technology downscaling, with the increase of the electric field applied to the gate oxide, and with the decrease of operating voltages.

1) *NBTI Lifetime Model*: The model for  $MTTF_{NBTI}$  is described by the following expression [28], [29]:

$$MTTF_{NBTI} \propto V_{gs}^{-\frac{1}{\beta}} \left[ \frac{1}{1 + 2e^{-\frac{E_1}{kT}}} + \frac{1}{1 + 2e^{-\frac{E_2}{kT}}} \right]^{-\frac{1}{\beta}} \quad (2)$$

where  $k$  is Boltzmann's constant, and  $E_1$ ,  $E_2$  are material and oxide electric field dependent parameters. In addition,  $E_2$  is a voltage dependent parameter and therefore it depends on the operation of circuit. Values of  $E_1$  and  $E_2$  are given by:

$$E_1 = E_{it} - E_g + E_F \quad (3)$$

$$E_2 = E_{fx} - E_F + \gamma E_{ox}^{\frac{2}{3}} \quad (4)$$

where  $E_{it}$  and  $E_{fx}$  are the trap energy level at the oxide/Si interface and the trap energy in the oxide, respectively.  $E_F$  is

Fermi energy,  $\gamma$  is a constant,  $E_{ox}$  is the applied electric field across the gate and can be computed as follows [32]:

$$E_{ox} \approx \frac{V_{gs} - 0.2V}{t_{ox}} \quad (5)$$

#### D. Electromigration (EM)

Electromigration is generally considered to be the result of momentum transfer from the electrons, which move in the applied electric field, to the ions which make up the lattice of the interconnect material. As a result, ions get dislocated from their original positions and migrate along the interconnect. Over time this phenomenon knocks a significant number of atoms far from their original positions. Failure results either from voids growing over the entire line width that cause breaking of the line or extrusions or hillocks that cause short circuits to neighboring lines.

1) *EM Lifetime Model*: EM has an exponential dependence on temperature. The model for  $MTTF_{EM}$  is based on Black's equation [5], [12] and is described by the expression below. This model is widely adopted and studied for a long time [13]. Its limitations depend on the probability distributions that one assumes for this failure mechanism; it is widely accepted that a lognormal distribution is more realistic [12].

$$MTTF_{EM} \propto (J - J_{crit})^{-n} e^{\frac{E_{aEM}}{kT}} \quad (6)$$

where  $J$  is the current density in the wire,  $J_{crit}$  is the critical current density required for electromigration,  $E_{aEM}$  is the activation energy for electromigration,  $k$  is the Boltzmann's constant, and  $T$  is the absolute temperature in Kelvin.  $n$  and  $E_{aEM}$  are constants. We use 1.1 for  $n$  and 0.9 for  $E_{aEM}$  as modeled in RAMP. Notice that  $J$  is usually 2 orders of magnitude higher than  $J_{crit}$  in interconnects; hence, we approximate  $J - J_{crit} \approx J$  [12], [15].

#### E. Thermal Cycling (TC)

Degradation due to each temperature cycle accumulates in time and can potentially lead to permanent damage. The effect is mostly seen in the package and die interface. The package is affected with two types of thermal cycles: (1) Large thermal cycles that occur a few times a day like powering up and down or going into stand-by mode. (2) Small cycles that occur a few times a second. These are due to changes in workload behavior and context switching. The effect of small thermal cycles at high frequencies has not been well studied by the packaging community, and valid models are not available. Hence, we do not consider models for the reliability impact of small thermal cycles, which is a limitation of the model that we adopt below.

1) *TC Lifetime Model*: The model for  $MTTF_{TC}$  is described by the following expression [15]:

$$MTTF_{TC} \propto \left( \frac{1}{T - T_{ambient}} \right)^q \quad (7)$$

where  $T$  is the average temperature of the structure, and  $T_{ambient}$  is the ambient temperature. Notice that  $(T - T_{ambient})$  models the thermal cycle.  $q$  is Coffin-Manson exponent, and for the package it is equal to 2.35 [15].

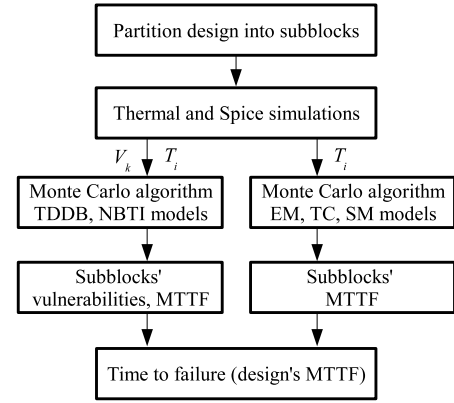


Fig. 1. Top level block diagram of the proposed reliability evaluation methodology.

#### F. Stress Migration (SM)

Mechanical stress because of different thermal expansion rates of different materials in devices and circuits can lead to stress migration. This mechanical stress is proportional to the change in the temperature which is measured with respect to the stress free temperature of the metal. In general, SM is a phenomenon where the metal atoms in the interconnects migrate. It can lead to open circuit, or increased resistance.

1) *SM Lifetime Model*: The model for  $MTTF_{SM}$  is described by the following expression [15]:

$$MTTF_{SM} \propto |T_0 - T|^{-n} e^{\frac{E_{aSM}}{kT}} \quad (8)$$

where  $T$  is the operating temperature,  $T_0$  is the stress free temperature,  $n$  and  $E_{aSM}$  are material dependent constants. We utilize a value of 2 for  $n$ , 0.9 for  $E_{aSM}$ , and 500K for  $T_0$  as advised in [12], [15].

Finally, we would like to emphasize that while the models described by equations 1 through 8 may have limitations and that enhanced models are proposed continuously by the research community, the proposed reliability evaluation methodology is flexible in that once improved models are discovered one can plug these new models in our framework for re-evaluation and to achieve an updated picture of reliability.

### III. PROPOSED RELIABILITY EVALUATION METHODOLOGY

The block diagram with the flow chart of the proposed reliability evaluation methodology is shown in Fig.1 while the corresponding pseudocode is shown in Fig.2. The salient features of our methodology are as follows. First, in order to deal with complexity due to circuit size we adopt a divide and conquer approach. The hierarchy of the structure of a design is partitioned to *zoom-in* to lower levels where the analysis is tractable within reasonable computational time. Second, similar to MaCRO method [29], [30], we employ subblock level Spice simulations to derive transistor operating parameters. However, we conduct Spice simulations at realistic temperatures (different subblocks have different temperatures) rather than at a single worst-case temperature for the entire system as it is done pessimistically in [29], [30]. Third, we model failure times using Weibull and lognormal distributions

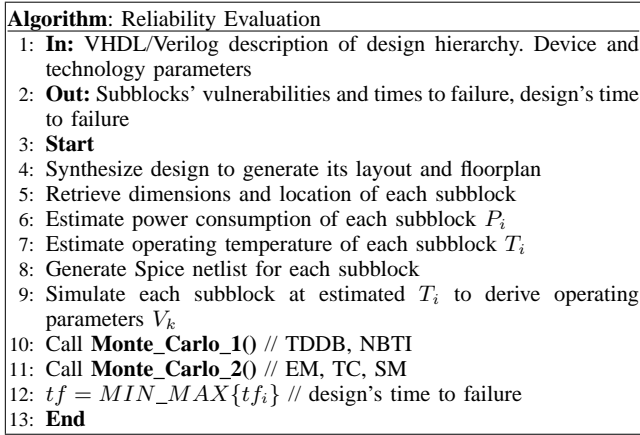


Fig. 2. Pseudocode description of the proposed reliability evaluation methodology.

that have been found to better fit experimental data [12]. Fourth, the block level reliability (as MTTF) is estimated via Monte Carlo simulations, which capture the combined effects of all the aging mechanisms considered. This process is implemented such that the design hierarchy is *zoomed-out* back to upper levels. Finally, as it will be discussed in the next section the proposed method has the ability to identify the most vulnerable subblocks from a reliability point of view.

The output of the proposed reliability evaluation methodology consists of the actual estimate of the time to failure<sup>1</sup> or MTTF of the design (line number 12 in Fig.2) and vulnerabilities of each individual subblock as percentage of transistors with average failure time shorter than the selected threshold (discussed in the next subsection). MTTF is estimated using a *MIN\_MAX* type of analysis similar to [16] in order to be able to handle redundant subblocks that may be introduced for improving reliability via, for example, redundancy based fault tolerance techniques.

Because of the hierarchical approach and of the Spice level simulations, the proposed reliability evaluation methodology enjoys the benefits of both RAMP like and Spice simulation based reliability evaluation approaches discussed in the first section. In the next subsections, we describe the two Monte Carlo (MC) algorithms from Fig.2. In the case of TDDB and NBTI failure mechanisms, the first MC algorithm works at the device level where operating temperatures and voltages are utilized. The remaining failure mechanisms, EM, TC, and SM, are modeled at the subblock level in the second MC algorithm where only operating temperatures are utilized.

#### A. Reliability Evaluation: TDDB and NBTI Failure Mechanisms

The block diagram that illustrates the main steps of the proposed reliability evaluation methodology to address TDDB and NBTI failure mechanisms is shown in Fig.3. Additional details are provided by the pseudocode description from Fig.4. Following the flow chart from Fig.3, the main steps of the proposed reliability evaluation methodology are as follows:

<sup>1</sup>In this paper, we utilize mean time to failure (MTTF), lifetime, and time to failure interchangeably. All of these bear the meaning of *mean* of the probability distribution assumed to model the *failure time random variable*.

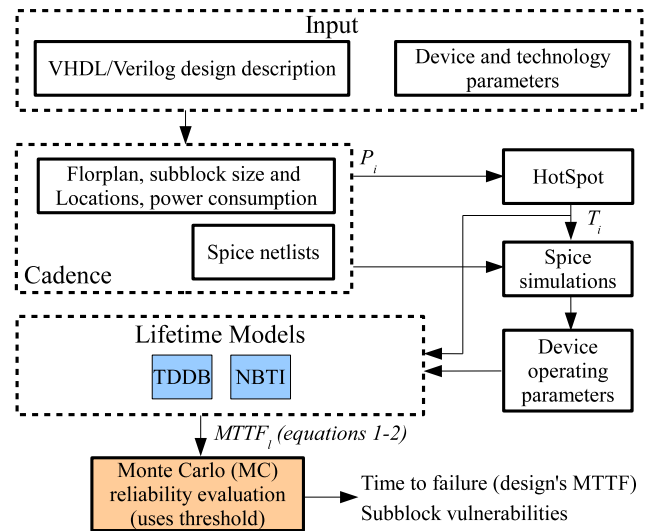


Fig. 3. Flow chart of the proposed reliability evaluation methodology for TDDB and NBTI failure mechanisms.

*Step 1:* We start from a given hierarchical description of the design under consideration. This description can be in any hardware description language such as VHDL or Verilog. In addition, transistor and technology parameters are assumed to be given based on the technology node in which the design is to be fabricated.

*Step 2:* The design is synthesized, placed, and routed using Cadence tools [44], but any other CAD tool can be utilized. The resulting layout represents the block level floorplan, which is divided into individual structures or subblocks based on the initial structural description of the design. In this way, we basically obtain for each subblock its layout, location, and aspect ratio. In addition, power consumption estimates are also generated using Cadence tools.

*Step 3:* The floorplan and power estimates are then fed into HotSpot [45]. HotSpot is an accurate and fast thermal model based on an equivalent circuit of thermal resistances and capacitances that correspond to microarchitecture blocks. The output of the HotSpot simulation is a list with temperatures of each subblock. Our approach addresses one of the limitations of MaCRO like methods [29], [30]. As mentioned earlier, instead of doing worst-case temperature simulations we work with the actual operating temperature for each subblock. In addition, we utilize Weibull and lognormal rather than exponential distributions. Therefore, reliability of each subblock can be evaluated more accurately.

*Step 4:* These temperatures are utilized together with circuit netlists generated from within Cadence tools to perform subblock level Spice simulations. These simulations provide us with the transistor operating parameters necessary to be plugged into the equations modeling the wearout mechanisms described in Section II. It is important to note that the level of design hierarchy at which this is done directly impacts the computational runtime, which increases with subblock-circuit size.

*Step 5:* At this stage we have everything that is needed by the lifetime failure models described by equations 1 and 2 (or equations 6, 7, and 8 utilized by the algorithm described in

**Algorithm: Monte\_Carlo\_1()**

```

1: In: Subblock level temperatures and device level operating volt-
ages
2: Out: Subblocks' vulnerabilities and times to failure due to TDDB
and NBTI
3: Estimate times to failure  $tf_i$  and percentage of vulnerable trans-
istors:
4: for  $i \leftarrow 1$  to  $S$  do //  $S$ : number of subblocks
5:   for  $l \leftarrow 1$  to  $F$  do //  $F$ : number of failure types
6:     Calculate  $MTTF_l$  using equations 1,2 from Section II
7:     for  $j \leftarrow 1$  to  $N$  do //  $N = 10^7$  Monte Carlo iterations
8:        $tf_{min}^j \leftarrow INF$  // Initialize
9:        $Counter_j \leftarrow 0$  // Initialize
10:      for  $k \leftarrow 1$  to  $D$  do //  $D$ : number of devices
11:         $tf_k \leftarrow generate\_sample(MTTF_l)$ 
12:        if  $tf_k < tf_{min}^j$  then
13:           $tf_{min}^j = tf_k$ 
14:        end if
15:        if  $tf_k < threshold$  then
16:           $Counter_j = Counter_j + 1$ 
17:        end if
18:      end for
19:       $Frac\_below\_threshold_j = 100 \cdot \frac{Counter_j}{D}$ 
20:    end for
21:     $tf_l = \frac{\sum_j^N tf_{min}^j}{N}$ 
22:     $Frac\_below\_threshold = \frac{\sum_j^N Frac\_below\_threshold_j}{N}$ 
23:  end for
24:   $tf_i = MIN\{tf_l\}$ 
25: end for

```

Fig. 4. Pseudocode of the device level Monte Carlo algorithm, Monte\_Carlo\_1() from Fig.3.

the next subsection). At the core of the proposed methodology we employ a Monte Carlo simulation algorithm (see Fig.4) implemented and run in Matlab [46]. Our technique is inspired from the RAMP method [15], [17], [26] but executed at the subblock level where the elementary unit is the device or transistor.

The MC algorithm proceeds with the following main steps (1) For each failure mechanism run  $N = 10^7$  simulations: (a) for each transistor, generate failure time *samples* from the corresponding distribution and (b) use MIN analysis of these times by assuming the subblock as a series system to calculate the time to failure  $tf_{min}^j$  of simulation  $j = 1, \dots, N$ . (2) Calculate the overall subblock time to failure for the current failure mechanism as  $tf_l = (\sum_{j=1}^N tf_{min}^j)/N$ . (3) Calculate the value of the overall subblock's time to failure as the minimum among the failure times due to each failure mechanism.

In our experiments, we found that in order to better differentiate between subblocks one only needs to focus on the most vulnerable transistors in a given subblock. Hence, we introduce a *threshold* that helps to identify transistors whose lifetime *samples* are smaller than this threshold. As an indicator of how vulnerable a subblock is, we calculate the percentage of transistors whose lifetime sample is smaller than the selected threshold. This is illustrated in the pseudocode description of the algorithm presented in Fig.4. The threshold value is selected during the reliability qualification process as a function of the desired expected lifetime. An example is provided in the simulation results section. Computational runtime of the methodology described in Fig.3 is in the order of hours for the design example studied later in the simulation

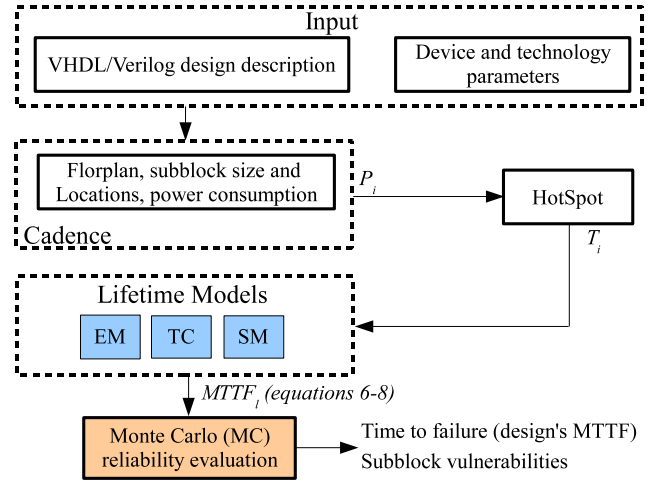


Fig. 5. Flow chart of the proposed reliability evaluation methodology for EM, TC, and SM failure mechanisms.

**Algorithm: Monte\_Carlo\_2()**

```

1: In: Subblock level temperatures
2: Out: Subblocks' times to failure due to EM, TC, and SM
3: Estimate times to failure  $tf_i$ :
4: for  $i \leftarrow 1$  to  $S$  do //  $S$ : number of subblocks
5:   for  $l \leftarrow 1$  to  $F$  do //  $F$ : number of failure types
6:     Calculate  $MTTF_l$  using equations 6,7,8 from Section II
7:      $tf_{min}^j \leftarrow INF$  // Initialize
8:     for  $j \leftarrow 1$  to  $N$  do //  $N = 10^7$  Monte Carlo iterations
9:        $tf_k \leftarrow generate\_sample(MTTF_l)$ 
10:      if  $tf_k < tf_{min}^j$  then
11:         $tf_{min}^j = tf_k$ 
12:      end if
13:    end for
14:  end for
15:   $tf_i = MIN\{tf_l\}$ 
16: end for
17: End

```

Fig. 6. Pseudocode of the subblock level Monte Carlo algorithm, Monte\_Carlo\_2() from Fig.5.

results section. This computational runtime is mainly due to the Spice simulations and does not include the time spent on coding in Verilog the structural description of the design or the synthesis step with Cadence tools.

### B. Reliability Evaluation: EM, SM, and TC Failure Mechanisms

The block diagram that illustrates the main steps of the proposed reliability evaluation methodology to address EM, SM, and TC failure mechanisms is shown in Fig.5 and is similar to that in Fig.3. The main difference is that here the Monte Carlo analysis is done at the subblock level as in the RAMP approach [16]. Therefore, only the HotSpot thermal simulator is utilized to estimate the operating temperature of each subblock. Details of the MC simulation, which bears similarities that from Fig.4, are provided by the pseudocode description from Fig.6. Because in this case we work at subblock level and do not perform Spice simulations, the computational runtime of the methodology described in Fig.5 is in the order of minutes for the design example studied later in the simulation results section.

### C. Discussion

The information acquired from the proposed reliability evaluation methodology described in Fig.2 can be useful to circuit and system designers to develop fault tolerant or robust circuits and systems. Armed with information about what are the reliability critical subblocks and transistors, designers can concentrate their design efforts [47], [48] with wearout mechanism specific techniques only on those, thereby saving area and power. In the next sections we provide two examples of scenarios where the proposed reliability evaluation methodology is utilized to search for lifetime aware floorplans and to investigate NoC routers.

## IV. LIFETIME AWARE FLOORPLANNING

As an example on how the proposed reliability evaluation methodology can be utilized, we propose a lifetime aware floorplanning strategy. The objective of the proposed floorplanning strategy is to seek a floorplan that offers the longest lifetime for the design it represents, aside from optimizing traditional objectives such as total wirelength or area. Because the lifetime estimation procedure (described by the algorithm from Fig.2) has a computational runtime that makes it impractical to be included within the inner loop of the simulated annealing (SA) optimization engine (which may have hundreds or thousands of iterations), we adopt a heuristic approach described in the pseudocode from Fig.7.

<p><b>Algorithm:</b> Lifetime aware floorplanning</p> <ol style="list-style-type: none"> <li>1: <b>In:</b> VHDL/Verilog description of design hierarchy</li> <li>2: <b>Out:</b> Floorplan with longest lifetime and best wirelength and area</li> <li>3: <b>Start</b></li> <li>4: Synthesize design to generate its layout // Cadence tools</li> <li>5: Set <math>N</math>, number of the floorplans to be generated, e.g., <math>N = 100</math></li> <li>6: Set <math>M</math>, number of best floorplans, e.g., <math>M = 5</math></li> <li>7: <b>for</b> <math>i \leftarrow 1</math> to <math>N</math> <b>do</b></li> <li>8:     Set different seed for random number generator</li> <li>9:     Run traditional floorplanner to obtain a new floorplan</li> <li>10:    Keep best <math>M</math> floorplans according to cost function</li> <li>11: <b>end for</b></li> <li>12: <b>for</b> <math>i \leftarrow 1</math> to <math>M</math> <b>do</b></li> <li>13:     Run lifetime estimation algorithm from Fig.2</li> <li>14:     Record the floorplan with longest lifetime so far</li> <li>15: <b>end for</b></li> <li>16: Return floorplan with longest lifetime</li> <li>17: <b>End</b></li> </ol>
--

Fig. 7. Pseudocode of the proposed lifetime aware floorplanning strategy.

The idea is to utilize an existing floorplanning algorithm and run it multiple times starting from different initial conditions and then retain for lifetime evaluation only a smaller number of final floorplans. The following steps describe the proposed lifetime aware floorplanning strategy:

*Step 1:* Start from a given HDL description of the target design and utilize Cadence tools (though any other available tool can be utilized) to generate an initial layout.

*Step 2:* Run traditional floorplanner a large number of times, say  $N = 100$ . We utilize an existing simulated annealing floorplanning algorithm, which works with a B\*Tree representation of the design and with a traditional cost function:  $\alpha \cdot WireLength + (1 - \alpha) \cdot Area$  [49]. Initial conditions are set by resetting with a different seed the internal random number generator utilized to generate random subblock swaps

during the annealing process. In this way, during each run, the floorplanning algorithm arrives to a different final floorplan whose quality thus depends on the initial seed and the selected weight  $\alpha$ . During this step, the best – according to the traditional cost function – say  $M = 5$  floorplans are recorded for processing in the next step. Each of these recorded  $M$  floorplans have already satisfactory wirelength and area.

*Step 3:* Estimate lifetime of each of the best  $M$  floorplans recorded in the previous step using the proposed reliability evaluation methodology from Section III. Record and finally return the floorplan with the longest lifetime.

Given that the B\*Tree floorplanner is very efficient and by keeping  $M$  reasonable small, the proposed lifetime aware floorplanning strategy is an effective approach to generate a floorplan solution that is a good tradeoff between wirelength, area, and reliability. The proposed floorplanning strategy is utilized in the simulation results presented in the next section. Finally, we note that one may want for the floorplanning process to be done such that certain constraints including fixed location or relative position among subblocks are satisfied. In such cases, one only needs to replace in the proposed strategy the floorplanning algorithm with another that is capable of handling such constraints.

## V. SIMULATION RESULTS

In this section, we demonstrate the use of the proposed reliability evaluation methodology and the lifetime aware floorplanning strategy on a Network-on-Chip (NoC) router as a design example. We select the router as our target design because it is the key component of an NoC, which has become the dominant communication paradigm in today's SoCs to cope with the ever increasing complexity of integrated circuits. In addition, the reliability of NoCs has been studied significantly less compared to that of cores. Thus, our objective is to analyze the microarchitecture of a typical NoC router to identify its most vulnerable components and to generate its most reliable floorplan. While our discussion focuses on an NoC router, the entire analysis is applicable to any other block.

### A. Router Architecture

We focus our attention on the popular pipelined router architecture [50] whose block diagram is shown in Fig.8. The main components of this architecture include: routing computation (RC), virtual channel allocation (VA), switch allocation (SA), crossbar switch, input ports, and output ports. We first code the router's structural description in Verilog. Specifics of this description include: 5 input and 5 output ports, 2 virtual channels per port, 4 sets of registers for each virtual channel of each port, and 16 bites wide links. The Verilog description is utilized as input to the proposed reliability evaluation methodology described in Fig.2 as well as the proposed lifetime aware floorplanning strategy from Fig.7.

### B. Technology Node and Set-up Parameters

We utilize Nangate 45nm Open Cell Library [51] within Cadence tools to synthesize and generate the layout of the router. In addition, Cadence tools generate Spice netlists and a

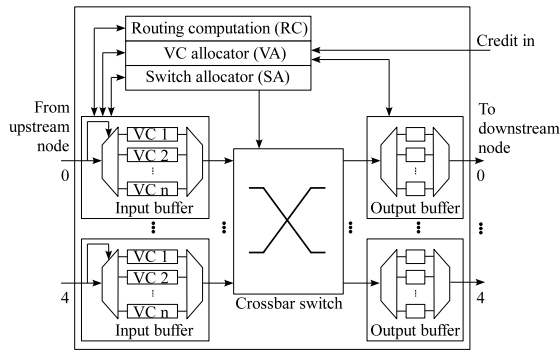


Fig. 8. NoC router architecture.

list of power consumptions for each subblock of the router. The traditional floorplanner utilized in the floorplanning strategy from Fig.7 is set to be run for  $N = 100$  times and  $M = 5$  best floorplans are recorded. The power consumption values and the floorplans are utilized by HotSpot to estimate temperatures of the all subblocks. As mentioned earlier we partition the router into the following subblocks: RC, VA, SA, crossbar, and input and output ports. Spice simulations of all subblock netlists are done at-temperature (as found by HotSpot) to estimate device operating parameters that are utilized inside the algorithm from Fig.4.

The Monte Carlo algorithms introduced in Section III require the generation of lifetime samples (i.e., MTTFs) for devices (Fig.4) or subblocks (Fig.6) from corresponding Weibull or lognormal distributions as modeled in Section II. To do that we utilize Matlab built-in functions. Because in the case of the Weibull distribution, we utilize a value for the shape parameter  $\beta = 1.2$  [20] and have available the mean value  $MTTF$  as computed by the equations from Section II, we need first to compute the scale parameter  $\alpha$  using the equation below to be able to use Matlab built-in functions.

$$\alpha = \frac{MTTF}{\Gamma(1 + \frac{1}{\beta})} \quad (9)$$

where  $\Gamma(\cdot)$  is the Gamma function and MTTF is the mean time to failure of the device/subblock computed by equations 1, 2, 7, and 8.

Because the router architecture has no redundancy (no fault tolerance techniques built-in) the overall lifetime is estimated like for a series system. In other words, the  $MIN\_MAX$  analysis from line number 12 of the algorithm from Fig.2 needs only to take the minimum among all subblocks' MTTFs.

### C. Results

1) *Lifetime Aware Floorplanning and Reliability Evaluation*: Once the layout of the router is generated with the Cadence tools we run the lifetime aware floorplanning algorithm described in Fig.7. The best  $M = 5$  floorplans (the one which turns out with the best MTTF is shown in Fig.9) are recorded for further reliability evaluation, which requires also thermal simulation with HotSpot. As we plan to make publicly available the proposed algorithms, the whole methodology is automated and can be run with a simple Perl script.

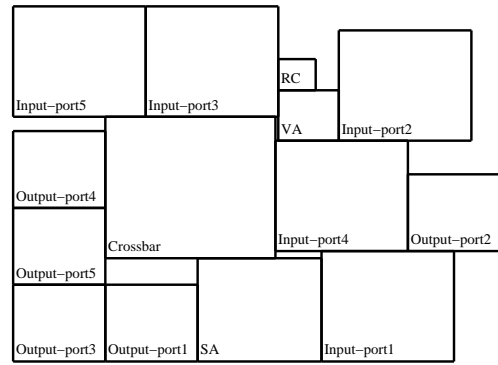


Fig. 9. The best floorplan of the NoC router found by the lifetime aware floorplanning strategy described in Fig.7.

Once the  $M = 5$  best floorplans are found out, we then evaluate each of them to estimate their time to failure. To do that, we utilize the reliability evaluation methodology described in Fig.2. The Monte Carlo algorithms on lines 10 and 11 of Fig.2 and detailed in Fig.4 and Fig.6 estimate the mean times to failure of all subblocks for each of the five best floorplans. These MTTFs are reported in Fig.10. We observe that while the MTTF of each block exhibits a sizable variation among all five floorplans, the relative comparison of MTTFs of different subblocks of a given floorplan stays relatively the same.

Overall MTTFs of all five floorplans are plotted in Fig.10.f. Note that the first floorplan has the longest lifetime and therefore it is identified as the most reliable floorplan for the studied NoC router. Fig.11 shows with how much the first floorplan is better from an expected lifetime perspective compared to the other four floorplans. This figure demonstrates the value of the proposed lifetime aware floorplanning strategy. In this example, the expected lifetime of the first floorplan is with 15% longer than the expected lifetime of the fifth floorplan.

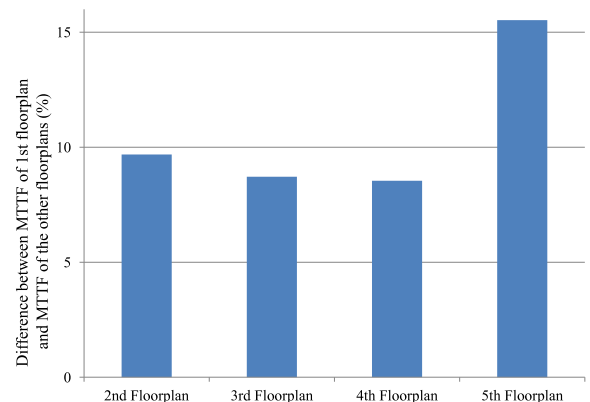


Fig. 11. Illustration of the amount of the improvement in the expected lifetime of the first floorplan compared to the other 4 floorplans.

2) *Vulnerability Analysis*: Note that the reliability evaluation methodology described in Fig.3 provides us with subblock vulnerabilities (computed as percentages of transistors with lifetime shorter than the selected threshold) to TDDB and NBTI failure mechanisms. This information basically helps us identify the most vulnerable subblocks in each floorplan.

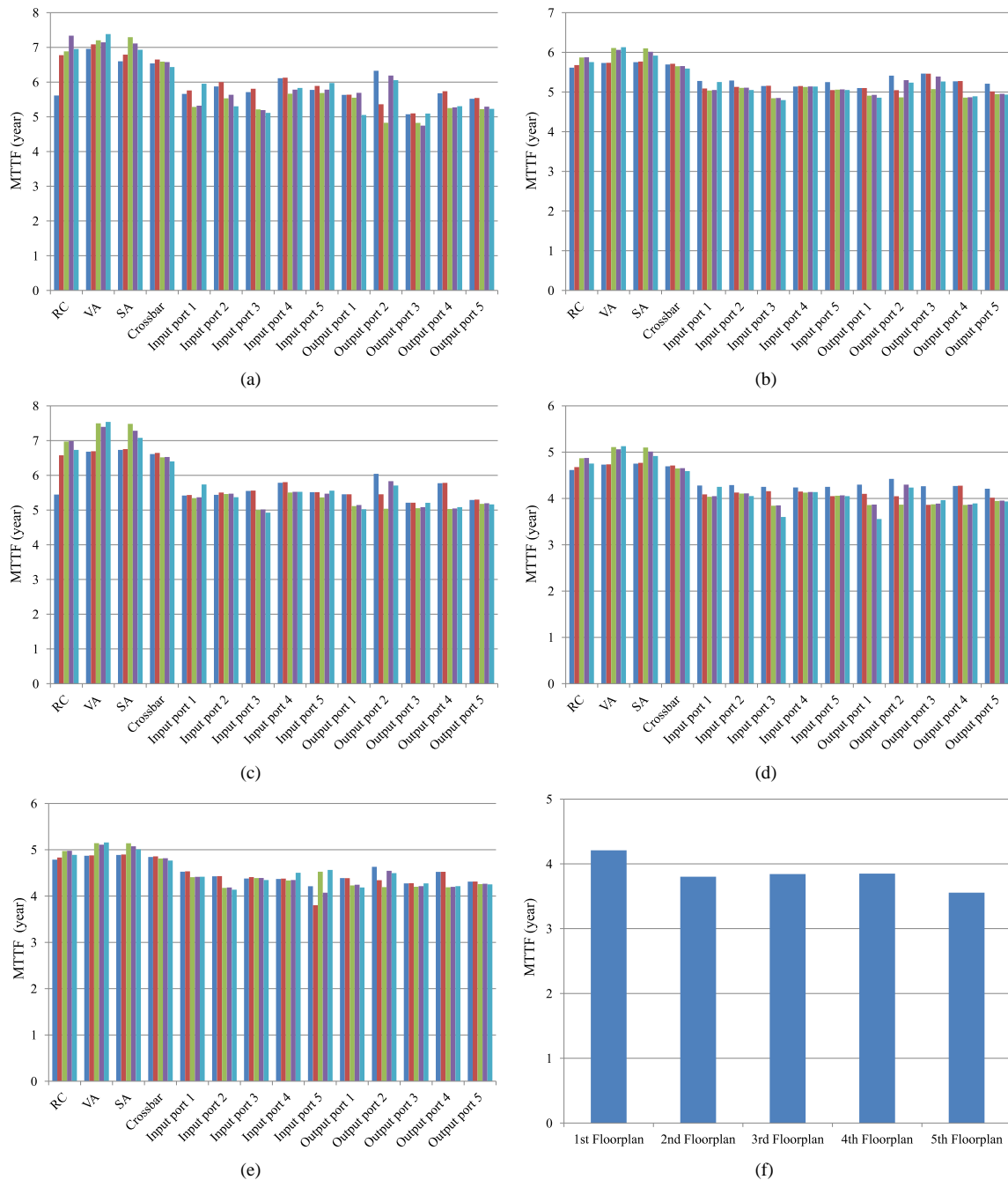


Fig. 10. Mean time to failure of individual subblocks for a) TDDB, b) NBTI, c) EM, d) TC, and e) SM cases. Each of the five bars in each cluster corresponds to each of the five best floorplans. f) Overall MTTF of each of the five best floorplans.

Although such information is not utilized by the lifetime aware floorplanning algorithm, it can prove very useful to system designers who want to develop effective (targeted) resilience techniques. This is the subject of our discussion in this section.

The proposed reliability evaluation methodology provides two types of vulnerability analysis. The first method operates at subblock level and takes into account all failure mechanisms described in Section II. It checks estimated MTTFs of all subblocks for different failure mechanisms and identifies the subblock with the smallest MTTF and its corresponding failure mechanism. For example, applying this method to the case of the first floorplan from Fig.9, the most vulnerable subblock is *output port 5* and the corresponding failure mechanism is

thermal cycling (TC).

The second method of vulnerability analysis operates at device level and only considers TDDB and NBTI failure mechanisms. This method is described in detail in Fig.4. It requires first a *threshold* value, which must be defined by the system designer. This threshold reflects the time until when the system designer expects/hopes that the system will operate correctly without any failure. The main idea of the second vulnerability analysis is to identify and report the subblock that has the highest percentage of transistors with MTTF less than the threshold value. In our example of the NoC router, we select the threshold value to be 8 years. The percentage of vulnerable transistors to TDDB and NBTI failure mechanisms



in each subblock is shown in Fig.12. We observe that RC and VA subblocks contain the highest percentages of transistors with lifetime shorter than the selected threshold despite the fact that their area is smaller compared to for example the area of input registers. This can be explained by the fact that RC and VA components experience higher switching activities compared to the other router components, which in turn leads to higher temperatures. Note that this information could not be obtained with RAMP like reliability evaluation approaches.

It is well known that typically, resilience techniques to harden a system against different failure mechanisms require some form of redundancy. Such redundancy consumes valuable area and power resources, especially for designs with tight area and power budgets. As such, it may not be practical and desirable to develop systems with resilience technique to all types of failure mechanisms. Both vulnerability analysis methods discussed above provide system designers with valuable information about the reliability critical subblocks and transistors. It can help them to concentrate their design efforts on the critical subblocks, thereby saving area and power resources.

## VI. CONCLUSION

We proposed and implemented a new circuit level divide and conquer based reliability evaluation methodology, which enjoys the benefits of transistor level accuracy and of block level efficiency. At the core of the lifetime estimation engine lies a Monte Carlo algorithm which works with failure times modeled as Weibull and lognormal distributions. Using the proposed reliability evaluation methodology we developed a lifetime aware floorplanning strategy. We consider the proposed strategy an important step towards reliability oriented design in general with the potential of improvement via floorplanning. The new floorplanning approach was able to find floorplans with up to 15% difference in the lifetime of a Network-on-Chip router design example. In addition, we applied the proposed reliability evaluation methodology to the router design and identified the routing computation and virtual channel allocation units as the most vulnerable subblocks.

In future work, we plan to apply the proposed reliability evaluation methodology to other design examples and utilize the information provided by this methodology to develop specific fault tolerance techniques targeted at specific parts of the design to improve its lifetime by facilitating localized resilience to selected aging failure mechanisms.

## ACKNOWLEDGMENT

This work was supported by the National Science Foundation (NSF), grant CCF-1116022. Any findings and conclusions or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the NSF.

## REFERENCES

- [1] S. Borkar, "Designing reliable systems from unreliable components: the challenges of transistor variability and degradation," *IEEE Micro*, vol. 25, no. 6, pp. 10-16, Nov. 2005.
- [2] V. Raghunathan, M.B. Srivastava, and R.K. Gupta, "A survey of techniques for energy efficient on-chip communication," *ACM/IEEE Design Automation Conference (DAC)*, pp. 900-905, 2003.
- [3] A. DeHon, H.M. Quinn, and N.P. Carter, "Vision for cross-layer optimization to address the dual challenges of energy and reliability," *ACM/IEEE Design, Automation and Test in Europe Conference (DATE)*, pp. 1017-1022, 2010.
- [4] S. Mitra, K. Brelsford, and P.N. Sanda, "Cross-layer resilience challenges: metrics and optimization," *ACM/IEEE Design, Automation and Test in Europe Conference (DATE)*, pp. 1029-1034, 2010.
- [5] M. White and J.B. Bernstein, "Microelectronics reliability: physics-of-failure based modeling and lifetime evaluation," *Jet Propulsion Laboratory*, California Institute of Technology, Pasadena, CA, JPL publication, Feb. 2008.
- [6] J.H. Stathis, "Reliability limits for the gate insulator in CMOS technology," *IBM Journal of Research and Development*, vol. 46, no. 2/3, pp. 265, 2002.
- [7] D.K. Schroder and J.A. Babcock, "Negative bias temperature instability: road to cross in deep submicron silicon semiconductor manufacturing," *Journal of Applied Physics*, vol. 94, no. 1, pp. 1-18, July 2003.
- [8] S.V. Kumar, C.H. Kim, and S.S. Sapatnekar, "A finite oxide thickness based analytical model for negative temperature bias instability," *IEEE Trans. on Device and Material Reliability*, vol. 9, no. 4, pp. 537-556, Dec. 2009.
- [9] W. Wang, S. Yang, S. Bhardwaj, R. Vattikonda, S. Vrudhula, F. Liu, and Y. Cao, "The impact of NBTI effect on combinational circuit: modeling, simulation, and analysis," *IEEE Trans. on VLSI Systems*, vol. 18, no. 2, pp. 173-183, Feb. 2010.
- [10] S.M. Alam, C.L. Gan, D.E. Troxel, and C.V. Thompson, "Circuit-level reliability analysis of Cu interconnects," *Int. Symposium on Quality Electronics Design (ISQED)*, pp. 238-243, 2004.
- [11] Z. Lu, J. Lach, M.R. Stan, and K. Skadron, "Temperature-aware modeling and banking of IC lifetime reliability," *IEEE Micro*, vol. 25, no. 6, pp. 40-49, Nov./Dec. 2005.
- [12] JEDEC, "Failure mechanisms and models for semiconductor devices," *JEDEC Publication JEP122E*, 2009.
- [13] P.S. Ho and T. Kwok, "Electromigration in metals," *Rep. Prog. Phys.*, vol. 52, pp. 301-348, 1989.
- [14] J. Srinivasan, S.V. Adve, P. Bose, J.A. Rivers, and C.K. Hu, "RAMP: a model for reliability aware microprocessor design," *IBM Research Report*, RC23048, Dec. 2003.
- [15] J. Srinivasan, S.V. Adve, P. Bose, and J.A. Rivers, "The case for lifetime reliability-aware microprocessors," *IEEE Int. Symposium on Computer Architecture*, pp. 276-287, June 2004.
- [16] J. Srinivasan, S.V. Adve, P. Bose, and J.A. Rivers, "Lifetime reliability: toward an architectural solution," *IEEE Micro, Special Issue on Future Trends in Microarchitecture*, vol. 25, no. 3, pp. 70-80, May 2005.
- [17] J. Srinivasan, S.V. Adve, P. Bose, and J.A. Rivers, "Exploiting structural duplication for lifetime reliability enhancement," *IEEE Int. Symposium on Computer Architecture (ISCA)*, pp. 520-531, 2005.
- [18] A.K. Coskun, T.S. Rosing, K. Mihic, G.D. Micheli, and Y. Leblebici, "Analysis and optimization of MPSoC reliability," *Journal of Low Power Electronics*, vol. 2, no. 1, pp. 56-69, Apr. 2006.
- [19] Z. Gu, C. Zhu, L. Shang, and R.P. Dick, "Application-specific MPSoC reliability optimization," *IEEE Trans. on Very Large Scale Integration Systems (TVLSI)*, vol. 16, no. 5, pp. 603-608, May 2008.
- [20] J. Fang and S.S. Sapatnekar, "Scalable methods for analyzing the circuit failure probability due to gate oxide breakdown," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 99, pp. 1-14, Oct. 2011.
- [21] M.R. Choudhury, V. Chandra, K. Mohanram, and R. Aitken, "Analytical model for TDDB-based performance degradation in combinational logic," *ACM/IEEE Design Automation and Test in Europe (DATE)*, pp. 423-428, 2010.
- [22] K. Kang, K. Kim, A.E. Islam, M.A. Alam, and K. Roy, "Characterization and estimation of circuit reliability degradation under NBTI using on-line IDDQ measurement," *ACM/IEEE Design Automation Conference (DAC)*, pp. 358-363, 2007.
- [23] E. Maricaud and G. Gielen, "Efficient reliability simulation of analog ICs including variability and time-varying stress," *ACM/IEEE Design, Automation and Test in Europe Conference (DATE)*, pp. 1238-1241, 2009.
- [24] M. Bashir and L. Milor, "Towards a chip level reliability simulator for copper/low-k backend processes," *ACM/IEEE Design, Automation and Test in Europe Conference (DATE)*, pp. 279-282, 2010.
- [25] J.B. Bernstein, M. Gurfinkel, X. Li, J. Walters, Y. Shapira, and M. Talmor, "Electronic circuit reliability modeling," *Microelectronics Reliability*, vol. 46, no. 12, pp. 1957-1979, Feb. 2006.

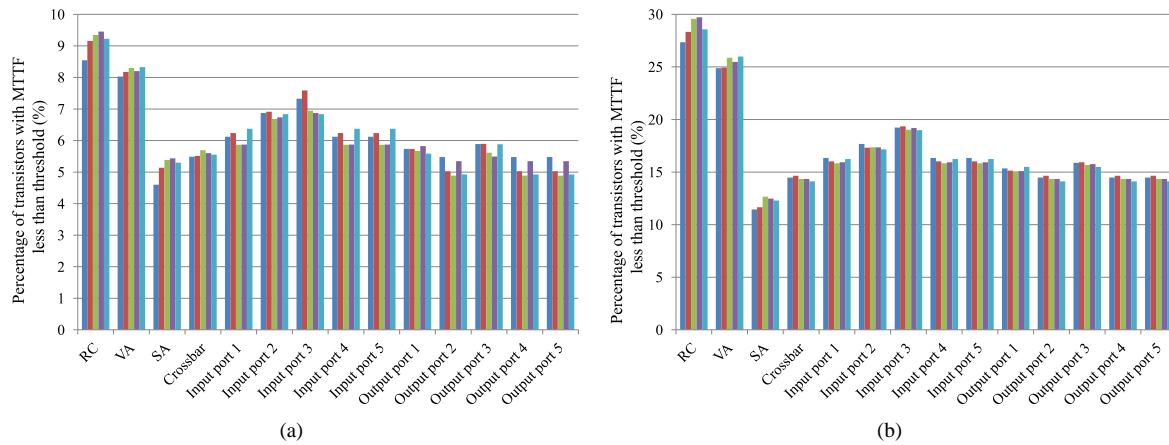


Fig. 12. Percentage of transistors with MTTF value lower than selected threshold for a) TDDB and b) NBTI cases.

- [26] P. Ramachandran, S.V. Adve, P. Bose, J.A. Rivers, and J. Srinivasan, "Metrics for architecture-level lifetime reliability analysis," *IEEE Int. Symposium on Performance Analysis of Systems and Software (ISPASS)*, pp. 202-212, 2008.
- [27] X. Li, B. Huang, J. Qin, X. Zhang, M. Talmor, Z. Gur, and J.B. Bernstein, "Deep submicron CMOS integrated circuit reliability simulation with SPICE," *IEEE Int. Symposium on Quality Electronic Design (ISQED)*, pp. 382-389, 2005.
- [28] S. Zafar, "Statistical mechanics based model for negative bias temperature instability induced degradation," *J. Appl. Phys.*, vol. 97, no. 1, pp. 1-9, Jan. 2005.
- [29] X. Li, J. Qin, B. Huang, X. Zhang, and J.B. Bernstein, "SRAM circuit-failure modeling and reliability simulation with SPICE," *IEEE Trans. on Device and Materials Reliability*, vol. 6, no. 2, pp. 235-246, June 2006.
- [30] X. Li, J. Qin, B. Huang, X. Zhang, and J.B. Bernstein, "A new SPICE reliability simulation method for deep submicrometer CMOS VLSI circuits," *IEEE Trans. on Device and Materials Reliability*, vol. 6, no. 2, pp. 247-257, June 2006.
- [31] A. Ghetti, Gate oxide reliability: physical and computational models, Springer Series in Materials Science, 2004.
- [32] X. Li, J. Qin, and J.B. Bernstein, "Compact modeling of MOSFET wearout mechanisms for circuit-reliability simulation," *IEEE Trans. on Device and Materials Reliability*, vol. 8, no. 1, pp. 98-121, March 2008.
- [33] Y. Han and I. Koren, "Simulated annealing based temperature aware floorplanning," *Journal of Low Power Electronics*, vol. 3, no. 2, pp. 141-155, Aug. 2007.
- [34] C.C. Ta, X. Zhang, L. He, and T.T. Jing, "Temperature aware microprocessor floorplanning considering application dependent power load," *ACM/IEEE Int. Conference on Computer-Aided Design (ICCAD)*, pp. 586-589, 2007.
- [35] W.L. Hung, Y. Xie, N. Vijaykrishnan, C.A. Quaye, T. Theocharides, and M.J. Irwin, "Thermal-aware floorplanning using genetic algorithms," *Int. Symposium on Quality of Electronic Design (ISQED)*, pp. 634-639, 2005.
- [36] K. Sankaranarayanan, S. Velusamy, M.R. Stan, and K. Skadron, "A case for thermal-aware floorplanning at the microarchitectural level," *The Journal of Instruction-Level Parallelism*, vol. 8, pp. 1-16, Oct. 2005.
- [37] J. Kung, I. Han, S. Sapatnekar, and Y. Shin, "Thermal signature: a simple yet accurate thermal index for floorplan optimization," *ACM/IEEE Design Automation Conference (DAC)*, pp. 108-113, 2011.
- [38] V. Nookala, D.J. Lilja, and S.S. Sapatnekar, "Temperature-aware floorplanning of microarchitecture blocks with IPC-power dependence modeling and transient analysis," *Int. Symposium on Low Power Electronics and Design (ISLPED)*, pp. 298-303, 2006.
- [39] H.D. Mogal and K. Bazargan, "Thermal-aware floorplanning for task migration enabled active sub-threshold leakage reduction," *ACM/IEEE Int. Conference on Computer-Aided Design (ICCAD)*, pp. 302-305, 2008.
- [40] S. Yang, W. Wolf, N. Vijaykrishnan, and Y. Xie, "Reliability-aware SOC voltage islands partition and floorplan," *IEEE Symposium on Emerging VLSI Technologies and Architectures*, pp. 343-348, 2006.
- [41] J. Minz, E. Wong, and S.K. Lim, "Reliability-aware floorplanning for 3D circuits," *IEEE Int. SOC Conference*, pp. 81-82, 2005.
- [42] A. Gupta, A. Djahromi, A. Eltawil, N. Dutt, and F. Kurdahi, "Managing leakage power and reliability in hot chips using system floorplanning and SRAM design," *IEEE Int. Workshop on Thermal Investigation of ICs and Systems, (THERMINIC)*, pp. 37-42, 2008.
- [43] J. Shin, V. Zyuban, Z. Hu, J. Rivers, and P. Bose, "A framework for architecture-level lifetime reliability modeling," *IEEE/IFIP Int. Conference on Dependable Systems and Networks (DSN)*, pp. 534-543, 2007.
- [44] www.cadence.com
- [45] <http://lava.cs.virginia.edu/HotSpot>
- [46] <http://www.mathworks.com/products/matlab>
- [47] J. Kim, D. Park, C. Nicopoulos, N. Vijaykrishnan, and C.R. Das, "Design and analysis of a NoC architecture from performance, reliability and energy perspective," *ACM Symposium on Architecture for Networking and Communications Systems (ANCS)*, pp. 173-182, 2005.
- [48] A. DeOrio, D. Fick, V. Bertacco, D. Sylvester, D. Blaauw, J. Hu, and G. Chen, "A reliable routing architecture and algorithm for NoCs," *IEEE Trans. on Computer-Aided Design (TCAD)*, vol. 31, no. 5, pp. 726-739, May 2012.
- [49] T.-C. Chen and Y.-W. Chang, "Modern floorplanning based on B\*-trees and fast simulated annealing," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 25, no. 4, pp. 637-650, April 2006.
- [50] W.J. Dally and B. Towles, "Principles and practices of interconnection networks," *Morgan Kaufmann*, 2004.
- [51] <http://www.si2.org>



**Hamed Sajjadi-Kia** (S'10) received the M.Sc. degree in electrical and electronics engineering from Urmia University, Iran, in 2007. The subject of his Masters research was mixed mode and low power circuit design. He is currently a Ph.D. candidate in the Department of Electrical and Computer Engineering, North Dakota State University, Fargo, ND. His main research interests include fault tolerant and adaptive systems-on-chip, reconfigurable and self-repairing networks-on-chip, and VLSI circuit design.



**Cristinel Ababei** (M'04) received the Ph.D. degree in electrical engineering from the University of Minnesota, Minneapolis, in 2004. He is an Assistant Professor in the Electrical Engineering Department, The State University of New York at Buffalo. Between 2008 to 2012, he was an Assistant Professor in the Electrical and Computer Engineering Department, North Dakota State University. From 2004 to 2008, he worked for Magma Design Automation, Silicon Valley. His research interests include design automation of systems-on-chip with emphasis on reliability, reconfigurable and parallel computing, and optimization of power systems.