Characterizing the activity factor in NBTI aging models for embedded cores

Characterizing the Activity Factor in NBTI Aging Models for Embedded Cores

Yukai Chen Andrea Calimera Enrico Macii Massimo Poncino
Department of Control and Computer Engineering
Politecnico di Torino
10129, Torino, Italy

ABSTRACT

In deeply scaled CMOS technologies, device aging causes cores performance parameters to degrade over time. While accurate models to efficiently assess these degradation exist for devices and circuits, no reliable model for processor cores has gained strong acceptance in the literature. In this work, we propose a methodology for deriving an NBTI aging model for embedded cores. Our model separates the dependency on logic values of NBTI (the *functional* term) from the relation of NBTI with the technological and environmental parameters of aging (the *physical* term). The physical term is a traditional NBTI analytical model only depending on elapsed time, in which the functional terms plays the role of the "activity-related" factor of the model.
Based on an accurate characterization on the netlist of the core, we were able to (1) prove the independence of the aging on the workload (i.e., executed instructions), and (2) calculate an equivalent average constant aging factor that justifies the use of the baseline model template.
We derived and assessed the proposed model by using a RISC-like processor core implemented in a 45nm process technology as a reference architecture, achieving a maximum error of 2.2% against simulated data on the core netlist.

Categories and Subject Descriptors

J.6 [**Computer-Aided Engineering**]: Computer-aided design (CAD); I.6.4 [**Simulation and Modeling**]: Model Validation and Analysis

General Terms

Design, Experimentation, Performance.

Keywords

Aging, Processors, Reliability, Modeling.

1. INTRODUCTION

In the recent years, a large bulk of research has addressed the issue of NBTI-induced aging. Most of these works have however been focused on logic blocks and SRAM structures; this is because the accurate characterization of NBTI aging requires the availability of the circuit netlist in order to extract the critical paths and the signal probabilities of the relative cells. These information are available for logic circuits during synthesis, and are implicit for SRAM structures whose topology is well-defined. Conversely, this is not the case with processor cores; cores are regarded as black-box, third-party IPs whose netlists are obviously not available. The state of the art in modeling of the aging of a core relies on simple approximations based on core "usage": an active core will age according to some constant *aging factor*, and it will simply not age when idle ([4]–[8]). The baseline models for the various aging mechanisms can be either analytical (taken from physics, as in [5, 6, 7, 8] or empirical (derived by fitting data as in [4]). The main drawback of these approaches is that the assumption of a constant "aging factor" is neither motivated nor validated; they provide no evidence on (i) how this factor can be computed, (ii) how general (i.e., applicable to other processors) it is. Moreover, the underlying models used in these works refer to the model for a single logic gate; however, extension at the core level of a gate-level model should be proved [8].
In this work, we propose to infer some general properties of an NBTI aging model from the analysis of a core with an available netlist. Our claim is that since most processors share a common architectural concept, the main conclusions will have a reasonably general validity. We target relatively simple embedded processors, for which we can have a better degree of confidence about the generality of the presented results. Our reference "open-netlist" core is a MIPS-based RISC architecture called Plasma [14].
The core netlist is used to obtain detailed aging data using a set of application kernels for characterization; these data are then used to fit the underlying model template. Statistical correctness of the evaluation is guaranteed by using different datasets for the characterization and for the validation. Two are the main conclusions drawn from the validation of the proposed model:

1. **Aging model is roughly independent of the workload**. Quite surprisingly, the impact of different applications and different data sets have negligible impact on the aging of the core.

2. **The aging factor roughly corresponds to the application of a 0.58 static probability at the critical path inputs**. Since the netlist is available, the aging factor can be quantified precisely. This is the fundamental result that motivates the use of a classical NBTI aging gate-level model for application to to an entire logic block.

The proposed model show extremely good accuracy: comparison against simulated values on the netlist yields a maximum error of 2.2%.

2. BACKGROUND AND PREVIOUS WORK

2.1 Background

NBTI occurs when a pMOS transistor is negatively biased (i.e., a logic '0' applied to the gate of the pMOS, called the *stress* state), and manifests itself as an increase of the threshold voltage V_t over time, resulting in turn in a degradation of the delay of a device. Conversely, when a logic '1' is applied (called *recovery* state), NBTI stress is partially removed, resulting in a decrease of the threshold voltage. A widely used model of NBTI-induced V_t drift is derived

from the reaction-diffusion model, and can be summarized as follows [1]:

$$\Delta V_t = \mathcal{A} \cdot f(V_{dd}, V_t, T, \mathbf{R}) \cdot t^n. \quad (1)$$

The model has three main factors:

1. $\mathcal{A}$ denotes the *aging factor* which reflects the actual stress/recovery pattern;
2. A term including all technological and environmental parameters ($f()$): supply voltage V_{dd}, threshold voltage V_t, temperature T, and all device parameters, lumped here for compactness into set $\mathbf{R}$, comprising oxide geometrical and electrical parameters, activation energy, device size, load, etc. See [1] for the precise mathematical expression of $f()$.
3. The dependence over time. n is 1/4 or 1/6 depending on the diffusing species. In this work we use $n = 1/6$.

For a given condition of $\mathbf{R}$, it is the value of $\mathcal{A}$ determines the actual V_t drift. Thanks to some mathematical properties of NBTI aging ([9]), it can be shown that it is possible to use *signal probabilities* instead of actual signal values to determine the effective stress. It is worth remarking that the model of Equation 1 technically applies to an individual transistor and, with minor adaptations, to a logic gate. The translation of the V_t drift on a more macroscopic performance metric such as circuit delay or processor maximum frequency still requires the availability of a netlist to determine the actual critical paths.

2.2 Previous Work

In this section we will review the literature on aging models for processors; for a more in-depth overview of NBTI modeling solutions for generic logic circuits and memories we refer the reader to [10, 2].

The work of [5] presents an NBTI-aware processor (Penelope) in which several aging mitigation strategies are proposed. For evaluation of these strategies, the authors do not use a true aging "model" but rather use an *NBTI efficiency* metric that combines the nominal delay, the TDP (thermal design power) limit, and the NBTI guard-band of the processor. The latter is obtained by choosing the maximum guard-band required by any block, assuming that all paths of the different blocks have been adjusted to fit the cycle time to save power. In some sense this work relies a "static" NBTI model, where aging is not truly evaluated but statically defined in terms of a guard-band for each sub-block.

The authors of [6] adopt a sort of "a priori" model. They assume target processor lifetime of 7 years, and evaluate two different aging rates which increase the delay of the critical paths by 10% and by 25%, respectively, in 7 years. They use an explorative approach for other parameters related to NBTI aging, namely the average fraction of stress time of PMOS transistors, and the the average ratio of PMOS to NMOS transistors in critical paths.

The work of [7] aims at balancing workload in multicores using an aging-related metric. The aging model for a core relies on a traditional gate-level model for a critical path; the authors provide no insight on (i) how the critical path(s) of the processor is detected, and (ii) how the percentage of stress on these critical paths is computed.

Another aging model was proposed in [8]; they adopt the model of Equation 1 as a reference and extend it at the core level; $\mathcal{A}$ is assumed to be available either by direct characterization on the core or by using specific aging monitors (e.g., [11]). For the first option, they also suggest an approach similar to ours, based on the collection of statistics about delay and core activity at different operating conditions (i.e., V_{DD}, T) running benchmarks with different activity levels A relationship between delay and core activity can be finally established, for example using regression analysis. However, the paper does not specify further details about its implementation nor results.

3. ACTIVITY-SENSITIVE AGING MODEL

The proposed model uses Equation 1 as the baseline template, yet. with two substantial differences:

1. **The aging factor $\mathcal{A}$ is empirically characterized** by a circuit-level analysis of the netlist of the processor;
2. Thanks to empirical evidence resulting from the characterization, it is possible to **use an "equivalent aging factor" for the entire core** (actually, the block containing the critical path). This justifies the assumption of a constant aging factor $\mathcal{A}$ to express the aging of the core, as done also in other works but without a substantiation of this claim.

Although our model is derived using a specific target processor core platform (the Plasma MIPS platform) as a reference, our claim is that since most small embedded cores share a relatively similar architecture, (e.g., RISC-based, relatively shallow pipelines, etc.) the main conclusions drawn here will have a reasonably general validity.

Since Equation 1 expresses a drift in threshold voltage and technically refers to a single gate, our first task in order to model the aging of a processor is to devise a macro-model that (i) tracks a quantity related to the system-level performance of a core, and (ii) uses a "core-level" activity factor (as opposed as a gate-level one).

The first objective can be met by modeling, rather than the drift in threshold voltage, *the maximum operating frequency of the core*. This is done by translating the threshold voltage drift of Equation 1 into a delay degradation using a classical alpha-power law [12], whose inverse determines the maximum operating frequency. Notice that this is possible since our methodology relies on the availability of a core netlist, thus it is possible to accurately extract the critical path by simulation and compute the maximum frequency.

The second requirement implies that we should use an aging factor $\mathcal{A}$ that is some function of the *workload* $\mathbf{W}$. From the core perspective the workload is a mix of applications, whereas at the gate-level this will translate into some signal probability pattern in the circuit. This "core-level" activity factor will therefore represents a sort of mapping of the workload onto signal probability values.

Equation 2 shows the expression of the proposed model.

$$f_{max,aged} = \mathcal{A}(\mathbf{W}) \cdot \mathcal{K}(t, V_{dd}, V_t, T, \mathbf{R}) \cdot f_{max,nom} \quad (2)$$

where $f_{max,aged}$ is the aged frequency at time t and $f_{max,nom}$ is the nominal frequency of a fresh core. Notice that both $\mathcal{A}$ and $\mathcal{K}$ are scale factors between 0 and 1.

The derivation of the model consists then of two phases. First, the value of the aging factor $\mathcal{A}$ is calculated through the characterization on the core netlist of appropriate workloads. Second, based on the corresponding aging resulting from the applied workloads, the factor $\mathcal{K}$ is empirically determined by fitting the results of the simulation. The details of the methodology are described in the following section.

4. MODEL DERIVATION AND VALIDATION

4.1 Reference Core Platform

The Plasma3 CPU [14] is a synthesizable 32-bit RISC microprocessor implemented in VHDL that executes all MIPS-I user-mode instructions. The main memory contains both instruction and data. The design features an interrupt controller, UART, SRAM or DDR SDRAM controller, and Ethernet controller. Our version is implemented with a three stage pipeline and an additional stage for memory read and writes. The critical path occurs within the execution stage of the core, it includes 110 gates and results into a critical delay of 4.834 ns and a $f_{max,nom}$ of 208.63 MHz.
We chose Plasma as a target platform because it has a relatively general architecture as our platform in order to have a good degree of confidence about the generality of our methodology.

4.2 Simulation Setup and Toolchain

Figure 1 shows the detailed flow of our methodology.

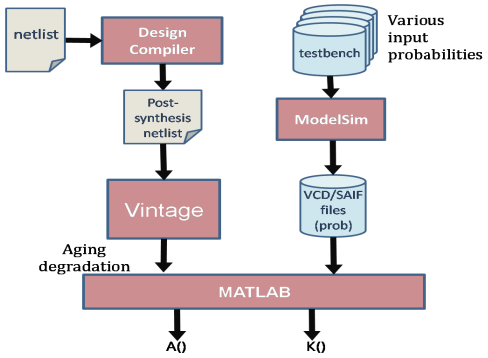

Figure 1: Methodology Implementation.

It uses Synopsys Design Compiler to synthesize the Plasma on the target 45nm cell library, and Mentor Modelsim for the simulation of the Plasma netlist both for characterization (with different signal probabilities on the inputs for $\mathcal{A}$ in $\mathcal{K}$) and for validation. We use Vintage [3], an academic tool that uses aging-characterized libraries to calculate gate-by-gate aging on a critical path, to carry out an NBTI-aware static timing analysis. Finally, Matlab is used for the statistical fitting of $\mathcal{K}$ and $\mathcal{A}$.

4.3 Model Characterization

4.3.1 *Characterization of $\mathcal{A}$.*

The first step of the methodology is the characterization of $\mathcal{A}$ as a function of the workload. To this purpose, we designed a set of small application kernels that induce different signal probabilities in the core. We chose relatively small kernels because of the limitation of memory size and simulation time. Table 1 lists the set of applications, where **LOC** denotes Lines of Code. In order to further exercise different probability values, strongly data-sensitive applications (sorting ones and CountNumber) have been fed with two input datasets, with different percentages of 0's and 1's. We ran these applications on our platform and extracted the aging with Vintage on the netlist. Figure 2 shows the frequency degradation over time.
The largest difference among applications is 7.14 MHz (about 3.5%); For data-dependent applications, the values represent the average of the two datasets used as inputs. The plot seems to suggest that aging degradation is roughly independent of the workload.

Application	LOC	Application	LOC
BubbleSort	396	CountNumber	697
HeapSort	402	CalculatePi	321
QuickSort	456	FFT	781
InsertSort	487	MatrixMult	644
Helloworld	250	ImageConvert	755

Table 1: Application list we tested in our experiment

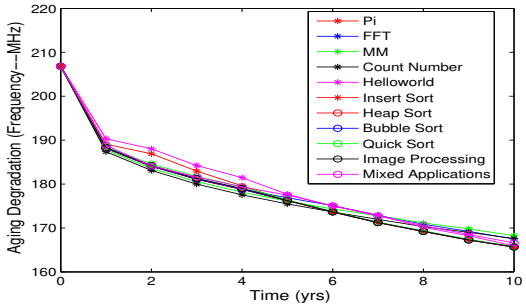

Figure 2: Aging Degradation for Test Applications.

To further support this claim, we manually forced all possible static probability values (from all 0's to all 1's in steps of 0.1) at the inputs of the pipeline stage that determines the critical path and have these values propagated into the netlist with Vintage. This experiment guarantees that the workload independence is not a consequence of possible poor choice of application mix, which might possibly exercise only a limited range of signal probabilities. This strategy is similar to approaches typically used to generate power macromodels for RTL power estimation [13].
Even more surprisingly, the simulation of these data confirms the results of Figure 2, as shown in Table 2. The table

	Static probability				
Year	0.0	0.2	0.4	0.6	0.8
0	206.83	206.83	206.83	206.83	206.83
1	191.67	191.64	191.67	191.67	191.64
2	188.52	188.49	188.52	188.52	188.49
3	185.81	185.77	185.81	185.81	185.77
4	182.14	182.09	182.14	182.14	182.09
5	179.05	179.01	179.05	179.05	179.01
6	176.42	176.37	176.42	176.42	176.37
7	173.94	173.89	173.94	173.94	173.89
8	171.82	171.76	171.82	171.82	171.76
9	169.93	169.87	169.93	169.93	169.87
10	167.99	167.92	167.99	167.99	167.92

Table 2: Max Core Frequency [MHz] for Different Input Signal Probabilities.

shows maximum frequency for various input static probability values (columns, in steps of 0.2) vs. time (rows); it is evident from the data that frequency degradation is virtually independent of **input** static probability. The largest difference among all points is only 0.034%.
Such independence of input probabilities allows us to prove the main result of the characterization phase: *since aging is not affected by the input probability values, it is possible to remove the dependency of the workload W from $\mathcal{A}$ in (2).*
However, this property alone does not allow to use a model meant for gate-level aging as Equation 1 at the core level. To this purpose, we need to identify a single, fixed aging factor that can be applied to all the critical gates and that can be characterized once and for all. In order to do this, we artificially *set different static probability values to* **all signals in the core netlist**. Notice that these values are not logically feasible and can only be forced by writing a corresponding

VCD file. The aging factor will be then obtained by finding the best match between the resulting aged frequency values and those of Table 2.
Table 3 reports the frequency values at different times and for different artificially forced probability values on all the internal signals of the cores. As in Table 2, for compactness only values of 0.0, 0.2, 0.4, 0.6, and 0.8 are reported. Aging effects are now sizable, the difference 10-year aging between the worst-case (signals all at 0 probability) and the case with all signals at a 0.8 probability is about 14%.

	Signal probability				
Year	**0.0**	**0.2**	**0.4**	**0.6**	**0.8**
0	206.83	206.83	206.83	206.83	206.83
1	184.82	186.02	187.36	189.05	191.24
2	179.55	181.51	183.19	185.27	188.08
3	173.87	176.72	180.07	182.71	185.78
4	169.52	172.55	176.00	180.54	183.98
5	165.87	169.07	172.87	177.55	182.56
6	162.90	166.25	170.09	174.99	181.27
7	160.05	163.64	167.69	172.67	179.70
8	157.62	161.27	165.50	170.70	177.74
9	155.42	159.06	163.40	168.81	176.00
10	153.27	157.10	161.59	167.16	174.60

Table 3: Max Core Frequency [MHz] for Different Static Probability Values for All Signals.

Searching the best match between columns of Table 3 and Table 2 (whose columns are approximately the same) appears to be around a probability value of 0.6. A finer-grain analysis around that point yielded a best match for a probability value of **0.58**. This is equivalent to say that *if we assume a 0.58 static probability value for* **all** *the nodes in the netlist, the core will age approximately as it will under application of typical inputs.* This allows us to use a single, fixed constant aging factor $\mathcal{A}$ to derive our physical term and to also use the gate-level aging model for the whole core without inconsistencies.
Notice that 0.58 is not very far from an intuitive choice of 0.5, in which signal are assumed to change randomly; this suggests that the above result is likely to have a reasonable generality over different platforms. A precise characterization of this value on a given platform would anyway require an analysis similar to the one described here.
As a side result, the table provides (loose) upper and lower bounds on the aging. Clearly, all 0's will age all p-transistors (worst case) and all 1's will allow them to recover (best case). Obviously, aging monotonically decreases for increasing static probability.

4.3.2 Calculation of $\mathcal{K}()$

Now that we have defined an equivalent aging factor, derivation of the physical term $\mathcal{K}()$ becomes straightforward. We take the aging of Table 3 corresponding to $\mathcal{A} = 0.58$ and use Matlab to fit the values to a polynomial curve, after testing different orders of polynomial, we found a fifth-order one has the least error, yielding:

$$\mathcal{K}(t) = -0.014t^5 + 0.015t^4 + 0.025t^3 - 0.048t^2 - 0.048t + 0.86$$

4.4 Model Validation

For the assessment of the model accuracy, we ran three applications not used for the characterization phase: and RGB image conversion (TB1), a matrix manipulation program (TB2) and a mix of the previous two (TB3). Table 4 shows frequency degradation for 1 to 10 years returned by the model and obtained from simulation The comparison shows very good match: the maximum error is 2.2%, while the average over all time points and the three application is only 0.7%.

	Each Aging Year Error[%] (Simulation *VS* Model)									
TB	1	2	3	4	5	6	7	8	9	10
1	1.2	0.2	0.3	0.1	0.0	0.4	0.6	0.4	0.5	0.5
2	1.6	0.7	0.2	0.0	0.2	0.1	0.1	0.1	0.7	0.1
3	0.7	2.0	2.2	1.7	1.7	1.9	2.0	1.5	0.7	0.3

Table 4: Accuracy of our NBTI-aging Model

5. CONCLUSIONS

In this paper we have presented an NBTI aging model for processor cores. The proposed methodology results into two major conclusions: (1) the aging degradation is independent of the workload, (2) it possible to identify an "effective" static probability that can be used as a core-wide stress probability, which allows one to use a gate-level aging model for the entire core. This value has been calculate as a 0.58 stress probability based on a large volume of statistical data. With verification of 45nm silicon data and a RISC processor core architecture(Plasma), our methodology supports statistical aging analysis in standard design flow, improving design predictability and helping avoid pessimistic guard-banding under the increasingly severe NBTI aging effect.

6. REFERENCES

[1] Alam, M. "Reliability-and process-variation aware design of integrated circuits."*Microelectronics Reliability* 48.8 (2008): 1114-1122.

[2] Paul, Bipul C., et al., "Temporal Performance Degradation under NBTI: Estimation and Design for Improved Reliability of Nanoscale Circuits," *DATE-06: Design Automation and Test in Europe*, pp. 1-6, March 2006.

[3] A. Calimera, E. Macii, M. Poncino, "NBTI-Aware Power Gating for Concurrent Leakage and Aging Optimization", *ISLPED'09: ACM International Symposium on Low-Power Design and Electronics*, 2009.

[4] J. Srinivasan, et.al., "Lifetime reliability: towards an architectural solution" IEEE Micro, pp. 70-80, 2005.

[5] Jaume Abella, Xavier Vera, Antonio Gonzalez, "Penelope: The NBTI-Aware Processor". *40th IEEE/ACM International Symposium on Microarchitecture*, pp. 85-96, 2007.

[6] A. Tiwari, J. Torrellas, "Facelift: Hiding and Slowing Down Aging in Multicores", *41th IEEE/ACM International Symposium on Microarchitecture*, 2008.

[7] J. Sun, et.al., "NBTI Aware Workload Balancing in Multi-core Systems," *ISQED'09: 10th International Symposium on Quality Electronic Design*, March 2009. pp. 833–838.

[8] F. Paterna, A. Acquaviva, L. Benini, "Aging-Aware Energy-Efficient Workload Allocation for Mobile Multimedia Platforms," *IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 8, August 2013, pp. 1489–1500.*

[9] Kumar, Sanjay V., Chris H. Kim, and Sachin S. Sapatnekar, "An analytical model for negative bias temperature instability," *ICCAD'06: 2006 IEEE/ACM international conference on Computer-aided design*,pp. 493–496, Nov. 2006.

[10] R. Vattikonda, W. Wang, Y. Cao, "Modeling and minimization of PMOS NBTI effect for robust nanometer design," *DAC-44: ACM Desing Automation Conference*, pp. 1047-1052, 2006.

[11] J. Keane, T.-H. Kim, and C. Kim, "An On-Chip NBTI Sensor for Measuring PMOS Threshold Voltage Degradation," *IEEE Trans. Very Large Scale Integration Systems*, vol. 18, no. 6, pp. 947-956, June 2010.

[12] T. Sakurai, A.R. Newton, "Alpha-Power Law MOSFET Model and its Applications to CMOS Inverter Delay and Other Formulas," *IEEE Journal of Solid-State Circuits*, Vol. 25, No. 2, Apr. 1990, pp. 584–594.

[13] A. Bogliolo, R. Corgnati, E. Macii, M. Poncino, "Parameterized RTL power models for Soft Macros," *IEEE Transactions on VLSI*, Vol. 9, No. 6, pp. 880-887, June 2001.

[14] Plasma Project, `http://opencores.org/project,plasma`.