Aging analysis at gate and macro cell level

Dominik Lorenz, Martin Barke and Ulf Schlichtmann
Institute for Electronic Design Automation, Technische Universität München, Munich, Germany
Dominik.Lorenz@tum.de, Barke@tum.de and Ulf.Schlichtmann@tum.de

***Abstract*—Aging, which can be regarded as a time-dependent variability, has until recently not received much attention in the field of electronic design automation. This is changing because increasing reliability costs threaten the continued scaling of ICs. We investigate the impact of aging effects on single combinatorial gates and present methods that help to reduce the reliability costs by accurately analyzing the performance degradation of aged circuits at gate and macro cell level.**

I. Introduction

Variability has always been a fact of life in the integrated circuit (IC) industry. Variations can be classified into four categories:

1) Variations in the operating environment - primarily changes in supply voltage and operating temperature.
2) Manufacturing variations - these denote deviations in process parameters from their nominal values that are present in an IC after it has been manufactured. They do not change over time once the IC has been manufactured.
3) Transient faults - single-event upsets (SEUs) typically resulting from ionizing radiation striking an IC.
4) Time-dependent variations - these denote changes in the physical (and consequently, in the electrical) properties of an IC over time caused by aging effects.

Variations in the operating environment are handled during the design process by specifying a range (e.g. $V_{DD,min}$ and $V_{DD,max}$) within which the IC has to meet the specified properties (e.g. frequency or power consumption). Manufacturing variations have traditionally been considered by specifying so-called process corners which describe e.g. for delay the best or worst realistic combinations of process parameters, thus establishing generous guardbands against parameter variations. This modeling is increasingly considered to be problematic, due to two reasons: in more advanced process technologies, parameter variations have been increasing relative to their nominal values, and variations of parameters within one die (which are not addressed by the corner-based design methodology) have gained in importance.

Statistical design methodologies have therefore been proposed as a remedy for dealing with manufacturing variations. Initially, the focus has been on analysis techniques for performance known as statistical static timing analysis (SSTA) [1], which later have been extended to power consumption. Building on these analysis techniques, some optimization approaches have also been suggested. This research has flourished mostly during the last five years. Meanwhile, for the major fundamental problems involved in SSTA (e.g. delay modeling for cells and interconnect; propagation of delay distribution through a circuit; consideration of correlations; integration into the design flow), various solutions have been proposed and a first consensus has developed in the research community about the most promising approaches. A detailed overview of this field is given in [2]. Startups have been created, established EDA vendors have integrated SSTA techniques into their timing analysis tools, and first reports of designers employing these approaches are available since some time already [3].

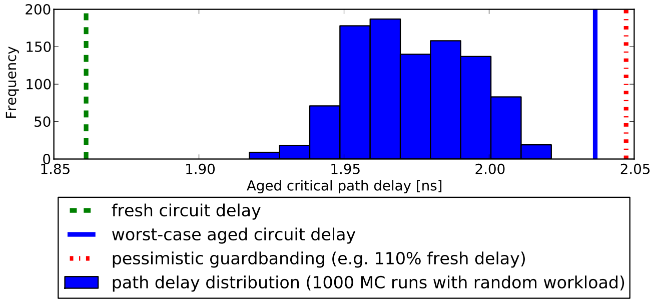


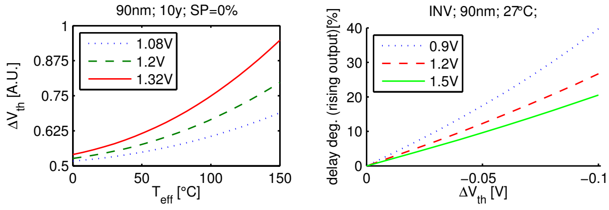
Fig. 1: Aging analysis of ISCAS circuit c7552.

Transient faults have been a problem for memory and they start to threaten logic as well. Several techniques have been published for analyzing the impact of, or reducing an IC's susceptibility to, SEUs [4].

Time-dependent variation, on the other hand has by far not received a similar amount of attention. Additional effort is required, because the performance gain by moving from one technology to the next is shrinking, and the reliability costs, necessary for a fault-free operation under variability, are increasing. Hence, the transition to the next technology may soon be no longer profitable because of the increased reliability costs [5]. To continue scaling as long as possible, the reliability costs have to be reduced.

This work reviews methods to analyze the performance degradation of digital circuits caused by aging. Without an accurate aging-aware STA (in the following also referred to as aging analysis), the impact of aging on circuit delay is hard to estimate and again generous guardbands have to be applied. If those guardbands are too pessimistic (see Fig. 1), area and power is wasted and the product is less competitive. An underestimation of the circuit degradation is even worse. The circuit may fail before the end of its specified lifetime. If the failure is detected, expensive redesigns are necessary. Otherwise, the chip may fail after it is shipped to the customer. Aging analysis enables an accurate estimation of the aged circuit delay. If no information about the workload is available, the worst-case aged circuit delay can be obtained. With workload information, the degradation can be estimated more accu-

978-1-4244-8192-7/10/$26.00 ©2010 IEEE



(a)

(b)

Fig. 2: (a) ΔV_{th} of a PMOS transistor. (b) Simulated sensitivity of an inverter.

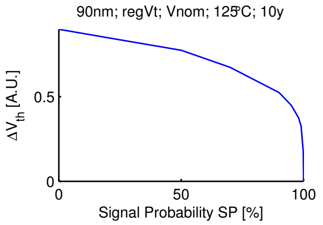


Fig. 3: ΔV_{th} over SP. Drift is maximal for SP=0 %.

rately, because the degradation strongly depends on the input signals over lifetime.

The next section investigates the impact of aging on combinatorial gates. An overview of the state of the art is provided in Sec. III. In Sec. IV and V aging analysis on gate and macro-cell level are presented. Sec. VI gives some results and a conclusion is drawn in Sec. VII.

II. Impact of aging effects on combinatorial gates

First, the impact of negative bias temperature instability (NBTI) and hot carrier injection (HCI) on transistor parameters is investigated. Then, the sensitivity of single gates with respect to a change of transistor parameters is simulated (see Fig. 2).

A. Threshold voltage drift caused by NBTI

Negative bias temperature instability (NBTI) is regarded as the most severe drift-related aging effect nowadays [6]. The impact of NBTI on PMOS transistors is modeled as a threshold voltage drift ΔV_{th}. There is no consensus yet on the physical foundation of NBTI. One popular opinion is that additional states at the interface between silicon and oxide are created. ΔV_{th} has an exponential dependence on temperature and supply voltage (see Fig. 2a) and the time dependence is logarithmic.

NBTI degrades a transistor that is in inversion. To determine the time a transistor is in stress the static signal probability (SP) is used [7]. SP is the average fraction of the clock period a signal is at logic high. Fig. 3 shows ΔV_{th} over SP. The graph neglects an effect called annealing that is not very well understood at the moment. Due to annealing ΔV_{th} recovers when the stress is removed, but there is still a debate whether the drift recovers completely or just partially.

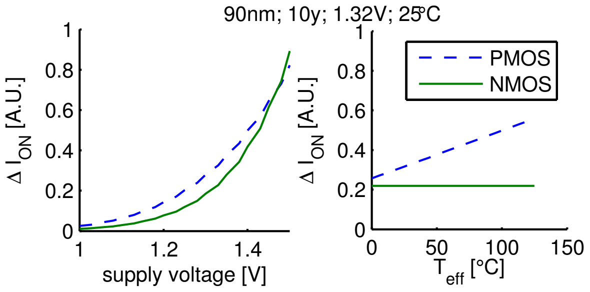


Fig. 4: Voltage and temperature dependence of HCI.

B. Drift of drain saturation current due to HCI

Hot carrier injection (HCI) is getting renewed interest in more recent technologies [8]. It damages a transistor due to accelerated carriers, which can be either holes or electrons. They are accelerated in the lateral electric field and receive enough energy to overcome the potential barrier between the silicon and the gate oxide and leave the channel. The transistor characteristics are degraded by a small portion of those carriers which are caught in the gate oxide. The degradation equation yields a degradation of the drain saturation current ΔI_{on}. HCI has a linear dependency on temperature. The dependency on supply voltage is exponential (see Fig. 4), and the time dependence is logarithmic. A transistor degrades due to HCI when there is a transition from non-conducting to conducting state. Transition density (TD) can be used to determine the time the transistor is in stress. It is defined as the average number of transitions at a net.

C. Sensitivities of single gates

The sensitivities of single gates are simulated using a fan-out-3 test structure (the gate under test has to drive three identical gates and it is driven by a gate that also has to drive three gates). Fig. 5 shows the degradation of the inverter delay for a drift of V_{th} and I_{on}, respectively. The sensitivities can well be linearized until 10 % I_{on} degradation and a 15 % V_{th} drift. These drift values are not reached even for quite demanding operating conditions over lifetime (10 y, 1.32 V and 125 °C). This is important for our proposed aging-aware gate model, because it uses linearized sensitivities.

NBTI only affects PMOS transistors. Therefore, NBTI only degrades the gate delay for a rising input signal transition at single-stage gates. In this case, the pull-up network has to charge the output load. Fig. 2b shows that the sensitivity strongly increases with reduced supply voltage. The sensitivities of different gate types are depicted in Fig. 6a. NOR gates are more sensitive compared to NAND gates or inverters due to their PMOS transistor stack.

D. Technology trend

There is no clear trend observable how parameter drifts are affected by technology scaling. The drifts are strongly dependent on the electric fields. Those fields increased over the last years, because the supply voltage was not scaled as aggressively as the transistor sizes. With the introduction of high-k metal gates the lateral fields decrease again, but

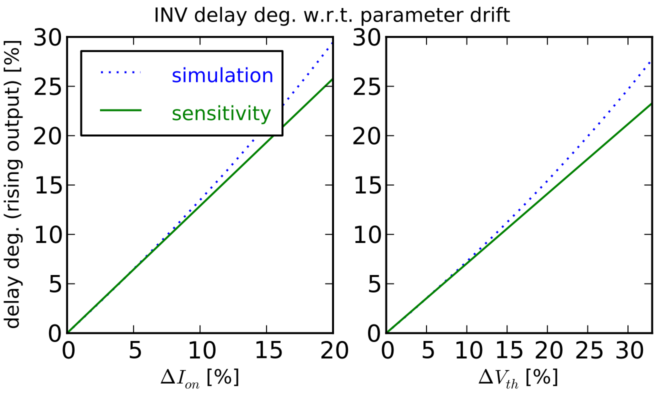
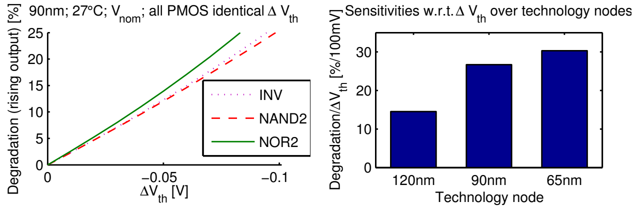


Fig. 5: Degradation of inverter delay by ΔV_{th} and ΔI_{on}



(a)

(b)

Fig. 6: (a) Sensitivities of different gate types. (b) Gate sensitivities for different technology nodes.

also a new aging effect called positive bias temperature instability (PBTI) is becoming relevant.

How the parameter drifts are going to evolve can not be projected well. What can be seen, however, is that the gates are getting more and more sensitive to a drift of transistor parameters for newer technologies. This is caused mainly by the continued reduction of the supply voltage (see Fig. 6b).

Modern circuits often dynamically adapt their supply voltage and frequency to reduce power consumption (DVFS). This complicates the aging analysis, because now the circuit's supply voltage over the lifetime is unknown during the design phase. The worst case condition is, when a circuit operates in a high performance mode for a long period of time and then switches to a low power mode (the transistor parameters drift a lot due to the increased supply voltage in the high performance mode and the circuit is very sensitive in the low power mode).

III. State of the Art

Approaches to analyze the impact of aging effects have been published at transistor and gate level [9]. To the best of our knowledge, there is no model available above gate level, except our aging-aware timing model for macro cells, presented in Sec. V.

A. Approaches on transistor level

Tools, such as BERT [10] or RelXpert [11], work as follows: The fresh circuit is simulated and the relevant current and voltage waveforms at the transistor terminals are stored. Those waveforms are used to generate degraded transistor models for each individual transistor. Finally, the degraded circuit performances are obtained by a second simulation with aged transistors.

Such tools have a very good accuracy if the provided degradation models describe the degradation of the transistor characteristics accurately [12].

B. Approaches on gate level for HCI

Aging analysis tools on transistor level require realistic input signals and the simulation is very time consuming. Hence, they are not applicable for complex digital circuits. Nevertheless, they can be used to characterize aged gate models. The aging-aware gate model GLACIER [13] considers HCI and defines a factor α as follows:

$$\alpha(s_{in}, C_L, TD) = \frac{d_{aged}}{d_0} \tag{1}$$

The aged gate delay d_{aged} and the fresh gate delay d_0 have to be calculated. d_0 is dependent on input slope s_{in} and output load C_L. d_{aged} is also dependent on the transition density TD at the input. For a multi input gate d_{aged} depends on TD at every input. To reduce the complexity, it is assumed that the gate delay for each input can be calculated by considering the contribution from the switching of all inputs as follows:

$$\alpha = \sum_{i=1}^{n} \alpha_i - (n-1) \tag{2}$$

Where n is the number of transistors connected in series and α_i is the contribution of one input pin when just this input switches.

A gate is characterized for one specific use profile. If for example the specified life time changes, the gate has to be re-characterized.

C. Approaches on gate level for NBTI

All proposed models have in common that d_{aged} is the sum of d_0 and the degradation as a function of the threshold voltage drift $\Delta d(\Delta V_{th})$ caused by NBTI:

$$d_{aged} = d_0 + \Delta d(\Delta V_{th}) \tag{3}$$

In [14] the dependence of ΔV_{th} on Δd is obtained during gate characterization. In addition, it is shown that ΔV_{th} does not depend on the actual signal waveform but it is enough to calculate the signal probability SP. [15] derives $\Delta d(\Delta V_{th})$ from the α-power-law. In [16] different temperatures during operation and stand-by are considered for the calculation of ΔV_{th}, and [17] introduces the stacking effect which takes into account that not all transistors in a transistor stack have V_{DD} as their gate source voltage.

All gate models have in common, that they use just one value for ΔV_{th}, although, in general ΔV_{th} differs for different transistors. Either ΔV_{th} is calculated for every transistor and the maximum is taken or the ΔV_{th} of the transistor with an input transition is taken.

IV. Aging analysis on gate level

For an aging-aware STA a gate model is needed which provides aged gate performances (e.g. d_{aged} and aged output slope $s_{out,aged}$). As shown in Sec. II, the aged gate performance is dependent on lifetime t, temperature over lifetime T_{eff}, supply voltage over lifetime V_{eff}, and

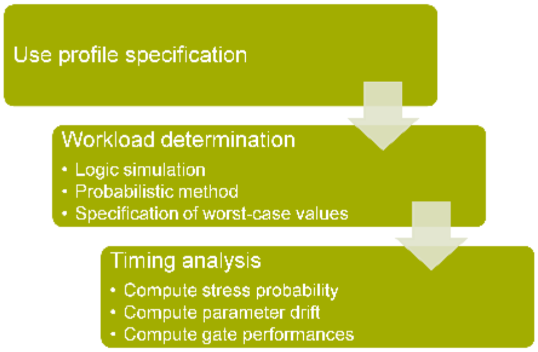

Use profile specification
Workload determination
• Logic simulation
• Probabilistic method
• Specification of worst-case values
Timing analysis
• Compute stress probability
• Compute parameter drift
• Compute gate performances

Fig. 7: Overview of aging analysis flow

workload WL at the gate inputs. WL is given by signal probability SP and the transition density TD. T_{eff}, V_{eff}, and t are combined into the use profile UP. Like the fresh gate performances, the aged gate performances are furthermore dependent on the current supply voltage V and temperature T when the circuit is analyzed.

The next section will give an overview of the aging analysis flow and in section IV-B our proposed aging-aware gate model, AgeGate, is discussed in detail. Compared to previous approaches AgeGate has the following advantages:

- It provides degraded performance considering both dominant aging effects (NBTI and HCI).
- It considers that the transistors of a gate degrade individually.
- The aged output slope is calculated in addition to the aged gate delay.

A. Overview of aging analysis flow

The first step of an aging analysis is to specify the use profile (see Fig. 7). In the current implementation it is assumed that all gates are exposed to the same UP, but it is also technically feasible for each gate to have its individual UP in case there is information about the temperature and the supply voltage distribution.

Next, the workloads for all the nets have to be determined. For this purpose we propose three methods, in order of decreasing accuracy:

- If realistic input vectors are available, a logic simulation can be performed to determine SP and TD for every net.
- Otherwise, if values for SP and TD at the primary inputs are available, a probabilistic method can be applied to propagate SP and TD through the circuit. A survey of probabilistic methods that are also used for power analysis is given in [18].
- In case no specific workload information is available, a worst-case aging analysis can be performed by specifying worst-case values for SP and TD for all nets.

Finally, the actual timing analysis is performed. This step is similar to a traditional STA but an aging-aware gate model is used. For every transistor of a gate the probability that this particular transistor is in stress condition is calculated. Then, the parameter drifts for each transistor are computed and the aged gate delay and output slope are determined.

B. AgeGate: Aging-aware gate model

The gate model [19] is based on the same idea that was also used to determine the impact of aging effects on combinatorial gates in section II. A canonical gate model provides aged gate delay and output slope for given parameter drifts of individual transistors and those drifts are calculated by technology specific degradation equations.

1) Canonical gate model: The aged gate delay d_{aged} is calculated by a first order Taylor approximation at the nominal transistor parameter values:

$$d_{aged} = d_0 + \Delta d \tag{4}$$

$$\Delta d = \sum_{m \in GATE} \left(\frac{\partial d}{\partial V_{th,m}} \cdot \Delta V_{th,m} + \frac{\partial d}{\partial I_{on,m}} \cdot \Delta I_{on,m} \right) \tag{5}$$

d_{aged} is the sum of the nominal gate delay d_0 and the degradation of this delay Δd. $GATE$ is the set of all transistors of the gate. The partial derivatives ${}^{\partial d}/_{\partial V_{th,m}}$ and ${}^{\partial d}/_{\partial I_{on,m}}$ are obtained during the characterization of the gate and $\Delta V_{th,m}$ and $\Delta I_{on,m}$ are the parameter drifts for a transistor m.

The degraded output slope is modeled similarly to the degraded gate delay.

2) Degradation equations: The parameter drifts depend on UP, the transistor sizes W and L and on the stress probability P_{stress}:

$$\Delta V_{th} = f(UP, P_{stress}, W, L) \tag{6}$$

$$\Delta I_{on} = f(UP, P_{stress}, W, L) \tag{7}$$

P_{stress} is the fraction of lifetime that a transistor degrades because the stress conditions for a particular aging effect are fulfilled.

$$P_{stress} = \frac{t_{stress}}{t} \tag{8}$$

3) Stress probability calculation: For every transistor the stress probabilities for NBTI $P_{stress,NBTI}$ and HCI $P_{stress,HCI}$ have to be determined.

P_{stress} is dependent on the voltages at the transistor terminals over the lifetime. The challenge is to calculate P_{stress} from the workload at the gate inputs and the internal gate structure.

Therefore, it is checked when a transistor is in stress condition and how this depends on the workload at the gate inputs. For instance, transistor M_{PB} in Fig. 8 degrades due to NBTI if input A and input B are at logic low. For independent input signals $P_{stress,NBTI}$ for M_{PB} is $(1 - SP_B) \cdot (1 - SP_C)$ [20].

In [19] the calculation of $P_{stress,NBTI}$ and $P_{stress,HCI}$ are derived formally, and it is shown how P_{stress} can be calculated for dependent gate input signals.

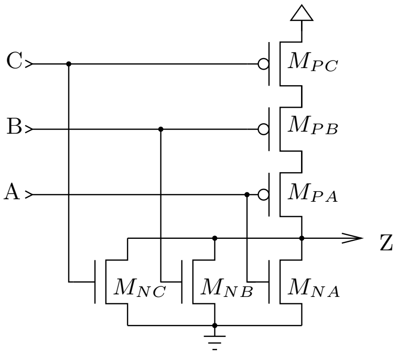


Fig. 8: NOR3 gate

4) Extensions for multi-stage gates: The gate model as described until now is applicable for single-stage gates (e.g. NAND, NOR). For multi-stage gates (e.g. AND, OR) two problems arise: (1) To calculate P_{stress} SP and TD are needed for all nets that are connected to a transistor gate terminal. SP and TD are just available for the gate inputs, but not for internal nets. (2) The signal slope s_{in} at the gate terminal is required for the calculation of $P_{stress,HCI}$. But s_{in} is also just available for gate inputs.

SP and TD for internal nets are obtained by decomposing multi-stage gates into multiple single-stage gates. Then it is possible to propagate SP and TD by a probabilistic method from the gate inputs to the internal nets. The slope of internal nets is obtained during gate characterization. In addition to the output slope the slope of internal nets is characterized as well.

V. Aging analysis of macro cells

To consider aging in an early design phase aging-aware timing models at higher abstraction levels are required.

Macro cells, such as adders or multipliers, are more complex than standard gates. These logical/arithmetic blocks are used to represent a circuit at register-transfer-level. A single value is enough to specify the timing of a macro cell if aging effects are not considered (e.g., adder with 1 ns delay). The maximum delay is determined by the delay of the critical path. Due to aging the path delays increase. Dependent on UP and WL, some paths may degrade faster and others may degrade slower. This means there are multiple possible critical paths for an aged macro cell. Which path becomes the critical aged path depends on the use profile and the workload the macro cell encounters.

An aging-aware timing model at higher abstraction levels enables us to:

- consider the impact of aging on a system early in the design process,
- determine the system performance quickly at the system level,
- perform an extensive exploration of the design space.

Such models can, for instance, be used in high-level synthesis (HLS). One important step in HLS is scheduling. During scheduling, arithmetic/logical operands are mapped on time slots of duration T_0 (see Fig. 9). Therefore, a pre-characterized library with different implementations of macro cells is required. The individual implementations differ in their characteristics (delay, area, power). The schedule is generated by choosing optimal implementations from the pre-characterized library [21].

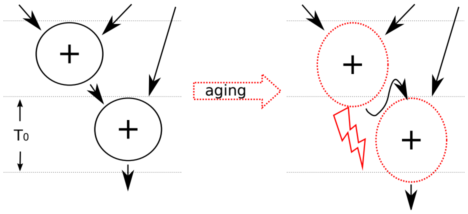


Fig. 9: The dotted circles indicate the aged performances. The circuit fails because the second adder needs the result before the first adder has finished its calculation.

If aging is not taken into account during synthesis, it is possible that the system fails before the end of its specified lifetime because the time of a macro cell for performing a calculation is no longer sufficient. At the moment a macro cell is characterized for the library, it is unknown how the cell will be utilized. Therefore, a timing model is needed which provides the maximum delay of a macro cell as a function of the use profile and workload.

A. Proposed timing model

The proposed timing model [22] takes the internal structure of the macro cell into account. It is based on a strongly reduced timing graph.

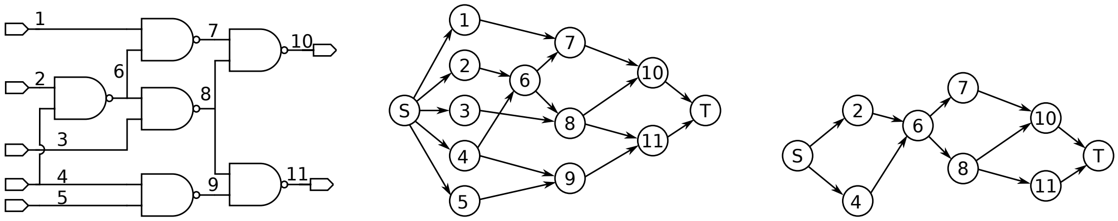
A timing graph (TG) is a weighted directed acyclic graph (see Fig. 10). The nodes represent the nets of the circuit and the edges represent timing arcs. A timing arc is an edge from a gate input net to a gate output net. Edge weights correspond to the cell delay for a signal transition at a particular gate input. Two additional nodes are added to the TG. A source node S is connected to all primary input nodes, and all primary output nodes are connected to a sink node T. The timing model considers different gate delays for rising and falling input transitions.

The basic idea for the macro cell timing model is to take the timing graph of a circuit and to remove all elements that do not belong to a possible critical path. This reduced timing graph no longer represents the entire circuit, but contains only those paths that might become critical for a specific UP and WL (see Fig. 10 (c)). Identifying those edges that are part of a possible critical path plays a key role in this approach.

The timing graph is reduced by successively applying three steps:

- **Block-based reduction step:** A static timing analysis with intervals is performed.
- **Path-based reduction step:** Only those paths can be possible critical paths which have a worst-case aged path delay that is greater than the critical path delay of the fresh (not aged) circuit. All other paths are removed from the TG.
- **Reconvergent fan-out reduction step**: For those paths that share common edges, it is checked whether the faster path is also a possible critical path or not.

The order of the reduction steps is chosen such that the more efficient steps (in terms of time complexity) are



(a) (b) (c)

Fig. 10: The ISCAS'85 circuit c17 is depicted in (a). A timing graph is shown in (b). An example of a reduced TG is shown in (c)

applied first. Hence, less efficient methods work on an already reduced TG.

When the timing model is generated, the use profile and the workload are unknown. Therefore, the exact gate delays are unknown as well. Only a lower and an upper bound for the gate delays can be determined. The delay of a gate has to be at least the fresh gate delay d_0 and can not be greater than the maximum aged gate delay $d_{aged,max}$. For the timing model generation a validity region has to be specified by defining maximum values for temperature, supply voltage and lifetime. These values determine the upper bound $d_{aged,max}$.

When a timing analysis is performed, the timing model of the macro cell has to be evaluated. Now, UP and WL are known and the exact delays for the remaining edges of the reduced TG can be calculated. To determine the workloads at the remaining nodes of the reduced TG, the workloads at the inputs of the macro cell have to be propagated through the TG. This can, for instance, be done with a probabilistic method from [18].

As long as the values for temperature, lifetime and supply voltage for the aging analysis are chosen from within the validity region, the macro cell timing model is as accurate as an aging-aware timing analysis at gate level. This is because only those edges of the TG are removed that are uncritical for the specified validity region. The stronger the TG can be reduced the larger is the speed-up of our proposed timing model compared to an aging analysis at gate level.

B. Block-based reduction step

When lower and upper bounds for the gate delays are available, a block-based static timing analysis (STA) with intervals can be performed. For each node of the timing graph the interval with the minimum and maximum possible arrival time is calculated by propagating the arrival time intervals from the primary inputs to the primary outputs. The two fundamental operations that are necessary for the STA - `sum` and `max` - are defined for intervals as follows:

$$\texttt{sum}([A_l, A_u], [B_l, B_u]) = [A_l + B_l, A_u + B_u] \quad (9)$$

$$\texttt{max}([A_l, A_u], [B_l, B_u]) = [\max(A_l, B_l), \max(A_u, B_u)] \quad (10)$$

For the reduction methods the lesser than operation for intervals is needed, too:

$$[A_l, A_u] < [B_l, B_u] = A_u < B_l \quad (11)$$

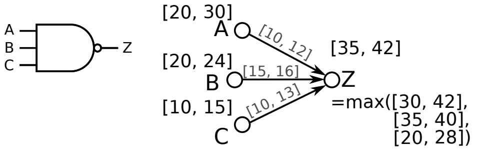


Fig. 11: Edge (C, Z) can be removed because arrival time at Z over this edge is less than arrival time at Z after max-operation (arrival times are black and arc delays are gray).

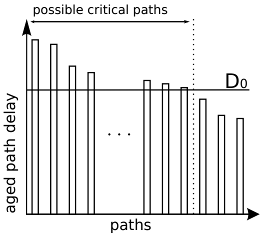


Fig. 12: Path-based reduction step

The greater than operation (>) can be formulated correspondingly.

After the arrival time intervals are calculated for all nodes, the TG can be reduced by removing those incoming edges that do not contribute to the arrival time interval at the node. An edge can be removed if and only if the resulting arrival time interval over this edge is smaller than the actual arrival time interval after the max-operation (see Fig. 11).

The block-based reduction step has a time complexity of $O(n)$, with n being the number of nodes.

C. Path-based reduction step

The path based reduction step exploits that a path can only become the critical path if the worst-case aged path delay D_{age} of this path is greater than the path delay of the fresh (not aged) critical path D_0 (see Fig. 12). The paths are enumerated with respect to D_{age} – from the slowest to the fastest path. To get D_{age} the upper bound of the edge weights must be used for calculating the path delays.

The path-based reduction step has an exponential worst-case time complexity with respect to the number of nodes. Fortunately, for most circuits only a small subset of all paths will have a D_{age} greater than D_0. Hence, the complexity is usually much smaller than the worst-case.

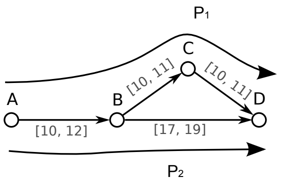


Fig. 13: Example of two paths with common edges. Edges are annotated with gate delays.

D. Reconvergent fan-out reduction step

The last reduction step checks for paths that share common edges whether the faster of both paths (smaller aged path delay D_{age}) is also a possible critical path.

The example in Fig. 13 illustrates the idea. The arrival time intervals at node D for P_1 are $D(P_1) = [30, 33]$ and for P_2 $D(P_2) = [27, 31]$. According to the block-based method, both paths are possible critical paths because the upper bound of $D(P_2)$ is not less than the lower bound of $D(P_1)$. But this consideration is too pessimistic, because for the two path delays different values of the gate delay over edge (A, B) were assumed. For the lower bound of $D(P_1)$ the gate delay was minimal and for the upper bound of $D(P_2)$ the gate delay was maximal. This is impossible. The gate delay is unknown, but for both paths the gate delay of (A, B) must have the same value. Therefore, common edges can be neglected. By just considering the disjoint sub-paths the edge (B, D) can be removed, because $D(P_1 \backslash P_2) = [20, 22] > [17, 19] = D(P_2 \backslash P_1)$.

For this method all paths have to be compared to all faster paths. The worst-case time complexity with respect to the number of nodes is exponential as well.

VI. Results

Transistor parameter drifts and aged signal slopes are mutually dependent. A small experiment should show, whether it is justified to calculate the parameter drifts from fresh output slopes or if a iterative approach is beneficial.

For this purpose a NOR2 ring oscillator is simulated with RelXpert (65 nm, 1.7 V, 145 °C, 700 h). In a first run, the fresh waveforms are used to degrade the transistors. In a second run, aged waveforms are used. The aged waveforms are obtained by simulating the degraded oscillator from the first run. The degradation of the oscillator frequency is 5.35 % and 5.43 %, respectively. An iterative approach would give a value in between. Hence, there is no significant advantage from an iterative approach.

A. Aging analysis on gate level

The ISCAS'85 benchmark circuits are analyzed with our proposed aging analysis flow. The use profile is 10 y, 1.32 V and 125 °C. For the workload either worst-case values are specified or a probabilistic method is used. An industrial 90 nm standard gate library was characterized for 27 °C and 0.9 V.

Fig. 14 shows the degradation of the critical path for different simulator settings. Either just one effect is considered (HCI, NBTI) or both effects (BOTH). To show the importance of taking the aged output slope into account, NO_SLP is an analysis were both effects are considered but no aged slope is calculated. On a 2.4 GHz CPU with 2 GB RAM the analysis of the largest ISCAS circuit took 34 s.

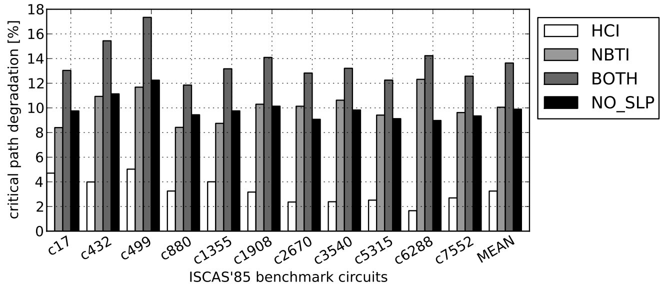


Fig. 14: Comparison of different analyzer settings.

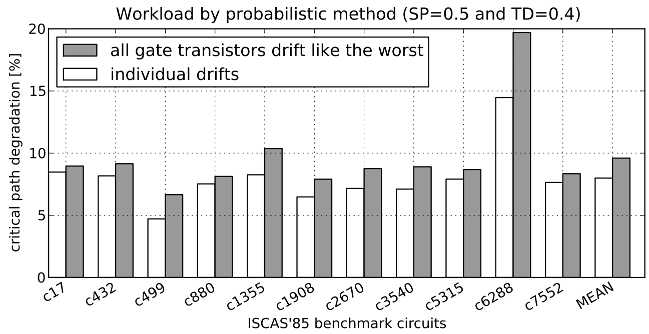


Fig. 15: Aging analysis with individual transistor drifts and with the assumption that all transistors drift maximal.

Fig. 15 shows the difference between calculating individual transistor drifts or assuming that all transistors of a gate degrade equally.

It is difficult to compare the results from our gate model to results from already published aging-aware gate models, because different technologies and degradation equations are used. But the results show that it is important to consider the aged output slope, otherwise, the degradation would be underestimated by 24 %. And by not calculating individual transistor drifts the degradation would be overestimated by 20 %.

B. Aging analysis of macro cells

For our aging-aware timing model for macro cells, we investigate how far the TG can be reduced, because this determines the achievable speed-up. The validity region is 5 y, 1.2 V and 100 °C. We focus our evaluation on the speed-up of the timing model, because as long as the use profile is from within the validity region the accuracy depends just on the used aging-aware gate model.

Fig. 16 shows reduction ratios after applying first the block-based reduction step, then the path-based reduction step and finally the reconvergent fan-out reduction step. The reduction ratio is defined as:

$$\frac{\text{removed nodes (edges)}}{\text{nodes (edges) of original TG}} \tag{12}$$

On average the nodes of the timing graph are reduced by 87 % and the edges by 91 %. This results in a mean speed-up of 27× (maximum speed-up 60×). The generation of

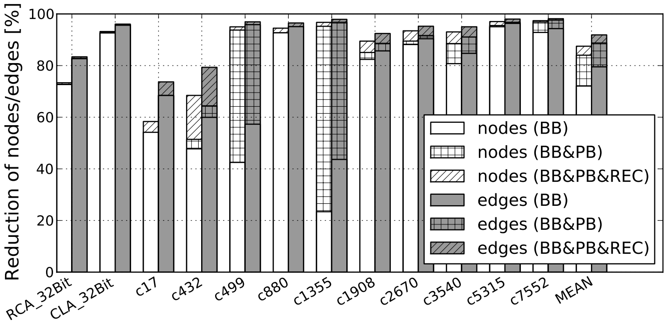


Fig. 16: Achieved reduction ratios with block-based (BB), path-based (PB) and reconvergent fan-out (REC) reduction step.

the timing model took less than 1 min for all circuits except c432 which took 7 min on a 2.67 GHz CPU with 12 GB RAM. Circuit c6288 could not be analyzed because the algorithm was not able to enumerate all necessary paths in the path-based reduction step.

VII. Conclusion

We investigated the impact of aging on combinatorial gates and introduced an aging analysis flow on gate level and an aging-aware timing model for macro cells.

There is no clear trend observable for the parameter drifts caused by aging in recent and future technologies, but the circuit sensitivity with respect to a parameter drift is increasing due to the continued reduction of the supply voltage.

Our aging-aware gate model, AgeGate, considers individual transistor drifts and an aged output slope. Without individual transistor drifts the degradation is overestimated by 20 % and without calculating a degraded output slope the degradation is underestimated by 24 %.

The aging-aware timing model for macro cells provides the maximum gate delay. The key idea is that the timing model just contains possible critical paths. It is as accurate as a timing analysis on gate level, but in average 27× faster.

Acknowledgment

The authors thank our industry partner, Infineon, for the support and the provided data. This work has been supported by the German Ministry of Education and Research (BMBF) within the projects HONEY (Project ID 01M3184) and ROBUST (Project ID 01M3087). The content is the sole responsibility of the authors.

References

[1] C. Visweswariah, K. Ravindran, K. Kalafala, S. G. Walker, S. Narayan, D. K. Beece, J. Piaget, N. Venkateswaran, and J. G. Hemmet, "First-Order Incremental Block-Based Statistical Timing Analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 10, Oct. 2006.

[2] D. Blaauw, K. Chopra, A. Srivastava, and L. Scheffer, "Statistical Timing Analysis: From Basic Principles to State of the Art," *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 4, pp. 589–607, 2008.

[3] N. C. Buck, E. A. Foreman, P. A. Habitz, J. G. Hemmett, S. G. Shuma, N. Venkateswaran, C. Visweswariah, and X. Wang, "Statistical timing: where's the tofu?" in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2009.

[4] N. Miskov-Zivanov and D. Marculescu, "Modeling and Optimization for Soft-Error Reliability of Sequential Circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 5, pp. 803–816, May 2008.

[5] T. Austin, V. Bertacco, S. Mahlke, and Y. Cao, "Reliable Sytems on Unreliable Fabrics," *IEEE Design and Test*, 2008.

[6] J. Hicks, D. Bergstrom, M. Hattendorf, J. Jopling, J. Maiz, S. Pae, C. Prasad, and J. Wiedemer, "45nm Transistor Reliability," *Intel Technology Journal*, vol. 12, no. 2, pp. 131–144, Jun. 2008.

[7] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "NBTI-Aware Synthesis of Digital Circuits," in *ACM/IEEE Design Automation Conference (DAC)*, 2007, pp. 370–375.

[8] V. Huard, C. Parthasarathy, A. Bravaix, C. Guerin, and E. Pion, "CMOS device design-in reliability approach in advanced nodes," in *IEEE International Reliability Physics Symposium*, 2009, pp. 624–633.

[9] Z. Liu, B. W. McGaughy, and J. Z. Ma, "Design tools for reliability analysis," in *ACM/IEEE Design Automation Conference (DAC)*, 2006, pp. 182–187.

[10] R. Tu, E. Rosenbaum, W. Chan, C. Li, E. Minami, K. Quader, P. Ko, and C. Hu, "Berkeley Reliability Tools - BERT," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 12, pp. 1524–1533, 1993.

[11] "Reliability simulation in integrated circuit design," white paper, Cadence Design Systems, Inc., Tech. Rep., 2003.

[12] H. Kufluoglu, V. Reddy, A. Marshall, J. Krick, T. Ragheb, C. Cirba, A. Krishnan, and C. Chancellor, "An Extensive and Improved Circuit Simulation Methodology For NBTI Recovery," in *International Reliability Physics Symposium (IRPS)*, 2010, pp. 670–675.

[13] L. Wu, J. Fang, H. Yonezawa, Y. Kawakami, N. Iwanishi, H. Yan, P. Chen, A. I.-H. Chen, N. Koike, Y. Okamoto, and C.-S. Ye, "GLACIER: a hot carrier gate level circuit characterization and simulation system for VLSI design," in *IEEE International Symposium on Quality Electronic Design (ISQED)*, 2000, pp. 73–79.

[14] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "An Analytical Model for Negative Bias Temperature Instability," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2006, pp. 493–496.

[15] B. C. Paul, K. Kang, H. Kufluoglu, M. Alam, and K. Roy, "Temporal Performance Degradation under NBTI: Estimation and Design for Improved Reliability of Nanoscale Circuits," in *Design, Automation and Test in Europe (DATE)*, vol. 1. Los Alamitos, CA, USA: IEEE Computer Society, 2006, pp. 169–174.

[16] Y. Wang, H. Luo, K. He, R. Luo, H. Yang, and Y. Xie, "Temperature-aware NBTI modeling and the impact of input vector control on performance degradation," in *Design, Automation and Test in Europe (DATE)*. San Jose, CA, USA: EDA Consortium, 2007, pp. 546–551.

[17] H. Luo, Y. Wang, K. He, R. Luo, H. Yang, and Y. Xie, "A Novel Gate-Level NBTI Delay Degradation Model with Stacking Effect," in *Integrated Circuit and System Design. Power and Timing Modeling, Optimization and Simulation*, ser. Lecture Notes in Computer Science, N. Azemard and L. Svensson, Eds. Springer Berlin / Heidelberg, 2007, vol. 4644, pp. 160–170.

[18] F. N. Najm, "A Survey of Power Estimation Techniques in VLSI Circuits," *IEEE Transactions on VLSI Systems*, vol. 2, no. 4, pp. 446 – 455, Dec. 1994.

[19] D. Lorenz, G. Georgakos, and U. Schlichtmann, "Aging Analysis of Circuit Timing Considering NBTI and HCI," in *IEEE International On-Line Testing Symposium (IOLTS)*, Jun. 2009, pp. 3–8.

[20] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "NBTI-aware synthesis of digital circuits," in *ACM/IEEE Design Automation Conference (DAC)*. New York, NY, USA: ACM, 2007, pp. 370–375.

[21] P. Coussy and A. Morawiec, *High-Level Synthesis from Algorithms to Digital Circuits*. Springer, 2008.

[22] D. Lorenz, M. Barke, D. Mueller-Gritschneder, G. Georgakos, and U. Schlichtmann, "Aging model for timing analysis at register-transfer-level," in *ACM/IEEE International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems*, Mar. 2010.