

Industrial Approaches for Performance Evaluation Using On-Chip Monitors

Mahroo Zandrahimi*, Philippe Debaud[†], Armand Castillejo[†], Zaid Al-Ars*

*Delft University of Technology, The Netherlands

{m.zandrahimi, z.al-ars}@tudelft.nl

[†]STMicroelectronics, Grenoble, France

{philippe.debaud, armand.castillejo}@st.com

Abstract—To overcome the increasing sensitivity to variability in nanoscale integrated circuits, operation parameters (e.g., supply voltage) are adapted in a customized way exclusively to each chip. A standard industrial approach to achieve customized circuit adaptations is the use of on-chip monitors that allow fast performance evaluation during production or lifetime. Such on-chip monitoring approaches estimate operation parameters either based on responses from performance monitors with no interaction with the circuit or by monitoring the actual critical paths of the circuit. In this paper, we discuss a number of well-known performance monitoring methodologies and compare them with each other in term of their advantages and disadvantages. This enables evaluating the suitability of various performance monitoring methodologies for specific applications based on their respective requirements in terms of accuracy, power efficiency and cost. In addition, we discuss the challenges that these monitoring methodologies face with decreasing node sizes, in terms of accuracy and effectiveness. By simulating ISCAS'99 benchmarks using the Nangate 45 nm open cell library, we show that the accuracy of these approaches is design dependent, and requires up to 15% added design margin.

I. INTRODUCTION

Measurement of operation parameters of integrated circuits can be done either during run-time using online parameter estimation approaches or during production using offline circuit monitoring. Online estimation approaches set the operation parameters for the chip based on the feedbacks they receive from on chip performance monitors. Thus, whenever a change in environmental variations occurs, the system updates the parameter estimation so that all parts of the chip are able to function properly at the target frequency. These approaches are very accurate in estimation and also very efficient in saving power since margins to compensate for environmental variations are measured online. On the other hand, this is rather difficult to implement since the software needs to be manipulated in order to perform online estimation based on the feedbacks received from performance monitors. Furthermore, these techniques are risky for final application since there is a possibility of failure if some parameters are not managed properly.

Offline estimation approaches create a pre-characterized look-up table that links operation parameters to each target frequency. Since parameter estimation for each chip during production should be done as fast as possible, running functional tests on CPU to measure operation parameters for each operating point is not feasible. Moreover, even though using functional patterns for programmable parts of the design such as CPU and GPU is possible, the rest of the design such

as interconnects and USB cannot be characterized using this approach. Hence, performance monitors should be embedded in the chip structure. Based on the frequency responses from performance monitors during production, the operation parameters are estimated exclusively for each operating point of each chip. Then, the margins for temperature and voltage variations as well as aging are added on top of the measured parameters to make sure that the chip works even in the worst-case condition. Although these approaches seem very pessimistic and thus not as power efficient as online approaches, they are very much cost effective and easier to implement since no changes in software is needed. Moreover, offline approaches can be seen as an incremental solution for existing devices, which mitigates the risk of the design.

Regardless of using online or offline approaches, performance monitors should be embedded in the chip architecture so that based on the frequency responses, the operation parameters could be estimated. Many process monitors have been proposed for both online and offline monitoring from simple ring oscillators to more complicated design dependent critical path replicas and in-situ delay monitors. In this paper we evaluate the accuracy and effectiveness of using performance monitors for operation parameter estimation. The contributions of this paper are the following:

- An overview of various on-chip performance monitors for online and offline circuit adaptation including a discussion about pros and cons of each approach.
- Investigation of the limitations of on chip performance monitors in terms of accuracy and effectiveness for ISCAS'99 benchmarks using Nangate 45 nm open cell library with different process corners.

The rest of this paper is organized as follows. Section II overviews process monitoring methodologies. Section III gives some recommendations of suitable process monitoring techniques based on design specification. Limitations on process monitoring methodologies are presented in Section IV using simulation results on ISCAS99 benchmarks. Section V concludes the paper and proposes potential solutions for future work.

II. PROCESS MONITORING METHODOLOGIES

Fig. 1 illustrates a taxonomy of process monitoring methodologies based on various monitoring architectures. According to this figure, circuit adaptation is done either using indirect measurement approaches or direct measurement approaches. Indirect measurement approaches estimate operating parameters through correlating frequency responses of performance monitors to the circuit frequency, whereas, direct measurement approaches set the circuit operating parameters by monitoring the actual critical paths of the circuit [9].

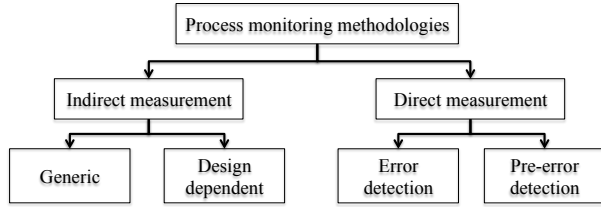


Fig. 1. Classification of process monitoring methodologies

A. Indirect measurement approaches

These approaches embed one or various performance monitors in the chip structure. Due to within-die variations, it is more efficient to place various performance monitors close or inside the block which is being monitored so that all types of process variations are captured and taken into account for parameter adaptation. The number of performance monitors depends on the size of the chip. There is no interaction between performance monitors and the circuit.

To be able to estimate the circuit frequency based on performance monitor responses during production, the correlation between performance monitors and circuit frequency should be measured during characterization, which is an earlier stage of manufacturing [1]. This procedure is done for the amount of test chips representative of the process window to find the correlation between performance monitors and circuit frequencies. Once the performance monitors are tuned to the design during characterization, they are ready to be used for parameter adaptation for each chip during production. Fig. 2 shows an example of a chip with multiple voltage islands, among which performance monitors are distributed. During production, based on the frequency responses from these monitors, the circuit frequency is estimated so that operating parameters can be adapted to each voltage domain of the chip.

Various performance monitoring structures have been proposed from simple generic ring oscillators to more complicated design dependent critical path replicas. The technique presented in [3] implements replica-paths, representing the critical paths of the circuit. Alternatively, the critical path replica can be replaced by fan-out of 4 (FO4) ring oscillator [4] or a delay line [5]. They claim that with varying operating conditions, the timing of monitors will change similarly to the actual critical path. Moreover, the method presented in [6] synthesizes a single representative critical path (RCP) for post-silicon delay prediction. They claim that the RCP is designed such that it is highly correlated to all critical paths for some expected process variations.

However, as the technology scaling enters nanometer regime, specially from 45 nm onwards, finding one unique critical path has become impossible. Depending to the process corner, voltage and temperature variations, and also workload many different timing paths might become critical, therefore, for real circuits the concept of finding one critical path and create a critical path replica as a performance monitor is too simplistic. As a result, regardless of using generic ring oscillators or design dependent replica paths, the characterization phase should be done to find the correlation between monitoring responses and the actual performance of the circuit.

The process monitors, which are widely used today for many products, are ring oscillators designed based on the most used cells extracted from the potential critical paths of the

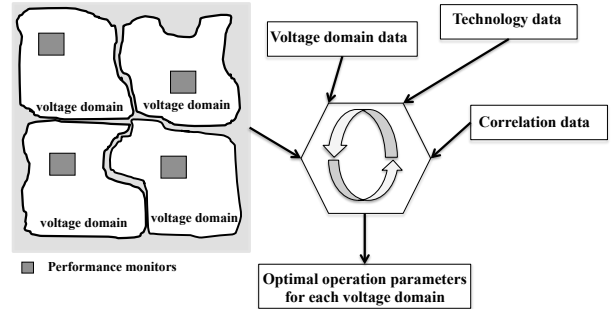


Fig. 2. Operating parameter estimation using indirect measurement approaches

design, reported by static timing analysis. So, based on the design, some standard logic cells are put in an oscillator to form performance monitors, which will be distributed among the chip to capture all kind of variations. During characterization, performance monitors are tuned to the design so that during production, according to the frequency responses of performance monitors, the operation parameters are adapted to each chip.

B. Direct measurement approaches

Direct measurement approaches estimate operation parameters by monitoring actual critical paths of the circuit. These approaches add one in-situ delay monitor per critical path. In-situ delay monitors are special latches or flip-flops, included at the end of critical paths to report the timing behavior of the circuit [7]. Circuit delay characterization using in-situ delay monitors can be done in two different ways. The first is by observing the regular operation of a circuit and to detect timing errors in the circuit itself during operation. With the error information, the critical operation parameters, which are needed for correct operation, can be determined. The second possibility is to observe an over-critical system. Here, a test module which is always slower than the most critical part of the chip is observed, and as soon as the test module fails, the system predicts a delayed data transition called a pre-error [8].

For the in-situ monitors, which are able to detect timing errors, error recovery circuits are needed to repeat single computations after malfunction. In contrast, for in-situ approaches which detect pre-errors, no additional hardware effort and complexity for the recovery circuitry is needed, thus, these approaches are easier to manage. Fig. 3 shows an in-situ delay flip flop which detects pre-errors. These in-situ flip flops detect pre-errors when the timing slack in critical paths drops below a certain value. The idea is to reduce the operation parameters as long as no pre-error is detected and to raise the operation parameters as soon as the pre-error rate is above a certain value.

III. WHICH APPROACH SUITS A DESIGN?

In this section we compare indirect measurement versus direct measurement approaches in terms of accuracy, tuning effort, impact on design planning, implementation risk, and area overhead as illustrated in Table I. With regard to accuracy and tuning effort, direct measurement approaches are very accurate and no tuning effort is needed, since they monitor the actual critical path of the circuit, and there is no need to add safety margins on top of the measured parameters due to inaccuracies. However, for indirect measurement approaches,

TABLE I. COMPARISON OF DIRECT MEASUREMENT VS. INDIRECT MEASUREMENT APPROACHES

Technique	Accuracy	Tuning effort	Impact on design planning	Implementation risk
Direct measurement	high	none	high	medium to high
Indirect measurement	medium	high	low	low

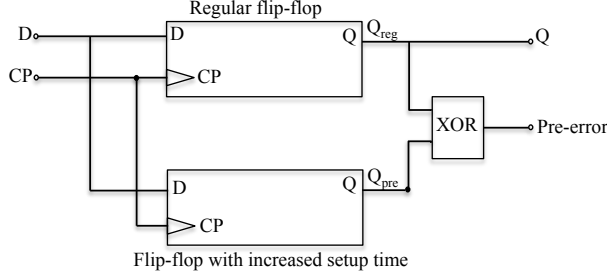


Fig. 3. Structure of in-situ flip-flops which detect pre-errors

since there is no interaction between performance monitors and the circuit, the correlation between performance monitor responses and the actual performance of the circuit is estimated during the characterization phase using the amount of test chips representative of the process window. Since there are discrepancies in the responses of same performance monitors from different test chips, the estimated correlation between the frequency of performance monitors and the actual performance of the circuit could be very pessimistic, which results in wasting power and performance. Hence in terms of accuracy and tuning effort, direct measurement approaches always win.

To validate our claim of low accuracy of indirect measurement approaches, we have done silicon measurement on 625 devices manufactured using nanometric FD-SOI technology [10]. 12 performance monitors (PM) are embedded in each device. First, we have measured the real value of optimal voltage (V_{min}) for each chip using test patterns. Then, we set an arbitrary voltage for each chip and collected frequency responses from all 12 performance monitors. Finally, we mapped each frequency response of a PM to the V_{min} of the chip in which that PM is located. Fig. 4 shows an example of such a plot for one specific PM on all 625 devices measured. To quantify the amount of this discrepancy in this figure, for each value of frequency response, we have looked for the V_{min} variation. We take the maximum amount of this variation as the V_{min} discrepancy for that PM. We measured the amount of V_{min} discrepancy for all 12 monitors, the result of which is presented in Fig. 5. This figure also presents the wasted power as a results of inaccuracy in V_{min} estimation using performance monitors. Results show that minimum voltage estimation based on performance monitors lead to nearly 10% of wasted power on average and 7.6% in the best case, when a single PM is used for performance estimation.

In terms of planning effort and implementation risk, direct measurement approaches are considered very risky and intrusive since adding flip-flops at the end of critical paths requires extensive modification in hardware and thus incurs a high cost. Moreover, for some sensitive parts of the design, such as CPU and GPU, which should operate at high frequencies, implementing direct measurement approaches is quite risky since it affects planning, routing, timing convergence, area, and time to market. On the other hand, indirect measurement approaches are considered more acceptable in terms of planning and implementation risk, since there is no interaction between performance monitors and the circuit, hence, performance monitors can even be placed outside

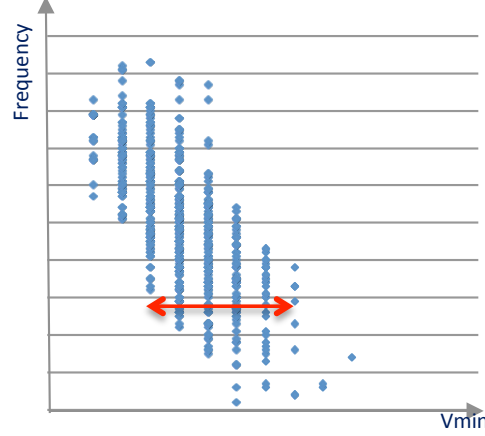


Fig. 4. Example of V_{min} discrepancy for one PM on all 625 devices measured

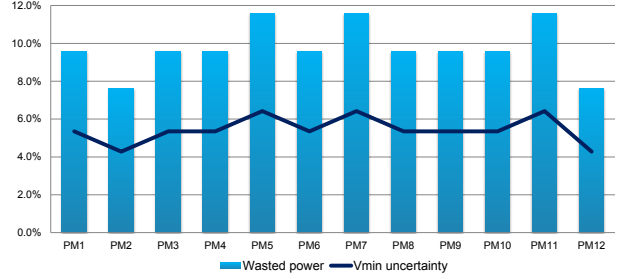


Fig. 5. Inaccuracy in the minimum operating voltages estimated using different performance monitors [10]

the macros being monitored, but not too far due to within die variations. Consequently, indirect measurement approaches seem more manageable due to the fact that they can even be considered as an incremental solution for existing devices and the amount of hardware modification imposed to the design is very low. Consequently, according to the application, one can decide which technique more suits a design. For example, for medical applications accuracy and power efficiency are far more important than the amount of hardware modification and planing effort, while, for nomadic applications, such as mobile phones, tablets, and gaming consoles, cost and the amount of hardware modification are considered the most significant.

IV. LIMITATIONS OF INDIRECT MEASUREMENT APPROACHES

As we discussed earlier, indirect measurement approaches estimate operation parameters based on responses from performance monitors with no interaction with the circuit. In deep sub-micron technologies, performance monitors are showing limitations to accurately estimate the silicon performance. Within die variations and the amount of parameters that should be taken into account tend to prevent accurate computation of needed optimum operation parameters for a given target frequency. To investigate the variability of critical paths of a design in different corners, first we present an industrial case study regarding critical path variability of a nanometric FD-SOI device through static timing analysis. Next, in order

TABLE II. PERCENTAGE OF CLOCK PERIOD SPENT ON 5000 MOST CRITICAL PATH IN 16 CORNERS

Corner	% of clock period	Corner	% of clock period
1	13.63	9	13.42
2	13.95	10	6.34
3	4.86	11	9.13
4	11.60	12	12.41
5	9.55	13	15.59
6	9.08	14	9.89
7	12.47	15	17.02
8	4.75	16	8.46

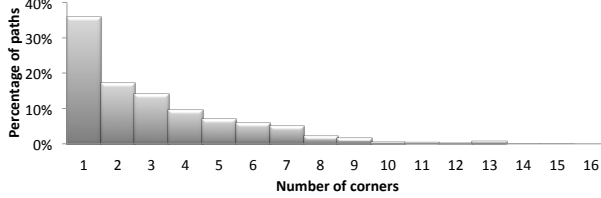


Fig. 6. Percentage of unique paths out of the 5000*16 critical paths present in 1 to 16 corners [10]

to generalize the idea of critical path variability as a result of process and environmental variations, we back up the industrial case study through simulation results on ISCAS'99 benchmarks using Nangate 45 nm open cell libraries.

A. Case study

We have done timing analysis on a nanometric FD-SOI device in sixteen corners with different process and environmental conditions [10]. For each of the sixteen functional corners, we have extracted the 5000 most critical paths of the device. The path lists are sorted from the most critical path to less critical.

In order to understand if five thousand paths are enough for our study, we have computed the distribution of these paths compared to the clock cycle. The objective is to check whether the spread of 5000 paths represents very small part of the clock cycle, which requires to increase the number of paths or is considered enough. For each corner, we have computed paths spread as follows:

$$Spread = (slack_{5000} - slack_1) / (T_{clock}) \quad (1)$$

where $slack_{5000}$ is the slack of the 5000th critical path, $slack_1$ is the slack of the most critical path, and T_{clock} is the clock period. Table II presents the percentage of clock cycle spent on the 5000 most critical path in 16 corners. As it can be seen in this table, depending on the corner, the spread of 5000 paths spans the range from 4.75% to 17% of the clock period, which is considered as enough for our study.

From the sixteen lists of 5000 critical paths, we have extracted the total number of unique paths. We have found 25936 unique paths out of 5000*16. Fig. 6 shows the percentage of the 25936 paths present in 1 or more corners. In this case, only 35.8% of paths are present in 1 corner, and only 53% are present in one or two corners. Two third of the paths are present in maximum 3 corners. None of the paths are present in the list of critical paths of all 16 corners, which means it does not matter which critical path we choose, it does not stay critical even within 5000 most critical paths of all corners.

These results show that identifying a critical path that covers all the corners is not possible. Therefore, when a path

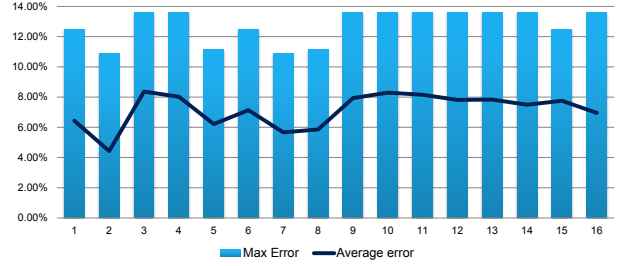


Fig. 7. Performance estimation error using the critical path of another corner [10]

is the most critical in a corner, it is important to know how this path is changing across various process, voltage and temperature conditions. Suppose that P_x is the critical path of corner X , P_y is the critical path of corner Y . First, we have computed the distance of the P_x from P_y for all 16 corners against each other in terms of delay. Then, we measured the maximum as well as the average error for each corner if we assume that the critical paths of other corners are the most critical in that corner. Fig. 7 presents average and maximum error measured when the critical path of corner X is used to evaluate performance in corner Y . Results are presented in % of clock period and have been clamped to the value of the 5000th path of the corner Y list. Based on these results, whatever the critical path and the corner we take, maximum error is above 10% of the clock cycle.

B. Simulation set up

This subsection explains the definition of parameters in order to characterize the simulation results. We use Nangate 45 nm open cell library [11] to investigate critical path variability on ISCAS'99 benchmarks [12] using Cadence RTL Compiler. ISCAS'99 contains 29 designs from small circuits with 21 cells to more complicated designs with almost 44 K cells. Nangate 45 nm library contains 5 different process corners with different characteristics in terms of process and environmental variations. These corners are typical, fast, slow, low temperature (low), and worst low (worst).

In order to characterize the results, we defined a parameter named $error_{max}$ which is measured for each design. If we assume the critical path of each design is the critical path of the typical corner, $error_{max}$ is the maximum percentage of critical path delay change when measured in the other corners. The concept relates to how much margin should be taken into account due to inaccuracies as a result of critical path variability in different corners, if we assume that for each design the critical path remains critical in all process corners. To be able to measure $error_{max}$ for each design, first we check if the critical path in each corner is different from the critical path of the typical corner. In the case of critical path difference, we measure $error_{corner}$ for the process corner by:

$$error_{corner} = (P_{corner} - P_{typ}) / P_{corner} \quad (2)$$

where P_{corner} is the delay of the critical path in that corner, and P_{typ} is the delay of the critical path of the typical corner in that corner. Once $error_{corner}$ is measured for all process corners, $error_{max}$ can be obtained for the design by:

$$error_{max} = \max_{all\ corners} [error_{corner}] \quad (3)$$

TABLE III. PERCENTAGE OF $error_{max}$ FOR ISCAS'99 BENCHMARKS USING NANGATE 45 NM LIBRARY

Benchmark	# Cells	$error_{max}$	Benchmark	# Cells	$error_{max}$
b01	30	6.93	b15	3142	0
b02	21	0.10	b15_1	3141	0
b03	76	11.65	b17	9559	0
b04	196	6.29	b17_1	9584	0
b05	390	2.85	b18	22175	15.03
b06	29	1.35	b18_1	22093	0
b07	179	0.84	b19	43916	9.24
b08	71	0	b19_1	43822	0.23
b09	94	0.52	b20	3970	0.69
b10	110	0	b20_1	4025	0.71
b11	326	0	b21	4022	0.72
b12	547	4.19	b21_1	4082	0.71
b13	154	0	b22	6102	1.12
b14	1967	0.69	b22_1	6164	0.74
b14_1	2043	0.67	-	-	-

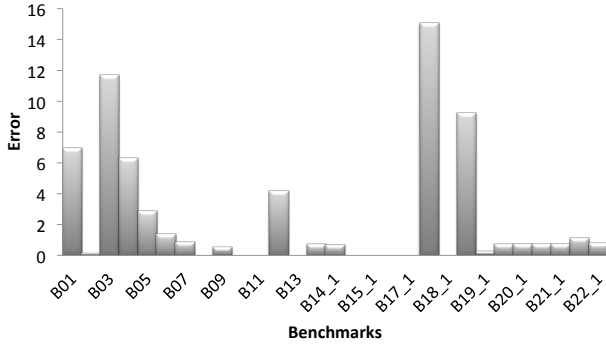


Fig. 8. Percentage of $error_{max}$ for ISCAS'99 benchmarks using Nangate 45 nm library

To further elaborate on how error is measured for each design, here we calculate error for one of the benchmarks (b03) with 76 cells. The delay of the critical path of the design in typical corner is 678ps. We name this path as P_{typ} . In the fast corner, P_{typ} is not critical anymore. It drops to the 55th path with the delay of 424ps, while the delay of the critical path of the fast corner (P_{fast}) is 453ps. So, $error_{fast}$ can be measured by:

$$error_{fast} = (453 - 424)/453 = 6.40\% \quad (4)$$

In the slow corner, P_{typ} stays critical, thus $error_{slow}$ equals to zero. For the low temperature corner, P_{typ} drops to the 247th path, and for the worst low corner, P_{typ} drops to the 12th path, hence the errors can be measured in the same way. $error_{low}$ equals to 11.65%, $error_{worst}$ equals to 2.12%. Consequently, $error_{max}$ is obtained by:

$$error_{max} = \max[error_{fast}, error_{slow}, error_{low}, error_{worst}] \quad (5)$$

$$= \max[6.40\%, 0, 11.65\%, 2.12\%] = 11.65\% \quad (6)$$

The $error_{max}$ is measured for all 29 ISCAS'99 benchmarks, the result of which is presented in the next subsection.

C. Simulation results

Fig. 8 illustrates the $error_{max}$ for all 29 ISCAS'99 benchmarks. As shown in this figure, although for some designs the error is zero or negligible, for some other designs the error is rather high and for one case, b18, it even reaches 15%. Table III presents the detailed simulation results for all 29 ISCAS'99 benchmarks. According to this table, it is not possible to find

a unique critical path for most designs, which stays critical in all 5 corners. Therefore, in order to investigate if the error further can be reduced, we took into account all the paths, which become critical in different corners for performance evaluation.

In order to discover if the error can be reduced for the designs with non-zero $error_{max}$, we estimated delay of each design based on all critical paths in all corners as well as the average critical path delay. To further elaborate, we perform the procedure for benchmark b01 as an example. Let $P1$, $P2$, and $P3$ be the paths of b01 that become critical in one or more of the 5 process corners. As it can be seen in Table IV, $P1$ is the critical path of the typical and slow corners; $P2$ is the critical path of the fast and low temperature corners; $P3$ is the critical path of the worst low corner. $P1P2P3$ is the average delay of these three critical paths. We let the circuit delay in each corner be the maximum delay of all critical paths ($delay$). We performed a linear least square regression analysis of the correlation between circuit delay and the delay for each critical path as well as the average critical path delay. The 4 regression functions are defined as:

$$est_{P1} = Func1(P1) \quad (7)$$

$$est_{P2} = Func1(P2) \quad (8)$$

$$est_{P3} = Func1(P3) \quad (9)$$

$$est_{P1P2P3} = Func1(P1P2P3) \quad (10)$$

Based on these 4 functions, we computed the delay of the circuit as est_{P1} , est_{P2} , est_{P3} , and est_{P1P2P3} . The estimated delay of b01 is defined as the maximum value of 4 estimations in each process corner (column est_{max} in the table).

For the est_{max} values, we calculated the estimation errors ($error_{est}$) as the difference between est_{max} and $delay$, as shown in Table V. According to the table, although we considered all critical paths of b01 to estimate the circuit delay, there is still an estimation error of up to 4.5% in delay estimation for different process corners ($error'_{max}$). We performed the same procedure for all benchmarks with non-zero $error_{max}$, the results of which are presented in Table VI. Based on this table, the error can be reduced up to 98.8%, which is for benchmark b07. However, although we estimated the design delay considering all critical paths of all corners, there is still some unacceptable error present for some designs such as b18. The error of b18 is reduced by 47.11%, remained 7.95% out of 15.03%, but still this error is not negligible.

Furthermore, simulation does not fully reflect the actual variations on manufactured silicon. On a physical circuit, other sources of variation, such as within-die variations and IR-drop could promote paths which are not reported as critical by static timing analysis, but will become critical on real silicon. Table VII illustrates paths which are ranked top 9 in one of the corners and the highest ranking of that same path in all other corners. According to this table, a path ranked 1 in one corner, drops above the rank 5000 in one of the other corners. Therefore, for more accurate delay estimation, more paths should be taken into account. The more paths we can cover, the more accurate the delay estimation will be.

We further investigated on the reason of the variability in $error_{max}$ for different designs. Each gate behaves differently when being exposed to process and environmental variations. Thus, corner changes incur a different error value to each design according to the gate structure of the critical path

TABLE IV. DELAY ESTIMATION IN [PS] OF BENCHMARK *b01* USING CRITICAL PATHS OF ALL CORNERS AS WELL AS THE AVERAGE CRITICAL PATH

Corner	$P1$	$P2$	$P3$	$P1P2P3$	$delay$	est_{P1}	est_{P2}	est_{P3}	est_{P1P2P3}	est_{max}
Typical	360	356	354	356,66	360	368,26	367,58	367,73	376,21	376,21
Fast	226	238	235	233	238	235,85	233,96	232,87	245,76	245,77
Low	188	202	199	196,33	202	198,30	193,19	192,08	207,09	207,09
Slow	1158	1052	1049	1086,33	1158	1156,76	1155,73	1155,31	1145,86	1156,76
Worst	316	326	326	322,66	326	324,78	333,61	336	340,35	340,35

TABLE V. ERROR ESTIMATION OF BENCHMARK *b01* FOR ALL CORNERS

Corner	$error_{est}$
Typical	4.5
Fast	3.26
Low temp	2.52
Slow	0.11
Worst low	4.40
$error'_{max}$	4.50

TABLE VI. ERROR ESTIMATION OF ISCAS'99 BENCHMARKS WITH NON-ZERO $error_{max}$ USING CRITICAL PATH OF ALL CORNERS AS WELL AS THE AVERAGE CRITICAL PATH

Benchmark	$error'_{max}$	reduction	Benchmark	$error'_{max}$	reduction
<i>b01</i>	4.50	35.1	<i>b18</i>	7.95	47.11
<i>b03</i>	6.40	45.1	<i>b19</i>	1.94	79.00
<i>b04</i>	3.04	51.7	<i>b19_1</i>	0.08	65.22
<i>b05</i>	1.83	35.8	<i>b20</i>	0.35	49.28
<i>b06</i>	1.19	11.8	<i>b20_1</i>	0.48	32.4
<i>b07</i>	0.01	98.8	<i>b21</i>	0.46	36.1
<i>b09</i>	0.35	32.7	<i>b21_1</i>	0.48	32.4
<i>b12</i>	1.30	68.9	<i>b22</i>	0.46	58.9
<i>b14</i>	0.47	31.9	<i>b22_1</i>	0.48	35.14
<i>b14_1</i>	0.47	29.8	-	-	-

of the design. To prove this point, we designed one of the benchmarks, *b03*, using only NAND logic. As it can be seen in Table III, *b03* is a small design with 76 cells, but the $error_{max}$ is rather high, 11.65%. By designing using only NAND logic, the error dropped to 0. However, since there is no simulated variation of RC delay in different process corners of Nangate 45 nm library, in actual circuits, a small error might be present in this case as well.

V. CONCLUSIONS AND FUTURE WORK

For some products such as nomadic applications, cost and design customization effort are considered significant. Despite the accuracy and effectiveness of direct measurement performance monitoring approaches, cost versus benefit is not proven since the implementation risk and the impact on design planning is high. Thus, indirect measurement performance monitoring approaches are considered more manageable for

TABLE VII. TOP 9 CRITICAL PATH RANKING OF *b14* IN DIFFERENT CORNERS

Least rank	Highest rank	Least rank	Highest rank
1	297	5	869
1	>5000	5	18
1	26	6	862
2	646	7	1165
2	56	8	902
3	496	8	21
3	71	8	34
4	2493	9	423
4	3967	9	4902
4	27	9	47
5	1429	-	-

many low cost products. However, in deep sub-micron technologies, indirect measurement approaches are showing limitations to accurately estimate silicon performance, which leads to unnecessary power loss. Based on simulation results on ISCAS'99 benchmarks as well as static timing analysis of a nanometric FD-SOI device, we showed that depending on the design, critical path can change dramatically as a result of PVT variations. Thus, the accuracy and effectiveness of indirect measurement approaches is low.

Our future work will concentrate on solutions to avoid these limitations. One possible solution could be using delay test patterns for delay estimation of a design. The main challenge of using test patterns for delay estimation is that there should be a reasonable correlation between delay test patterns and functional test patterns. Test time should also be reasonable compared to the indirect measurement approaches which are very fast during production.

ACKNOWLEDGEMENTS

This work is carried out under the BENEFIC project (CA505), a project labelled within the framework of CATRENE, the EUREKA cluster for Application and Technology Research in Europe on NanoElectronics.

REFERENCES

- [1] T. Chan and A.B. Kahng, *Tunable Sensors for Process-Aware Voltage Scaling*, in ICCAD, pp. 7-14, 2012.
- [2] T. Chan, et al., *DDRO: A Novel Performance Monitoring Methodology Based on Design-Dependent Ring Oscillators*, in ISQED, pp. 633-640, 2012.
- [3] A. Drake, et al., *A Distributed Critical-Path Timing Monitor for a 65nm High-Performance Microprocessor*, in ISSCC, pp. 398-399, 2007.
- [4] T.D. Burd, et al., *A dynamic voltage scaled microprocessor system*, in ISSCC, pp. 294-295, 2000.
- [5] J. Kim and M.A. Horowitz, *An efficient digital sliding controller for adaptive power-supply regulation*, in IJSSC, vol. 37, no. 5, pp. 639-647, 2002.
- [6] Q. Liu and S.S. Sapatnekar, *Capturing Post-Silicon Variations Using a Representative Critical Path*, in TCAD, vol. 29, no. 2, pp. 211-222, 2010.
- [7] M. Wirthofer, et al., *A Variation-Aware Adaptive Voltage Scaling Technique based on In-Situ Delay Monitoring*, in DDECS, pp. 261-266, 2011.
- [8] M. Eireiner, et al., *In-Situ Delay Characterization and Local Supply Voltage Adjustment for Compensation of Local Parametric Variations*, in IJSSC, vol. 42, no. 7, pp. 1583-1592, 2007.
- [9] M. Zandrahimi and Z. Al-Ars, *A Survey on Low-power Techniques for Single and Multicore Systems*, in ICCASA, pp. 69-74, 2014.
- [10] M. Zandrahimi, et al., *Challenges of Using On-Chip Performance Monitors for Process and Environmental Variation Compensation*, in DATE, 2016.
- [11] <http://www.nangate.com>
- [12] <http://www.cad.polito.it/downloads/tools/itc99.html>