Incorporating the Impacts of Workload-Dependent Runtime Variations into Timing Analysis

Farshad Firouzi[1] Saman Kiamehr[1] Mehdi Tahoori[1] Sani Nassif[2]

[1]Department of Computer Science and Engineering
Karlsruhe Institute of Technology
Karlsruhe, Germany
e-mails: {firouzi, kiamehr, mehdi.tahoori}@kit.edu

[2]Austin Research Lab
IBM, Austin
TX 78758
e-mail: nassif@us.ibm.com

***Abstract*—In the nanometer era, runtime variations due to workload dependent voltage and temperature variations as well as transistor aging introduce remarkable uncertainty and unpredictability to nanoscale VLSI designs. Consideration of short-term and long-term workload-dependent runtime variations at design time and the interdependence of various parameters remain as major challenges. Here, we propose a static timing analysis framework to accurately capture the combined effects of various workload-dependent runtime variations happening at different time scales, making the link between system-level runtime effects and circuit-level design. The proposed framework is fully integrated with existing commercial EDA toolset, making it scalable for very large designs. We observe that for benchmark circuits, treating each aspect independently and ignoring their intrinsic interactions is optimistic and results in considerable underestimation of timing margin.**

I. Introduction

VLSI chips manufactured at nano-scale technology nodes face various reliability challenges [1], [2], [3]. Runtime variation including transistor aging together with workload-dependent voltage and temperature variations is considered as one of the major sources of unpredictability in VLSI designs resulting in considerable timing mismatch between design time and runtime [4], [5]. Moreover, each source of short-term and long-term timing uncertainty is considered as an independent factor in existing techniques [2]. However, all these phenomena, although affecting the circuit at different time scales, are tightly coupled together (see Figure 1), thereby, separate analysis of each source of timing variations leads to significant inaccuracy in the estimated timing margins [2], [6].

Transistor aging mostly due to Bias Temperature Instability (BTI) has become one of the dominant limiting factors of circuit lifetime. BTI gradually shifts the threshold voltage of the transistors, thereby increases the circuit delay over the time (*long-term effects*) [7]. Runtime variations of the temperature and the supply voltage due to workload signature have emerged as another dominant factor in circuit delay uncertainty (*short-term effects*) [8]. Due to increased circuit complexity, spatial/temporal variations of voltage and temperature are considerably large [8], [9]. Runtime variations are interdependent, although occurring at different time scale, which makes their timing impacts extremely complex to analyze.

In this paper, we provide a link between workload dependent runtime variations and timing analysis during design. Specifically: **1)** We propose a new methodology to incorporate workload dependent runtime variations, occurring at different time scales, into circuit-level delay estimation. This is achieved by performing system-level profiling and accurately projecting the effects at the circuit-level. **2)** We consider the combined effects of voltage and temperature together with BTI effect during the timing analysis. This method significantly reduces the inaccuracy of the safety margin when each source of timing variation is considered independently. **3)** We propose a new *Look-Up Table (LUT) based* temperature- and voltage-aware timing analysis method considering BTI effect, providing very high accuracy and low runtime. **4)** Our proposed framework is built on top of commercial EDA tool chains and therefore it scales very well.

The rest of the paper is organized as follows. The proposed runtime variations aware timing analysis method is described in Section II. Experimental results are demonstrated in Section III. Finally, Section IV concludes the paper.

II. Proposed Flow

In this section, we present our framework to capture the combined effects of various runtime variations, through system-level workload profiling, and project them into a circuit-level static timing analysis model. To account for the combined effects of the runtime variations on timing analysis, we propose an iterative approach to extract the temperature and voltage profiles as well as the BTI-induced delay degradation considering their interdependences. Since equation-based gate delay models cannot completely capture the combined and complex effects of these parameters, we propose an LUT-based technique to model the gate delay. This approach has two benefits. First, the accuracy is comparable to transistor-

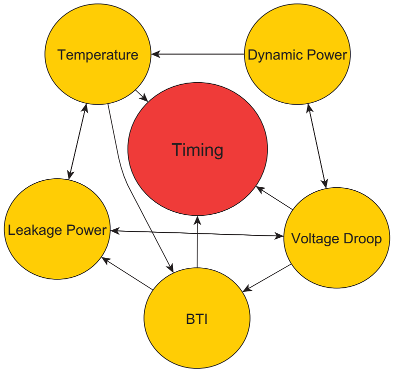

Fig. 1. Interdependence of different sources of variation and their impact on BTI that considered in existing method and our proposed technique.

978-3-9815370-0-0/DATE13/ ©2013 EDAA IEEE

level SPICE timing simulation while providing significant runtime speedup. Second, LUT-based model is compatible with traditional timing analysis flow. Figure 2 shows the overall flow, which consists of six different steps:

A. *Workload profiling*

For microprocessor based design, a performance simulator (e.g. M5 [10]) is exploited to extract workload signature including clock gating time, power gating time, and input patterns for different functional blocks of the processor for a target application. For other designs (e.g. ASIC) high-level (system level) simulation vectors can be used for profiling. This system-level workload-dependent profiling data which contains usage and power information (considering power management scenarios) is then given to the *Step B* to be further processed for timing analysis. By this approach, we establish a link between system-level workload dependencies ,happen at the runtime, to timing analysis at the design time.

B. *Circuit-level simulation of system-level profiles*

This step acts as a link layer between system-level workload dependent profiles and the traditional circuit-level static timing analysis tools. In this step, the obtained system-level information from *Workload profiling* step is processed at circuit-level to extract the workload-dependent usage (logic level usage) details of each individual gate (and transistor) inside the circuit. For this purpose, the circuit is synthesized and mapped to the technology library. Next, generated gate-level netlist and profiled workload signature are given to a logic simulator (e.g. Modelsim) in order to obtain signal probability and switching activity of each transistor inside the netlist. By using a method presented in [11], output of the logic simulator is further processed to obtain the effective duty cycle of each transistor in every gate by considering the stacking effect for accurate BTI-induced V_{th} shift and delay change analysis. This information is also used afterwards for voltage-temperature profiling and BTI estimation.

C. *Power-Voltage-Temperature profiling and BTI estimation*

In this step, the extracted workload-dependent usage of each device (output of the *Step 2*) is translated to voltage, temperature and BTI-induced threshold voltage change. Considering the correlation and interdependencies among different sources of runtime variations while accounting their different time scales is a major challenge. Voltage droop has a short term variation (ns) which is a result of different input vectors which are applied to the circuit. Temperature varies at higher time scale (ms) and as a result for thermal profiling, only considering the DC-behavior of the voltage droop would be sufficient [12]. On the other hand, BTI is a phenomenon which increases the circuit delay gradually (several weeks and months). Therefore, to estimate the BTI-induced delay degradation over time, it is sufficient to use the average value of the temperature and the supply voltage at the time scale which BTI is considered.

As depicted in Figure 2(b), the proposed flow consists of two nested loops. These two loops are used to accurately model the interdependences among the voltage droop, temperature, and BTI. In the inner loop, power consumption of each grid is calculated by adding up the power consumption (composed of dynamic and leakage power dissipation) of each cell located inside the given grid. Once the power profile is obtained, it is converted into the temperature profile. Temperature profile can be extracted by using a sign-off thermal-profiling tool (e.g. HotSpot). Afterwards, a resistive 2D mesh is made to model the power (voltage) grid. This network is connected to power supply (i.e. V_{dd}) every 500 microns horizontally and vertically [13]. Therefore, the voltage droop as a function of drawn current is written as follows [12]:

$$V = G^{-1}I \qquad (1)$$

where V is the vector of supply voltages of the grids. I is the vector of current drawn off the power grids and G is the conductance matrix. The current drawn from each grid can be calculated by adding the dynamic and leakage current of all the gates inside the grid (calculated by power profiling). Next, obtained temperature and voltage droop profiles are used to update the gate power and in turn power profile. This loop is iterated until convergence is reached.

After reaching a convergence, voltage and temperature information of each gate is used to estimate the BTI-induced threshold-voltage change using a model presented in [14]. The new threshold voltage is then used to update the power, temperature, and voltage profiles. In other words, the inner loop and BTI-estimation are parts of the outer loop. These two loops are iteratively executed until all the profiles reach a convergence. According to our observations, each loop, at worst case, only needs 10 iterations to converge.

D. *Cell characterization*

In this step, which has to be performed only once, each cell in the technology library file is characterized by using accurate SPICE simulations. The extracted delay information including 1) *gate delay* and 2) *output transition time* is stored in $n+4$-dimensional LUTs. The generated LUTs precisely define the delay and output slope of each cell in the library as a function of: 1) V_{th} shifts of different transistors (n transistors corresponding to n dimensions in LUT) inside the cell, 2) input slope, 3) output load, 4) temperature, and 5) voltage. By this method, the combined effect of the NBTI, PBTI, as well as the stacking effect could be captured. These LUTs are later used in *Interpolation* step.

Please note that although the characterization process for simple primitive gates is simple, it is more complicated for complex cells such as flip-flops and half-adder. However, since this process is performed once, the proposed flow does not suffer from runtime issue. Another important issue during LUT generation is accuracy, i.e. sampling frequency (table index) of each dimension. We observed that $10\,^{\circ}\mathrm{C}$, $0.05v$, $0.02v$ as the sampling intervals are reasonable choices for temperature, voltage, and threshold voltage, respectively, for a good trade-off between runtime and accuracy. For the other dimensions (i.e. input slope and output load) we use the default sampling rate as defined in the original technology library file.

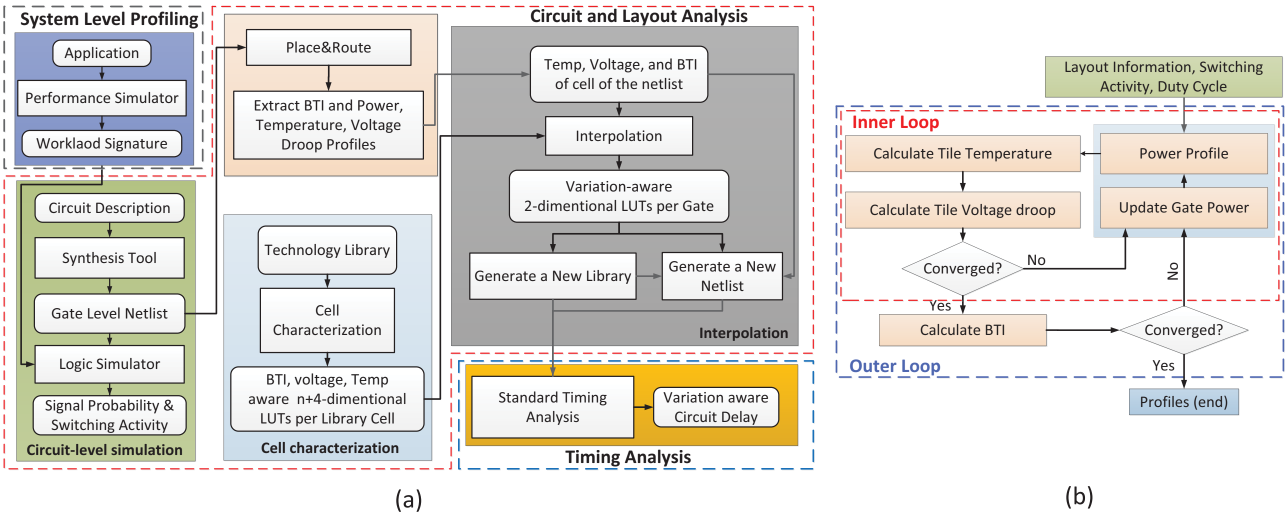

Fig. 2. (a) Overall flow of the proposed runtime variations aware timing analysis (b) Overall flow of Power-Temperature-Voltage profiler and BTI analyzer

E. Interpolation

In this step, the workload dependent information of runtime variations (i.e. extracted temperature, voltage and BTI of each gate obtained in *Step 3*) are organized in a way that can be used by traditional circuit-level static timing analysis tools. In conventional static timing analysis tools (e.g. Synopsys PrimeTime), gate delay and gate output transition time are modeled as a function of only input transition time and output load capacitance (2-dimensional LUTs). Therefore, we need to reduce the dimensions of the $n + 4$-dimensional LUTs generated in *Cell characterization* step (i.e. those dimensions which reflect aging, voltage, and temperature). In the interpolation phase, temperature, voltage, and BTI information of the cells (generated in profiling step) and $n+4$-dimensional LUTs (generated in cell characterization step) are given to a linear interpolation method. Interpolation is responsible to construct a new data point within the range of already known data points. By performing interpolation, for each cell in the netlist, a newly generated library element (2-dimensional LUTs) is obtained which captures the BTI and voltage-temperature information of the cell. In addition, the original gate-level netlist of the circuit is modified accordingly in a way that each cell is mapped to the corresponding generated library element. As an example, it is possible that two different instances in the original netlist from a same primitive gate (e.g. $NAND2\_X1$ in the original library) are mapped into two different elements in the newly generated library file to capture different runtime variations (e.g. voltage, temperature).

F. Timing analysis

Finally, the modified gate-level netlist and the generated runtime variations aware technology library (one element per each cell in the netlist) which are extracted in *Step 5* are given to a static timing analysis tool to determine the circuit delay. Since LUTs are able to capture the effect of different parameters (such as temperature, voltage information and ΔV_{th} of different transistors within a cell) on gate delay, the estimated gate delay by this approach is very close to transistor level SPICE information. Another advantage of our method is that, it can be extended to handle other aspects of gate delay by augmentation of LUTs with other parameters such as process variation. Moreover, our LUT-based approach has the capability of a space/accuracy tradeoff and can be calibrated with post-silicon data as well. Please note that, since BTI is time-varying parameter and changes from time to time, the extracted circuit delay is valid for a specific time point. However, our methodology is general and can easily be used to predict the circuit delay for each requested time point.

III. Experimental Results

Several IWLS and ISPD benchmark circuits [15], [16] are used to evaluate the efficiency and accuracy of the proposed methodology. Circuits are synthesized by Synopsys Design Compiler using Nangate 45 nm library [17] and then the gate-level netlists are placed using Cadence SOC Encounter. Besides, each cell in the library is characterized by accurate HSPICE simulations. Moreover, HotSpot [18] is used to obtain the thermal profile of the circuit. BTI-induced threshold voltage change is estimated by assuming a delay degradation of 15% in 5 years. To show how BTI, Voltage droop, and temperature affect the circuit delay, we consider six different scenarios listed in Table I.

TABLE I
Different Scenarios of Runtime-Variations

1	**-V-T-BTI**	No run-time variations
2	**+V-T-BTI**	Only voltage droop
3	**-V+T-BTI**	Only Temperature
4	**+V-T+BTI**	Only BTI
5	**Superposition**	Simple addition of Scenarios 2,3,4
6	**+V+T+BTI (Proposed)**	Combined effect of Scenarios 2,3,4

Table II shows the circuit relative delay increase (w.r.t. *-V-T-BTI*) due to runtime variations with different schemes. Comparing the seventh (proposed method) and sixth (Simple addition or Superposition) columns of the table reveals that independent analysis of temperature, voltage, and BTI leads to 17% inaccuracy in circuit delay estimation in average. To verify the scalability of our method the runtime is calculated when all of the simulations are performed on a workstation

TABLE II

RELATIVE CIRCUIT DELAY INCREASE (W.R.T. **-V-T-BTI**) DUE TO RUNTIME VARIATIONS ($Error = (Proposed - Superposition)/Proposed$)

Circuit	# cells	+V-T-BTI	-V+T-BTI	-V-T+BTI	Superposition	Proposed	Error	Time(s)
b17	27k	6%	6%	6%	17%	22%	25%	654
b18	88k	9%	6%	6%	21%	25%	16%	978
b19	165k	8%	7%	8%	22%	32%	29%	1071
b22	40k	9%	7%	6%	17%	23%	24%	658
dsp	42k	2%	6%	17%	25%	28%	13%	444
leon2	995k	3%	9%	11%	23%	29%	20%	3245
leon3mp	721k	3%	7%	15%	25%	30%	18%	2458
vga_lcd	114k	5%	16%	21%	41%	48%	14%	1059
risc	61k	10%	10%	13%	33%	39%	16%	754
des_perf	84k	2%	19%	19%	40%	44%	10%	1060
average							17%	

with Intel Xeon E5540 2.53GHz (2 quad-core processors), 16GB RAM. As shown in Table II, even for very large circuits, the runtime of our proposed method is less than an hour.

To investigate the effect of the temperature and voltage variations on BTI, delay degradation is obtained under four different conditions namely: 1) **-V-T** 2) **+V-T** 3) **-V+T** 4) **+V+T** (proposed method). In [19] the effect of temperature (and partially voltage variations) on BTI analysis is well studied. Unfortunately, their proposed timing analysis flow only considers some corner cases. According to Table III, assuming a constant temperature ($T_{nom} = 25°\mathrm{C}$) leads to 10% error in the estimated BTI-induced delay degradation (compared to *+V+T*). Considering a constant power supply voltage ($VDD_{nom} = 1V$) results in 12.8% inaccuracy in estimated BTI-induced delay increase.

TABLE III

THE EFFECT OF NEGLECTING VOLTAGE AND TEMPERATURE ON BTI-INDUCED DELAY DEGRADATION (ERROR ARE CALCULATED W.R.T SCHEME: +V+T)

Circuit	-V-T	+V-T	-V+T
b17	-5.0%	-10.0%	30.0%
b18	-15.1%	-15.9%	1.59%
b19	-11.9%	-13.9%	4.3%
b22	-2.0%	-4.0%	5.0%
dsp	-19.2%	-21.9%	20.6%
leon2	-12.4%	-14.7%	9.5%
leon3mp	-14.0%	-16.7%	16.2%
vga_lcd	-21.4%	-23.8%	9.5%
risc	-1.2%	-6.7%	3.4%
average	-10.2%	-12.8%	10.0%

Input activity due to workload variation influences the voltage and temperature profiles and in turns affects the BTI. Figure 3 shows the circuit delays at different primary input activity factors (0.2,0.5,0.8). Higher inputs activity factors leads to larger circuit delay. This figure also reveals the effect of the representative applications as workloads on the estimated delay. Representative application could be a single program or a basket of n programs. Since different applications lead to different circuit activity and signal probability, in our experiments we consider the worst case scenario to calculate the circuit delay. However, our methodology is general and can take an activity profile as one of its inputs.

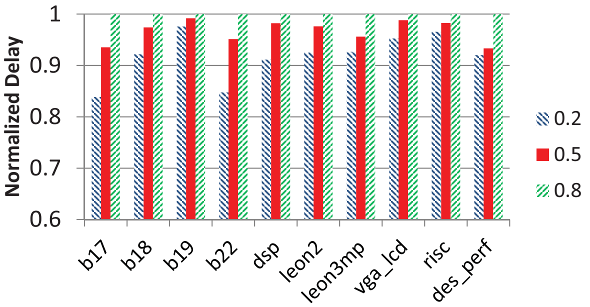

Fig. 3. The effect of activity factor on the circuit delay

IV. CONCLUSIONS

In this paper, we present a static timing analysis methodology to accurately consider the combined effect of short-term and long-term runtime variations including BTI, voltage-temperature variations. This is achieved by performing system-level workload profiling and projecting the information into circuit-level models. The proposed technique which is built on top of commercial EDA toolset scales very well to handle large circuits. The proposed approach can fast and accurately account for workload-dependent runtime variations effects at design time using standard design flow tool chain. We discovered that neglecting the interaction among variation factors results in remarkable underestimation of timing margin.

REFERENCES

[1] W. Wang, S. Yang, S. Bhardwaj, S. Vrudhula, F. Liu, and Y. Cao, "The impact of NBTI effect on combinational circuit: modeling, simulation, and analysis," *Very Large Scale Integration (VLSI) Systems, IEEE Trans.*, vol. 18, no. 2, pp. 173–183, 2010.

[2] M. Gupta, J. Rivers, P. Bose, G. Wei, and D. Brooks, "Tribeca: design for pvt variations with local recovery and fine-grained adaptation," in *Microarchitecture, 2009. MICRO-42. 42nd Annual IEEE/ACM International Symposium on*. IEEE, 2009, pp. 435–446.

[3] "International technology roadmap for semiconductors," http://lpsolve.sourceforge.net/5.5/, November 2010.

[4] J. Jaffari and M. Anis, "Statistical thermal profile considering process variations: Analysis and applications," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 27, no. 6, pp. 1027–1040, 2008.

[5] B. Lasbouygues, R. Wilson, N. Azemard, and P. Maurine, "Temperature-and voltage-aware timing analysis," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, no. 4, pp. 801–815, 2007.

[6] A. Rogachev, L. Wan, and D. Chen, "Temperature aware statistical static timing analysis," in *Proceedings of the International Conference on Computer-Aided Design*. IEEE Press, 2010, pp. 103–110.

[7] S. Bhardwaj, W. Wang, R. Vattikonda, Y. Cao, and S. Vrudhula, "Predictive modeling of the nbti effect for reliable design," in *Custom Integrated Circuits Conference, 2006. CICC'06. IEEE*. Ieee, 2006, pp. 189–192.

[8] P. Li, "Critical path analysis considering temperature, power supply variations and temperature induced leakage," in *Quality Electronic Design, 2006. ISQED'06. 7th International Symposium on*. IEEE, 2006, pp. 6–pp.

[9] H. Su, F. Liu, A. Devgan, E. Acar, and S. Nassif, "Full chip leakage estimation considering power supply and temperature variations," in *Proceedings of the 2003 international symposium on Low power electronics and design*, ser. ISLPED '03. New York, NY, USA: ACM, 2003, pp. 78–83.

[10] "M5," http://www..m5sim.orgt/.

[11] S. Kiamehr, F. Firouzi, and M. Tahoori, "Input and transistor reordering for nbti and hci reduction in complex cmos gates," in *Proceedings of the great lakes symposium on VLSI*. ACM, 2012, pp. 201–206.

[12] K. Haghdad and M. Anis, "Power yield analysis under process and temperature variations," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, no. 99, pp. 1–10, 2011.

[13] S. Nassif, "Power grid analysis benchmarks," in *Proceedings of the 2008 Asia and South Pacific Design Automation Conference*. IEEE Computer Society Press, 2008, pp. 376–381.

[14] S. Krishnappa, H. Singh, and H. Mahmoodi, "Incorporating effects of process, voltage, and temperature variation in bti model for circuit design," in *Proc. IEEE Latin American Symp. on Circuits and Systems*, 2010, pp. 236–239.

[15] "iwls," http://www.http://iwls.org/.

[16] "ispd," http://www.http://ispd.cc/.

[17] "Nangate," http://www.nangate.com/.

[18] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. Stan, "Hotspot: A compact thermal modeling methodology for early-stage vlsi design," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 14, no. 5, pp. 501–513, 2006.

[19] W. Wang, S. Yang, S. Bhardwaj, R. Vattikonda, S. Vrudhula, F. Liu, and Y. Cao, "The impact of nbti on the performance of combinational and sequential circuits," in *Proceedings of the 44th annual Design Automation Conference*. ACM, 2007, pp. 364–369.