# SPICE-Inspired Fast Gate-Level Computation of NBTI-induced Delays in Nanoscale Logic

Sergei Kostin, Jaan Raik, Raimund Ubar, Maksim Jenihhin
Tallinn University of Technology, Tallinn, ESTONIA
{skostin | jaan | raiub | maksim}@ati.ttu.ee

Thiago Copetti, Fabian Vargas, Leticia Bolzani Poehls
Catholic University – PUCRS, Porto Alegre, BRAZIL
thiago.copetti@acad.pucrs.br, vargas@computer.org,
leticia.poehls@pucrs.br

***Abstract*—Accurate prediction of circuit aging is essential to reliable design, in particular for critical applications. Based on intensive HSPICE electrical simulations, we developed a predictive model to compute NBTI-induced path delay degradation at gate-level. The method is based on a static timing analysis that computes path delay under NBTI-induced $V_{THp}$ (pMOS transistor threshold voltage) degradation. The proposed approach is demonstrated on an industrial ALU circuit design. The obtained results demonstrate a good fitting between the developed model and HSPICE simulations with several orders of magnitude gain in simulation speed.**

***Keywords—Negative Bias Temperature Instability (NBTI), NBTI-induced path delay estimation, static timing analysis, predictive model, aging, logic circuit.***

## I. INTRODUCTION

In Very Deep Sub-Micron technology, lifetime reliability has become one of the key design factors to guarantee *CMOS* integrated circuit robustness. One of the most critical downsides of technology scaling beyond the 65nm node is related to the non-determinism of the devices' electrical parameters due to time-dependent deviations in the operating characteristics of the device. In [1] a brief overview of major sources of time-dependent variations is presented. The most important of them is certainly the *Negative Bias Temperature Instability (NBTI)* [2]. *NBTI* is a physical/chemical effect that causes a degradation of the gate oxide, which results in a drift of the *pMOS* transistor threshold voltage over time. It is a common agreement in the specialized literature that *NBTI* has become the major reliability issue when the gate oxide is thinner than 4 nm [3].

*NBTI* is defined as the effect that occurs when a *pMOS* transistor is negatively biased. The effect manifests itself as an increase of the *pMOS* transistor threshold voltage $V_{THp}$ over time. This results in drive current reduction and noise increase, which in turn causes a degradation of the device delay. The threshold variation is estimated to be 5-15% per year [4], [5] depending on the targeted technology and its environment, while the path delay degradation follows the same trend, though with a smaller magnitude. *NBTI's* effect on the long-term stability of functional logic expresses itself through the incapability of storing a correct value at memory elements such as flip-flops due to the de-synchronization between clock distribution and signal propagation through the logic paths of a circuit. The de-synchronization effect is observed when the summation of all gate delays along a given path is larger than the time slack allocated by the designer for that path. When this condition occurs, the propagated signal reaches the memory elements at the end of the path in a time instant later than the clock front and, as consequence, an incorrect value may be stored at that point of the circuit.

The analysis of the *NBTI* effect is more complicated than other traditional reliability issues, such as hot-carrier injection [3], as it includes stress and recovery phases. The stress phase occurs when a *pMOS* transistor is under a negatively biased condition, i.e., $V_{GS} = -V_{DD}$. In this situation, the interaction between the inversion layer holes and the hydrogen-passivated *Si* atoms breaks the *Si–H* bond generated during the oxidation process. Then, the *H* atoms combines into $H_2$ molecules. When $H_2$ molecules diffuse away, interface traps are left. When interface traps accumulate between silicon and the gate oxide interface, they cause a shift in the threshold voltage $V_{THp}$. However, in the recovery phase, when the biased voltage is removed, a reverse reaction is observed. Some hydrogen diffuses back toward the interface and bonds with *Si*, which reduces the number of interface traps and the *NBTI* effect. Although the recovery phase can partly reverse the *NBTI* effect, it does not eliminate all the interface traps generated during the stress phase, and the *pMOS* threshold voltage will increase over time.

It has been shown that NBTI depends on many factors [6]. One of them is the *signal probability* $P_z$ [7]. The signal probability $P_z$ for a logic gate's input is defined as the ratio of time where that input signal is set to a logic 0. When the input signal probability is 1.0, it means that the gate's input signal is constantly 0, and the *pMOS* transistor is under maximum stress – *static NBTI*. *Dynamic NBTI* occurs when stress and recovery phases are interchanging.

In this work, based on *HSPICE* electrical simulations, we developed a predictive model to compute *NBTI*-induced path delay degradation at gate-level for circuits designed in nanotechnology. In more detail, this method first calculates the delay degradation of each gate using the worst-case duty cycle. This is done by capturing the dependence of gate output delay on $V_{THp}$ shift. Then, based on static timing analysis, this approach computes the aging information for all delay-critical paths of the entire circuit.

978-1-4799-6780-3/15 $31.00 © 2015 IEEE
DOI 10.1109/DDECS.2015.53

The outline of the rest of the paper is as follows: Section II presents an overview of related works. Section III describes the aging characterization for logic gates based on the *HSPICE* electrical simulator, from *Synopsys*. Section IV introduces the proposed predictive models to characterize *NBTI*-induced path delay degradation at gate-level for nano-circuits. Section V presents the experimental results. Finally, Section VI draws the conclusions of the paper.

## II. Related Works

Several previous works have addressed the *NBTI* problem. Among them, [4] proposed an approach as a means of alleviating the *NBTI*-induced aging effects in static random access memories (*SRAMs*). Another approach [8] uses an experimentally verified *NBTI* model to study *DC* noise margins in conventional 6T *SRAM* cells as a function of *NBTI* degradation in the presence of process variations. Considering functional logic, the authors of [9] propose a transistor sizing technique that not only mitigates *NBTI* induced delay of the gate under consideration, but also minimizes its impact on the adjacent gates. This technique seems to be very effective, but it is mandatory to identify the critical gates and paths within the circuit in order to apply such technique. Otherwise, this technique should be applied to all the circuit gates, which would result in an unacceptable area overhead and eventually, excessive power consumption. In [10] the authors present a method characterizing the delay of every gate in a standard cell library as a function of the signal probability ($P_z$) of each of its inputs. In the sequence, a technology-mapping technique that incorporates *NBTI* stress and recovery effects in order to ensure optimal performance of the circuit during its entire lifetime is applied.

Although the calculation process in [7] and [10] is handy as it allows derivation of aging curves for logic components, it is also prohibitively time consuming. This is due to an extremely large number of stress recovery cycles that have to be computed. In this paper we propose the use of *mathematically convenient functions* for characterizing gate aging along the critical paths of logic circuits and calculating the path delays and their degradation caused by *NBTI* aging. Derivation of such functions is based on *SPICE*-guided electrical characterization of logic gates. Experiments with a real *ALU* circuit expose the good match of such gate-level approach to the electrical simulation results with the simulation speed-up of several orders of magnitude.

## III. Gate Aging Charactarezation in SPICE

In order to grab as much information as possible to guide the definition of our predictive models, we have performed intensive *HSPICE* electrical simulations. This simulator is available under the Synopsys design environment. The selected technology to develop our models was the 65nm *Predictive Technology Model* (*PTM*) [11] with 0.9V Vdd and 0.365V $V_{THp}$. For a different technology, the same gate aging characterization procedure in *SPICE* must be repeated.

The *SPICE* simulation process consisted of simulating the basic cells of the technology library for different $\Delta V_{THp}$ (*pMOS* transistor threshold voltage shift) values in order to capture the dependence of gate output delay on $V_{THp}$. Fig. 1 displays the typical *n*- and *p*-networks displaying device interconnections for *INVERTER*, *NAND* and *NOR* gates considered for simulation.

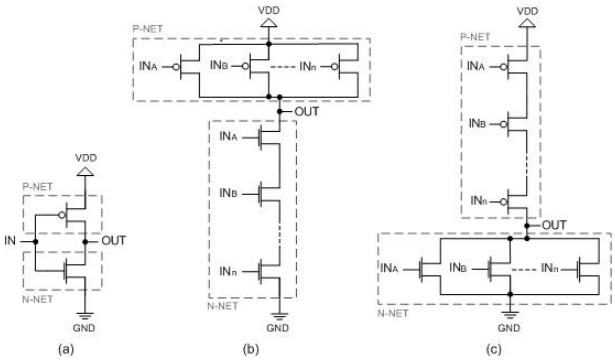


**Fig. 1** Typical n- and p-networks displaying device interconnections for (a) Inverter, (b) NAND and (c) NOR gates considered in this work

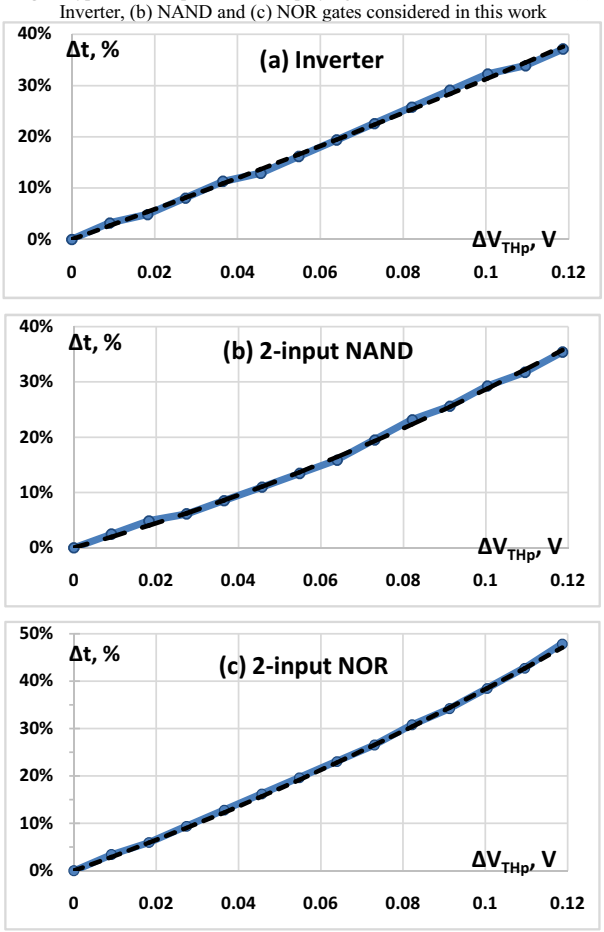


**Fig. 2** Dependence of gate output delay on $V_{THp}$ shift for the rising transition 0→1 in SPICE (blue curve) and the curve function (dashed black) that matches the SPICE results for (a) Inverter, (b) NAND2 and (c) NOR2

Fig. 2 shows the gate-aging characterization step in SPICE for Inverter, NAND2 and NOR2 gates. It captures the dependence of Gate Output Delay on $V_{THp}$ for the rising input transition 0→1. The following mathematically convenient function (dashed black curve) was matched to the curves characterized with SPICE (blue curve) in order to calculate the percental change in gate delay values:

$$\Delta t_{gate} = \lambda \cdot \Delta V_{THp}(x_i) + (\mu \cdot \Delta V_{THp}(x_i))^2 \quad (1)$$

where $\Delta t_{gate}$ is the nominal gate delay increase in percent for the gate, $\Delta V_{THp}(x_i)$ is the change of threshold voltage $V_{THp}$ for *pMOS* transistors at the gate input $x_i$ and λ and μ are technology dependent constants. In our experiments λ and μ are set to 3.1 and 2.7 for the NOR gate, to 2.05 and 2.85 for the NAND gate and to 2.9 and 1.5 for the Inverter, respectively.

Note that only the increased gate delay for the 0→1 transition at gate inputs were characterized as the *SPICE* experiments revealed no deviation, or at least there was a negligible deviation in the input 1→0 transition delay after aging. This can be explained due to the fact that as the *pMOS* device in the *p*-network is getting older, it facilitates the task of discharging the gate output capacitance by the *nMOS* device, placed in the *n*-network. The exception is a *NOR* gate, especially *NOR* with multiple inputs, e.g. *NOR4*, where gate delay degradation for input 1→0 transition became slightly negative, i.e. transition delay decreased compared to the nominal one.

Note that when compared with the *NOR* gate, the *NAND* gate does not display reduction of the output delay because of the *n*- and *p*-network topologies: in the *NOR* gate case, *pMOS* devices are connected in series (resp. in parallel for *NAND* gate), whereas *nMOS* devices are connected in parallel in *NOR* (series in *NAND* gate, see Fig. 1). This device interconnection facilitates the discharge of the gate output capacitance by the *nMOS* devices, while rendering more difficult the task of *pMOS* devices to charge-up the output capacitance in the case of *NOR* gates, that is the reason we observed reduction of the output delay for the fall-edge delay.

## IV. Gate-Level Calculation of NBTI Path Delay

In order to calculate the path delays using the SPICE-inspired characterization, we exploit an approach where for each signal $x_i$, signal probabilities $P_z(x_i)$ are calculated by gate-level logic simulation and the numbers of gate fanouts $F(x_i)$ are derived by structural analysis at the gate level netlist, respectively. These parameters together with the expected operation time in years are applied as an input to the gate-level approach of calculation *NBTI* path delays, which includes also transistor level design of logic gates.

The voltage threshold shift $\Delta V_{THp}(x_i)$ needed for delay degradation evaluation due to *NBTI* aging was calculated by the following mathematically convenient function:

$$\Delta V_{THp}(x_i) = (\alpha \cdot P_z(x_i))^{\beta} \quad (2)$$

where $P_z(x_i)$ is the signal probability for input signal $x_i$ and α and β are technology dependent constants that were set to generate a graph that matches the curves obtained by *NBTI*-aging analysis in [7] for PTM 65 nm technology. In our experiments β is set to 0.18868 and α is set to $15 \cdot 10^{-7}$ for 1 year of aging and to $115 \cdot 10^{-7}$ for 10 years of aging, respectively.

In order to calculate the $\Delta V_{THp}(x_i)$ values for the static case where $P_z(x_i) = 1$, we assigned $\Delta V_{THp} = 0.18$V for 1 year and $\Delta V_{THp} = 0.27$V for 10 years of aging, respectively. Fig. 3 shows the fitting of the mathematically convenient function (2) (the red and blue curves) to the comprehensive analysis of [7] (the white and black dots).

The proposed functions (1) and (2) closely match the *SPICE* electrical characterization and the *NBTI* data from [7], respectively. These functions were implemented in *gate-level aging simulation to provide extremely fast calculation for NBTI-induced delay degradation*.

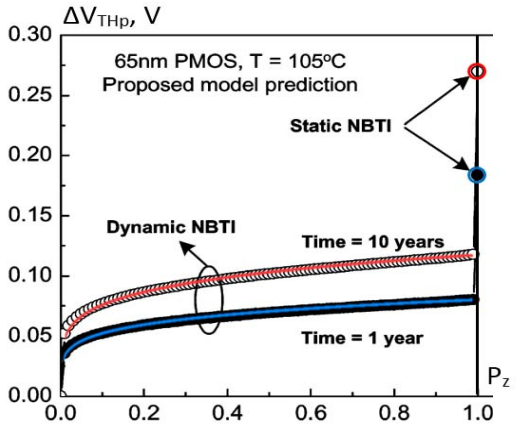


**Fig. 3** Threshold voltage shift $\Delta V_{THp}$ as a function of signal probability $P_z$

### A. NBTI-critical path delay calculation

In the following, a method for fast calculation of the *NBTI*-induced delay degradation at paths of a gate-level circuit is proposed. In the calculation process we use the following notations:

$d(G_k)$ – is the nominal delay of the fresh gate $G_k$ without considering aging, i.e. its delay at time zero;

$\tau(G_{k,i})$ – is the increase in the delay of the gate $G_k$ from the *i*-th input to the output of the gate caused by NBTI-induced aging;

$t(G_{k,i})$ – is the total delay of the gate $G_k$ from the *i*-th input to the output of the gate caused by NBTI-induced aging,

$$t(G_{k,i}) = d(G_k) + \tau(G_{k,i});$$

$t(G_k)$ – is the total maximum delay of the gate $G_k$ over all its inputs $m_k$, when taking into account *NBTI*-induced aging.

$$t(G_k) = \max\ \{t(G_{k,1}), t(G_{k,2}), \ldots, t(G_{k,mk})\};$$

$D(G_k)$ – is the delay calculated for the slowest signal path in the cone $C_{IN}(G_k)$ based on the values of $t(G_{k,i})$ for all gates on this path,

$$D(G_k) = \max\ \{(D(G_i) + t(G_{k,i}))\,|\,G_i \in IN(G_k)\},$$

where $IN(G_k)$ is the set of input gates of $G_k$ , and $t(G_{k,i})$ is the total delay of the gate $G_k$ from the output of the gate $G_i$ caused by aging;

Consider a combinational circuit as a network of gates where all the gates have numbers which show the ranking of gates in a partial order such that:

(1) all the input gates are numbered in an arbitrary order,

(2) all other gates may get their numbers only if all their predecessor gates have already got their numbers.

We present Algorithm 1 for calculating $D(G_k)$, which is based on processing the gate-level netlist, gate by gate, from inputs to outputs. The method calculates the maximal degraded path delay values $D(G_k)$ for all the gates of the circuit based on the estimates of $t(G_k)$, where $NG$ is the number of gates in the circuit.

**Algorithm 1**. NBTI-aware static timing analysis

**FOR** all gates $G_k$, k = 1, 2, …, $NG$:

$$t'(G_{k,i}) = \begin{cases} t(G_{k,i}), & x_i = 0 \\ d(G_k), & x_i \neq 0 \end{cases}$$

$$D(G_k) = \max\ \{D(G_i) + t'(G_{k,i})\,|\,G_i \in IN(G_k)\}$$

**END FOR**

As a result, fast an accurate calculation of NBTI-degraded paths will be performed at the gate-level.

## V. Experimental Results

The proposed approach is demonstrated on the *ALU_4bit* circuit shown in Fig. 4, which is a design of the 4-bit ALU 74HC/HCT181 from Philips [13] with minor modifications: XOR gates are substituted with NAND and Inverter gates.

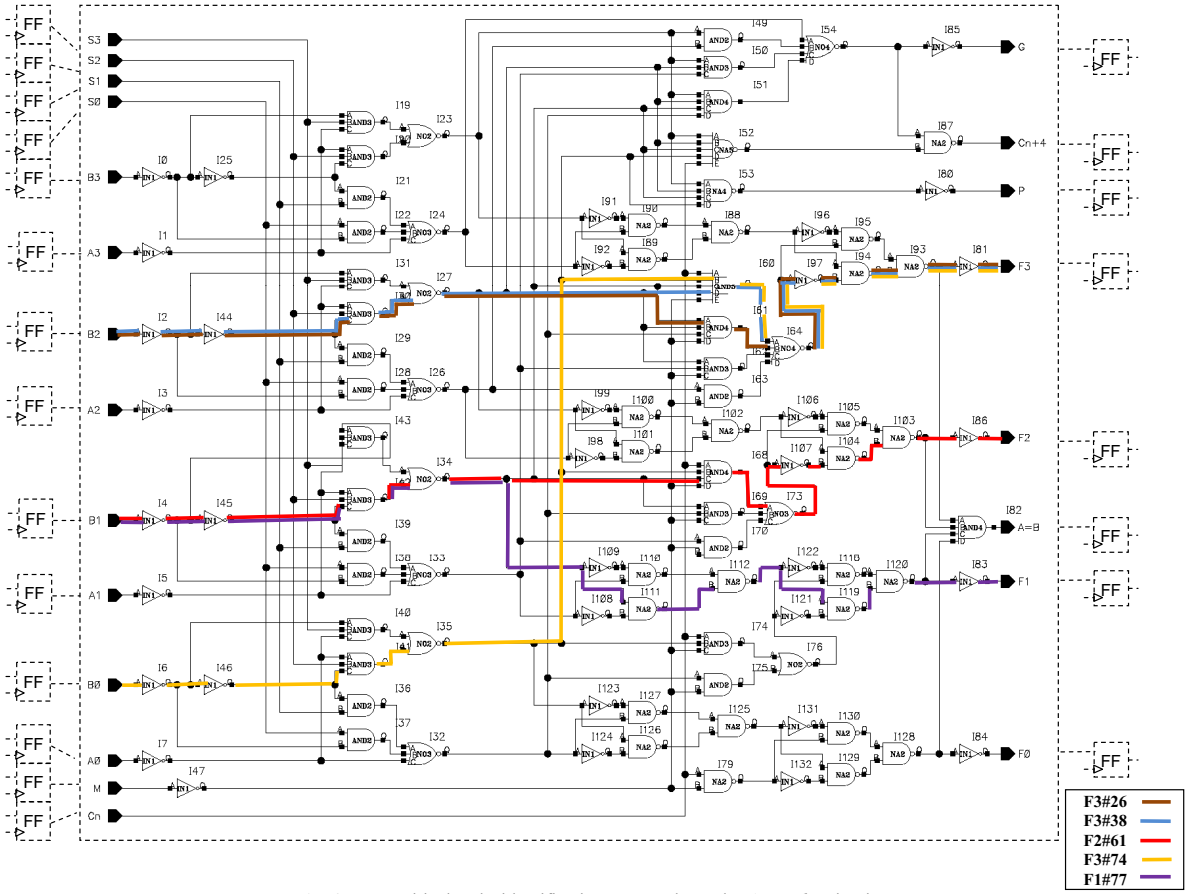


**Fig. 4** NBTI-critical paths identification case study on the *ALU_4bit* circuit.

TABLE I. PATH DELAY DEGRADATION CALCULATED BY GATE-LEVEL ESTIMATION

| Path | Nominal Delay | Path delay after 1 year | | | | Path delay after 10 years | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | 0→1 (rise-edge delay) | | 1→0 (fall-edge delay) | | 0→1 (rise-edge delay) | | 1→0 (fall-edge delay) | |
| | *units* | *units* | Δ, *%* | *units* | Δ, *%* | *units* | Δ, *%* | *units* | Δ, *%* |
| F3#26 | 186.00 | 198.90 | **6.94** | 202.10 | **8.66** | 205.80 | **10.65** | 211.20 | **13.55** |
| F3#38 | 182.00 | 194.50 | **6.87** | 198.00 | **8.79** | 201.20 | **10.55** | 207.00 | **13.74** |
| F2#61 | 175.00 | 186.10 | **6.34** | 191.10 | **9.20** | 192.00 | **9.71** | 200.20 | **14.40** |
| F3#74 | 176.00 | 188.90 | **7.33** | 190.30 | **8.13** | 195.80 | **11.25** | 198.30 | **12.67** |
| F1#77 | 146.00 | 151.80 | **3.97** | 162.60 | **11.30** | 154.70 | **5.96** | 172.00 | **17.81** |
| Average | | | **6.29** | | **9.23** | | **9.62** | | **14.43** |

It is a gate-level combinational logic design (as outlined by the dashed border), where primary inputs and outputs are connected to the flip-flops (FF). For our experiments 5 NBTI-critical paths leading to various outputs have been selected to compare the proposed gate-level *NBTI-induced delay degradation* estimation against SPICE simulation results and estimate the accuracy of the proposed method. These paths are demonstrated in Fig. 4 by bold lines.

Table I shows path delay degradation for selected paths calculated by fast gate-level simulation due to 1-year and 10-years NBTI-induced effect. Column 2 shows the nominal path delays in units. The values were pre-calculated using gate-level relative approach [14] that is technology independent. The units can be converted into nano- or picoseconds applying an appropriate technology specified coefficient. Further, results of path delay degradation for rise-edge and fall-edge transitions of primary input signal are presented: the final path delay in *units* and delay increase in percent Δ*, %*.

So as to perform accurate gate-level estimation, first, input signal probabilities $P_z$ for *pMOS* transistors are calculated. According to the specification of the circuit [13] all input stimuli are possible. Therefore a large number of pseudo-random input patterns (totally 16384 test vectors) has been generated and selected as possible functional sequences for the input stimuli. For large designs where exhaustive logic simulation is infeasible, a sufficient number of pseudo-random or functional patterns still can be simulated or statistical static timing analysis can be performed to obtain signal probabilities $P_z$ . Based on the obtained $P_z$ values voltage threshold shift $\Delta V_{THp}$ for *pMOS* transistors is calculated according to the fitting function (2). This step is performed only at the gate-level and requires 0.27 seconds of CPU time. The resulted $\Delta V_{THp}$ values are further exploited by both gate-level estimation with the fitting function (1) and SPICE simulation to obtain individual gate delays as well as path delay degradation.

For example, the fall-edge delay of path F1#77 after 1 year of aging is increased by 11.30% and after 10 years of aging by 17.81% which the largest increment for given paths, i.e. 172-146 = 26 units. Nevertheless, the final delay of the path (172 units) is even less than nominal delay (186 units) of the most delay critical path F3#26. Therefore, path F1#77 can be excluded from the list of NBTI-critical paths. Moreover, the path delay for rise-edge transition after 1 year and 10 years of aging is 3.97% and 5.96%, respectively, that is very small compared to fall-edge delay for given path. It means that at fall-edge of primary input on F1#77 path *pMOS* transistors are under more stress or belongs to gates that are more likely prone to aging. For instance, according to Fig. 2 if *pMOS* threshold voltage $V_{THp}$ shifts by 0.1V then gate output delay for 2-input NAND grows by 30%, whereas for 2-input NOR the increase is by 38%. On the other, the path delay after 10 years of aging for F3#26 is 10.65% and 13.65%. Assuming that the nominal delay (186 units) of given path is the largest one for all paths in the circuit and the designer allocates time slack equal to 10% of the longest delay, then on the output of given path may be produced an incorrect value after 10 years of exploiting the circuit.

Thus, Table I demonstrates the path delay degradation trend of *pMOS* transistors being under *NBTI*-stress for years.

In order to ascertain in an efficiency and an accuracy of the proposed gate-level delay degradation estimation the selected *NBTI* critical paths shown in Fig.4 were simulated in *SPICE* environment. The *pMOS* transistor threshold voltage shift $\Delta V_{THp}$ values calculated by function 2 were manually inserted in *SPICE* netlist. It takes almost 5 minutes to prepare the netlist for each path. For large circuits that may have hundreds of gates along the path the $\Delta V_{THp}$ values can be inserted automatically using appropriate tools. However, the simulation time for each path was about 1 minute in *SPICE* compared to the proposed gate-level aging delay calculation which required only 0.56 seconds of CPU time (Dual core 2.2 GHz, 4GB RAM) for simulation and estimation of all NBTI-critical paths in the circuit. Thus, the speed-up for given example is almost 1000 times.

For large designs the complexity of SPICE simulation will grow up considerably that excessively negatively affects the simulation times, whereas for the proposed method the CPU time will increase linearly to number of gates in the path. The only thing is that the number of possible NBTI-critical paths will explode enormously that can lead to memory overflow. However, the explosion can be restricted by the limit number of input and output paths for each gate that will be considered

in *NBTI* analysis. At the same time the accuracy of selecting *NBTI* critical paths will remain the same.

Table II demonstrates the comparison between the proposed fast gate-level calculation and actual transistor-level simulation in *SPICE* for the selected paths after 10 years of the *NBTI*-induced effect. Columns 2 and 3 show path delay degradations in percentage for the rising-edge input transition in the case of gate-level calculation and SPICE simulation, respectively. Columns 4 and 5 represent the same for the falling path input transition. For falling edge, the results of the proposed gate-level aging delay estimation correlate very precisely with *SPICE* simulation allowing maximum 1.56% distortion. For rising edge there is constant 4% difference between results that can be explained by incorrect interpretation and calculation of output delay degradation for 2-input *NOR* that has many fan-out branches on its output and which is being under *NBTI* stress. In Fig. 4 for path F3#26 and F3#38 consider gate I27 which has 9 fan-outs, for path F2#61 and F1#77 – gate I34 (9 fan-outs), for path F3#74 – gate I35 (7 fan-outs). After removing this 4% constant difference the results will correlate with the same accuracy as for falling edge data. The remaining slight 0.51% variation of results is caused by possible output delay decrease for *NOR* gates in 1→0 transition mentioned in Section III. Moreover, the *pMOS* transistor location in a multiple input gate, i.e. distance to the output node, also impacts on the propagation delay. These two aspects were not considered in our approach, but will be investigated in the future.

TABLE II. COMPARISON BETWEEN THE PROPOSED GATE-LEVEL ESTIMATION AND THE REFERRED SPICE SIMULATION OF PATH DELAY DEGRADATION

| Path | 0→1 (rise-edge delay) | | 1→0 (fall-edge delay) | |
|---|---|---|---|---|
| | Gate-level | SPICE | Gate-level | SPICE |
| | Δ, % | Δ, % | Δ, % | Δ, % |
| F3#26 | **10.65** | **14.88** | **13.55** | **13.03** |
| F3#38 | **10.55** | **14.38** | **13.74** | **13.54** |
| F2#61 | **9.71** | **13.85** | **14.40** | **14.26** |
| F3#74 | **11.25** | **14.92** | **12.67** | **12.69** |
| F1#77 | **5.96** | **9.84** | **17.81** | **16.25** |
| Average | **9.62** | **13.57** | **14.43** | **13.95** |

## V. CONCLUSIONS

We developed a predictive model to characterize NBTI-induced path delay degradation at gate-level for circuits designed in nanotechnology. This model was based on intensive *HSPICE* electrical simulations of basic gates such as Inverter, *NAND* and *NOR* up to 5 inputs. The proposed method first calculates the delay degradation of each gate using the worst-case duty cycle. This is done by capturing the dependence of gate output delay on *pMOS* transistor threshold voltage $V_{THp}$ shift. Then, based on static timing analysis, this approach computes the aging information for all delay-critical paths of the entire circuit.

The proposed approach has been demonstrated on an industrial *ALU* circuit. The obtained results demonstrate an excellent fitting between the developed model and *HSPICE* simulations at both, gate and path levels. Additionally, the proposed approach requires a very short runtime when compared to *SPICE* simulations at the electrical-level.

## ACKNOWLEDGMENTS

The work has been supported in part by CNPq (Science and Technology Foundation, Brazil) under contract n. 303701/2011-0 (PQ), by FAPERGS/CAPES 014/2012, by EU's FP7 STREP project BASTION, by Estonian ICT program project FUSETEST, by Research Centre CEBE funded by European Union through the European Structural Funds and by Estonian SF grant 9429.

## REFERENCES

[1] S.Hamdioui, D.Gizopoulos, G.Guido, M.Nicolaidis, A.Grasset, P.Bonnot, "Reliability Challenges of Real-Time Systems in Forthcoming Technology Nodes", ACM/IEEE DATE, March 2013, pp. 129-134.

[2] S. Mahapatra, D. Saha, D. Varghese, P. B. Kumar, On the Generation and Recovery of Interface Traps in MOSFETs Subjected to NBTI, FN, and HCI Stress, IEEE Trans. Electron. Dev. 53, 7, 1583-1592, 2006.

[3] Ing-Chao Lin, Chin-Hong Lin, Kuan-Hui Li, Leakage and Aging Optimization Using Transmission Gate-Based Technique, IEEE Trans. on Computer-Aided Design of Integratedd Circuits and Systems, Vol. 32, No. 1, Jan. 2013, pp. 87-99.

[4] C. Ferri, D. Papagiannopoulou, R. Iris Bahar, A. Calimera, NBTI-Aware Data Allocation Strategies for Scratchpad Memory Based Embedded Systems, IEEE 12th Latin American Test Workshop (LATW'11), March 27-30, 2011, Porto de Galinhas, Brazil.

[5] Ceratti, A.; Copetti, T.; Bolzani, L.; Vargas, F.; , "Investigating the use of an on-chip sensor to monitor NBTI effect in SRAM," Test Workshop (LATW), 2012 13th Latin American , vol., no., pp.1-6, 10-13 April 2012.

[6] M. Khan, P. Weckx, P. Raghavan, S. Hamdioui, et al., Comparison of Reaction-Diffusion and Atomistic Trap-based BTI Models for Logic Gates, Device and Materials Reliability, IEEE Transactions, June 2013.

[7] W. Wang, S. Yang, S. Bhardwaj, S. Vrudhula, F. Liu, Y. Cao, The Impact of NBTI Effect on Combinational Circuit: Modeling, Simulation, and Analysis, IEEE Trans. On VLSI, vol. 18, no. 2, 2010, pp. 173-183.

[8] F. Ahmed, L. Milor, Reliable Cache Design with On-Chip Monitoring of NBTI Degradation in SRAM Cells using BIST, 2010 28th IEEE VLSI Test Symposium, pp. 63-68.

[9] S. Khan, S. Hamdioui, Modeling and Mitigating NBTI in Nanoscale Circuits, 2011 17th International On-Line Testing Symposium, pp. 1-6.

[10] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "NBTI-aware synthesis of digital circuits," in *Proc. Design Autom. Conf.*, 2007, pp. 370–375.

[11] Predictive Technology Model for Nano-CMOS Design Exploration. ACM Journal on Emerging Technologies in Computing Systems, Vol. 3 Issue 1, April 2007, Article No. 1 (doi:10.1145/1229175.1229176), http://ptm.asu.edu/, http://ptm.asu.edu/modelcard/2006/65nm_bulk.pm

[12] I. Sutherland, R. Sproull, D. Harris "Logical Effort:Designing Fast CMOS Circuits", Morgan Kaufmann, 1999.

[13] Data sheet "74HC/HCT181 4-bit arithmetic logic unit", Philips, 1998.

[14] I. Sutherland, R. Sproull, D. Harris "Logical Effort:Designing Fast CMOS Circuits", Morgan Kaufmann, 1999.