Timing analysis journey from OCV to LVF

Mahajan Rita¹, Sharma Aru², and Bansal Manish³

¹,²Punjab Engineering College Chandigarh, India
³STMicroelectronics, Greater Noida, India
Email: ritamahajan@pec.ac.in, arusharma854@gmail.com, manish.basal@st.com

Abstract. In the present era, robust, reliable, and efficient SOCs are the need of the hour due to the rapid growth of IoT, networking, artificial intelligence, and many newer technologies. There is huge competition in the market to decrease the size of the electronic devices. All these market demands have put lots of pressure on silicon industries to accommodate more functionality on a single chip. As a result of this, with time, there is very fast shrinkage in the technology node. But this technology advancement comes with plenty of challenges. One of the primary challenges is undesired on-chip local variations (OCV) & its accurate modeling for timing analysis (STA). This paper discusses all the aspects of timing analysis and the evolution of different methodologies that shape timing analysis over the course of time. The methodologies to define margins for accurate timing analysis is the heart of accurate analysis at lower nodes. The paper will include an emphasis on how the timing analysis have moved from a global derate value to the architecture, condition-based derate value, used through all these methodologies. The paper will focus on the concept of all of the important methodologies and how they have evolved through time.

1. Introduction
From the early stage of chip manufacturing, timing analysis is one of the important steps which ensure the accuracy and performance of the chip. To ensure this, Static Timing Analysis originates which not only answer all the questions regarding the timing analysis but also plays a significant role in the timing optimization. Previously industries were using a conventional STA tool, which was not analyzing the effects of the variation on the timings.

The variations were always there, from the day first chip was manufactured, but at that time, it was not something that bothers the functionality. But with development in the chip industry and an increase in a number of devices on the chip, these variations start affecting the design functionality. Engineers started working in this direction. Previously, at higher nodes, the variation effect was only considered to be global, and the PVT based models were working fine. But with the node shrinkage, the effect of process variation becomes much pronounced. With the work towards the nanometer regime, different models were formed named OCV, AOCV, POCV, LVF so that the chip may not fail at the timing signoffs. This paper also describes how these models evolve through time.

In Section 2, the root cause of these "variations" is discussed, how they originated, and tackled. In Section 3, it is described how these variations came out to be a problem. In Section 4, all the evolved methodologies are discussed, later Section 5 concludes with the comparison between the first solution to this problem, and the one we have reached through the course of the time. We will sum up this review paper with a conclusion in Section 6.
2. Root Cause
In this section of the paper, light is thrown on how these variations came into the picture of timing predictions and why we need such rigorous timing analysis models. Firstly, we will discuss where these variations originate as we move from design to silicon. After that, we will discuss them in some detail and the challenges we have overcome to meet our timing closures.

2.1. Encounter and Amelioration
With technological advancements like automation, machine learning, and other technologies, the chips are designed with great accuracy and efficiency. But the actual accuracies of all the work done can be analyzed only when the design match with the silicon. After fabrication, some chips fail even after closing the timings accurately, and this is attributed to the variations which came after fabrication and are termed as process variations. Even due to the different environmental conditions, the timings of the design may vary.

Before discussing how these variations are modeled. We will first discuss them in some detail how these variations originate. Firstly, process variations are discussed. Process variations occur mainly due to equipment faults and process limitations. After fabrication, no two devices on the chip behave similarly. Such behavior is accounted directly to process variations and is summed up mainly in two types: 1. Intra Die called as Local Variations. 2. Die-Die, named as Global Variations [1]. To clarify more, we can describe Local Variations as the variations that cause an inverter placed on the same chip at two different locations, on the same die, to give different propagation delays. Random dopant fluctuations, Oxide thickness variations, Metal thickness variations, Channel Length variation, which may have happened during Lithography, Implanting, Annealing, Diffusion, etc. are one of the reasons. All these directly affect the characteristics of the devices. All these kinds of variations are induced at the atomic level, and it is very difficult to tackle these variations with great efficiency, whereas the Global variations are the variations between the devices, which are on different Lots, wafers, and dies. The reason for these variations is also accounted for the process variations. Process Corners was the answer to the global variation problem, and these corners were defined by the foundries. The analysis was done to tackle the global variations that include all the aspects of deviation. Apparently, the work in the fabrication labs and foundries started.

Foundries model these variations by intentionally fabricating the test chips way out of specifications in one direction or another. Then the electrical characteristics of the transistors on these chips were measured, and a set of specifications was given.

These specifications can be represented using models, usually called Process Corner Models. These process corners are able to generalize the behavior of the devices in different conditions. Timing libraries are already characterized by different process corners. Process Corners manages the delays due to variation by defining corners as nominal (tt), slow (ss) and fast (ff). Process corner based approach ensure that the two devices, on different chips, will behave according to the corner they are working on, despite on the fact that both are manufactured in different condition for say fabricated in a different country. Some global variation can also be there if the chips work in different environmental conditions, which is attributed to Temperature and Voltage variations. Foundries not only provide models for these variations accurately, along with the global variations process corners but also define corners with voltage and temperature variations. And these models are called PVTs. Timing libraries provide timings at defined PVTs, so that accurate timing closures can be done.

Since the timing variations due to global Variations are modeled with this PVT based approach. With the advancement in scaling, at lower nodes, it became extremely important to model the effect of Local variations also. But this global corner based (PVT) approach didn’t work well for the local variation. Local variations also needed to be modeled for these global corners.

Local variations effect can be defined in two ways [2]
- Random effects
- Systematic effects
Random Variations are mostly attributed to the functioning at the atomic level. Obtaining accurate results at this level is very difficult. But to work to some extent, foundries provide Gaussian distribution, which needs to be closed in design methodology.

Systematic Variations are due to the spatial variation. Variations tend to affect the closely placed devices in a similar manner, making them more likely to have similar characteristics than those placed at some large distance. Evolution of different techniques that model the local variation effect is discussed in the next section, which is the essence of this paper.

3. Problem Formulation and Basic Solution Approaches

3.1. Problem Formulation
In the chip designing industry, the crucial phase in the design process is the efficient timing closures. These timing closure processes determine if the chip meets the critical time to market window or not. The solution to the problem was answered by doing STA on the chip.

3.2. Solution Approaches
3.2.1. Static Timing Analysis. In digital circuits, all the paths from primary input to primary output are analyzed. An upper bound is computed on the delays to enable the circuit to work at all the conditions, irrespective of the inputs. Static simulations are carried out to obtain this upper bound, and these simulations are described as Static Timing Analysis [3]. Delay information is characterized by finding the critical path of the circuit, with the help of highly efficient STA.

In the circuitry shown in figure 1. With STA, we will get the latest time at which second flop (FF2) can be clocked, so that valid data is being latched up in all components.

![Figure 1. Represents a simple circuit with two ideal flipflops and some combination circuitry between them.](image)

Pre characterization of all the combinational logic is done to ensure these timings, before doing STA. The delay timings obtained through characterization is either modeled through equations or stored in a look-up table. Since delay varies with different conditions like input slope, fanout, output capacitive load, so the pre characterization process involves the circuit simulation at the different power supply, temperature, voltages, and loading conditions [4]. Delay data obtained by these simulations are abstracted into the timing model for each block.

This Analysis carried out in two phases. In the first phase, for each signal's latest arrival time is labeled, this latest arrival time is the time at which all the correct digital values can be guaranteed. The latest arrival time is computed by propagating forward the delay of each signal through the combinational block, taking the delays from the pre-characterized delay models. The wire delays are also considered. Another step is to calculate the required arrival time, which is defined as the latest time at which the signal must have its correct value so as to meet the correct timing requirements.

The difference between the required arrival time and the actual arrival time is called slack. After that, signals are sorted on the basis of their slack. Negative slack will give an indication of low performance. The analysis is further enhanced by computing early and late arrival time. The minimum
and maximum delay of each block can be characterized to do this analysis. The early mode is defined with the best case for the arrival time of input signal to a block, and late mode is computed using the worst scenario [5].

Timing analysis at different PVTs is not naïve. The designer has to work on the worst-case scenario to ensure the functionality in these operating conditions. But with the variation problem, these methodologies fail. New methodologies were developed to model the variations. These approaches beautifully evolve through time. Nowadays, the timing closures are provided by defining derates. Derating is applied to the timings across the circuit path, so that path timings take the effect of variations and will able to follow all the timing constraints. Derate values to reduce the pessimism, provided for timing considerations evolve along with these models to reduce the pessimism. Many methodologies like on-chip variation (OCV, for lower nodes) AOCV, POCV, and LVF are being used till now. All these methodologies are discussed in the next section.

4. Methodologies

4.1. On Chip Variations
On chip variation analysis was done around 130-nm node. The effect of different variations is compensated by adding margins to the timing paths. In OCV the derate applied to define the margins is same for all the paths. Calculated margins are applied to clock and data paths as per setup and hold constraints. The margins are technology-specific and are provided by the DK models. DK models analyze the process variations like oxide thickness, channel length, and provide a derate value. On applying these derate values, the OCV model was able to work properly, but only at upper nodes. Since the effect of variations cancels out each other as the logic depth increases [9]. Application of the same derating factor for different path lengths makes OCV a pessimistic approach for longer paths and an optimistic approach for shorter paths. The pessimistic approach of OCV is one of the major disadvantages. Considering this approach at lower nodes start affecting the performance and area of the chip adversely.

The maximum frequency of the design may be limited by the excessive setup derating, affecting the performance. For hold timing, excessive buffering may be required to meet pessimistic hold margins, directly affecting the area. At 40 nm and below, these effects became more adverse. To overcome these, Statistical Analysis methods emerged which overcame the pessimism associated with OCV.

![Figure 2. Description of global derate applied across the entire path in OCV. The nominal timing value is shown to be increasing with path depth, along with the corresponding derated value.](image-url)
4.2. Statistical Analysis

4.2.1. Monte Carlo. With the advancement in node scaling below 65nm, the effect of the variations became much pronounced. OCV results were way too pessimistic. Statistical approaches were considered to overcome the limitations of the OCV. Monte Carlo is also one of the statistical models and is proved to be one of the best ways to model the variations statistically [5]. Monte Carlo, in its analysis, does a number of trials, which means full-scale circuit simulations. Different process parameters are simulated with their range of variations given by the DK models, and these DK models are techno specific. The required Process parameter is sampled from its distribution for simulation, and then the delay is computed by performing STA. The process is repeated over thousands of trials, and from the set of output delays, the delay distribution is obtained. Trials are done repeatedly to predict the distribution with measurable confidence. From the distribution, the fraction of samples for which timing constraints are satisfied is taken, and from them, timing yield is estimated. The results of Monte Carlo are very accurate as no approximations were made. Monte Carlo simulations are computation intensive [7]. At lower nodes, the variations factors increase drastically, increasing its complexity further. All this computation complexity makes it impractical for analyzing the actual design. In such a situation, another method named Statistical Static Timing Analysis proved to be much efficient. The random variations are modeled as Random variables. Circuit delay is modeled as the probability density function. The results obtained for both are equivalent, with SSTA having the benefit of lesser time and less computation complexity.

4.2.2. Statistical Static Timing Analysis. To overcome the drawback of sample space enumeration based Monte Carlo, new methods that define the delays, slacks, arrival times as probability distribution, are used. These methods are named as Probability Analysis Methods. All the delays are defined as the random variable, and calculations are done with probability distribution function. The advantage of doing so is that the circuit performance under the influence of variations can be predicted with single timing analysis[6]. All the variations like With in die variations, aging, hot electron effect, electro migration can be modeled with the probabilistic techniques.

In this, graphical analysis representation of the gates and the interconnects is done. All the analysis is done statistically. The Delay of each gate is considered random due to variations and is represented as PDF. The delay of the whole circuit is represented as a statistical sum of the gates, which are present in the path[8]. SSTA was done about 65nm, and libraries were created statistically using the SSTA model. SSTA models were developed, but they fail to impress the designers due to its probabilistic nature, more complexity, and also high expense. After SSTA, Advanced OCV came into the picture. AOCV was able to gain popularity due to its efficiency, as this model was able to reduce the pessimism associated with OCV margins.

4.3. Advanced On Chip Variation

The AOCV comes with the solution of many questions which remain unanswered while working on the OCV models. Many questions like how we can obtain efficient margins with the same derating when it is evident that large variation is present even between adjacent cells. And why we didn't consider the cancellation effect of variations in our models. AOCV, in opposite to OCV, uses context-specific derate value in place of single global derate[9]. Hence AOCV adopts a practical and realistic approach that helps in getting the timings in an efficient way. AOCV also reduces the complexity as present in SSTA, thus reducing the timing closure cycles. AOCV calculates the derate value, with the help of two parameters 1.Logic Depth 2. Cell/net locations. Accurate analysis of these parameters adds more quality to the derating factor. A general approach for AOCV was that the variations vary with cell position and depth, and global derating can’t be applied. AOCV approach comes with the insight that the effect of random variations on any path depends upon its logic depth, and the effects of systematic variations are proportional to the cell location of the path being analyzed. Since, previously with OCV, we consider that random variation effects are position-independent, and now we conclude that it is proportional to path depth. This contradictory behavior arises due to the
cancellation effect of random variations. Path depth can be defined as the position of the transistor in a series of instances. The density of instances in the center is quite huge in comparison to peripheries, so random variation cancellations increase with an increase in path depth. So random variations at boundaries are considerable while in the center are negligible. For systematic variations, it increases with path distance, and this can be explained as the systematic variations for two cells that are placed closer to each other will be less as compared to the cell, which is farther in the distance. Thus increasing the derate value. To reduce the pessimism and to obtain accurate results, Monte Carlo analysis is performed to measure the delay variations at each stage. Derating factor can then be computed efficiently. Derating values are present in the tables to close the timings, through AOCV, corresponding to the Logical depth and distance. The figure 3 shows the variation of the derating factor with increasing path depth. To close the timings through AOCV derating values are present in the tables corresponding to the logical depth and distance. There are two approaches, according to which derate is applied to circuits

- Path-based analysis.
- Graph-Based analysis.

4.3.1. The graph-Based analysis. It is encapsulated almost in all the tools. By default, almost all the timing engines are based on Graph-Based Analysis. To understand how AOCV derating is applied to GBA based analysis, we will take an example. Consider a circuit, as shown below in the figure 4.

From the figure 3, we can see that there are three paths present. For analysis, we will consider two paths FF2 to FF4, and another one is FF3 to FF4. Logic depth of the first path a is assumed here to be X, the derating applied to all the combinational circuits between this path is according to logic depth X.

Now, if we consider the second path from FF3 to FF4, i.e., path b the logic depth is Y. Expectations here are that the derating will be applied according to the depth of the cells only, i.e., X. But this will not be the case since from U4 to U7, and common elements are present, the tool will do the worst-case analysis and derate will be applied according to the logic depth of Y.

As a result, the path from FF3 to FF4 will become pessimistic. This is one of the disadvantages of the AOCV approach, as the logic depth increases derate will be applied to each cell depending upon the shortest path that goes through the cell, adding pessimism to other paths that go through the cell.

Figure 3. Description of AOCV derates, applied across the entire path. The nominal timing value is shown to be increasing with path depth, along with corresponding derated value, which decreases with the path depth.
4.3.2. **Path-Based Analysis.** For the removal of pessimism and for the refinement of critical paths, both the Graph-based AOCV and path-based AOCV are recommended. PBA is performed after a few iterations of GBA so as to avoid the extensive computation of PBA AOCV. Path specific calculations to obtain the derate value are done for the critical path, which includes Path specific path-depth and location-based bounding box values. PBA for the type of circuit having some of the path common for lunch and capture, is carried out as up to common point effect of variation is considered to be zero, as the path is common for both of launch signal and capture signal, from the common point PBA, cell path depth is computed, and derate is calculated from the table which is defined for that path depth. Path distance analysis is done separately for the nets and the cells. Since the location of the nets, have no effect on the systematic variation. The net path distance is calculated by diagonal of the box that consists of all the nets in the path[10]. Similarly, for cell, the path distance is calculated. The cell and net bounding box distances are characterized to model the effect of systematic variations between the cells. With this recalculation in PBA, the accurate derating is applied, reducing pessimism to a great extent. Due to large computational complexity, and the pessimism addition in graph-based analysis, AOCV does not prove to be an efficient way to do the timing analysis. A new technique was required to close the timings for the node below 20nm. Further advancements were made, and a new approach Parametric OCV comes into the picture.

![Figure 4. Showing a simple example, having common elements with different logic depth.](image)

4.4. **Parametric On Chip Variation**

POCV utilizes a single statistical parameter sigma, which represents variation. The sigma value is characterized for each cell at each timing arc. The variations are calculated through Monte Carlo HSPICE simulations. Previously for OCV and AOCV, we work around calculating min-max delay values for timing arcs. But POCV works more precisely, on the nominal value. As with POCV analysis, there are maximum chances for the delay variations to fall near the nominal value. POCV does statistical analysis to calculate the arrival time and required time. In POCV analysis, the nominal delay represented by µ and variation sigma σ is analyzed by the tool. Most of the time, POCV analysis calculates the variation at 3sigma from the mean value. The sigma calculations for POCV is done by applying a single coefficient to each of the library cell, hierarchical cell, or design. The POCV coefficient is calculated through MC and is calculated for each of the timing arc, and the value is stored in the timing library or in some external file[11]. The value of the coefficient is calculated without considering the slope and load effect on the cell, hence called the POCV coefficient.

The delay variation sigma = nominal value * POCV coefficient.

The tool picks up the value of the sigma from the POCV enabled timing library. The tool analyzes each timing arc, and the calculations are done statistically to calculate the nominal value of the delay. The variations are also calculated. After that, the delay of the whole path is calculated by statistically summing up all the calculated delays. The result is analyzed to check if the design work at the expected frequency or not. Parametric OCV variation models enable good accuracy, but its limitation
is that it assumes the same cell in different paths have the same variation, the slope and load dependency is neglected.

A new method was originated which work precisely on this limitation of POCV and is named as Liberty Variability Format.

4.5 Liberty Variability Format

In POCV, the variation of the cell is modeled with respect to the single-slope and load only. Liberty Variability Format came out to be, further advancement in this slope and load-based approach. In LVF, the variations are modeled for multiple slopes–load conditions. The variation information is provided in the timing library. The effect of variations is seen on the design by loading them from the library generated. By considering this LVF, based approach, the accuracy of timing closures below 16nm is improved.

LVF is the latest technology that is being used in the industry. LVF utilizes the Monte Carlo simulations. Monte Carlo simulation is the most accurate in addressing timing variation, accounting for each timing arc, rising and falling edges or transitions, conditions for side inputs, and dependency on input slews and output loads. However, considering the number of cells in a library, the number of corners need to be simulated, and the number of slew/load multipliers in each cell, and end up having to perform billions of such simulations for each library. So to overcome this, many techniques which are industries internal are being developed, making this method a robust one.

Coming back to LVF modeling, the sigma values are modeled in the liberty file in the table for the timing arcs.

Example: ocv_sigma_cell_rise(ocvtable_XY) { sigma type: late/early; values ("X1, X2, X3", "Y1, Y2, Y3"); }

![Figure 5. Showing the percentage variation of the LVF derating on varying delays. Delays are calculated at different slope and load.](image)

5. Comparison

Results of the OCV and LVF are compared to show how timing analysis has evolved through time. The analysis is done through the Primetime timing report. The timing report is generated for IN Pin, of a dummy design.

The reports are generated for
- OCV derating.
- LVF derating.

For the design as shown in the figure 5. The analysis is done for the “Hold constraint”. Slack is calculated, i.e., required time subtracted from arrival time. With the analysis, we will see that for OCV, the slack will reduce to a greater amount, adding pessimism. For LVF, slack will be improved, adding very less pessimism as compared to OCV.

![Diagram](image)

Figure 6. Design is considered with a large no. of Buffer present in the path. Hold analysis of the above circuit is done. Launch flop timings is considered on the basis of position of this circuit in the design.

Primetime report with OCV and LVF derating are obtained for the above circuit. Without any derating, the slack is 0.3360, with OCV, it is 0.26178, and with LVF it is 0.32011.

| Startpoint: IN (input port clocked by CLK) | Endpoint: PLL/1 (flip-flop clocked by CLK) | Path Group: CLK |
|-----------------------------------------|------------------------------------------|-----------------|
| Point | Derate | Incr | Path |
| clock CLK (rise edge) | 0.00000 | 0.00000 | |
| clock delay (propagated) | 0.00000 | 0.00000 | |
| input external delay | 0.00000 | 0.00000 | |
| IN (in) | 0.00000 | 0.00000 | f |
| BUFFER | 0.90000 | 0.00002 | 5.00002 | f |
| BUFFER | 0.90000 | 0.00003 | 5.00003 | f |
| BUFFER | 0.90000 | 0.00004 | 5.00004 | f |
| BUFFER | 0.90000 | 0.00005 | 5.00005 | f |
| BUFFER | 0.90000 | 0.00006 | 5.00006 | f |
| BUFFER | 0.90000 | 0.00007 | 5.00007 | f |
| BUFFER | 0.90000 | 0.00008 | 5.00008 | f |
| BUFFER | 0.90000 | 0.00009 | 5.00009 | f |
| BUFFER | 0.90000 | 0.00010 | 5.00010 | f |
| PLL/1 (in) | 0.00000 | 0.00000 | 5.00000 | f |

| Data arrival time | 0.45662 |
| clock SCAN_CLK (rise edge) | 0.00000 | 0.00000 |
| clock source latency | 0.00000 | 0.00000 |
| CLK (in) | 0.00000 | 0.00000 |
| BUFFER | 1.10000 | 0.00003 | 6.00003 | r |
| BUFFER | 1.10000 | 0.00004 | 6.00004 | r |
| BUFFER | 1.10000 | 0.00005 | 6.00005 | r |
| BUFFER | 1.10000 | 0.00006 | 6.00006 | r |
| BUFFER | 1.10000 | 0.00007 | 6.00007 | r |
| BUFFER | 1.10000 | 0.00008 | 6.00008 | r |
| BUFFER | 1.10000 | 0.00009 | 6.00009 | r |
| BUFFER | 1.10000 | 0.00010 | 6.00010 | r |
| PLL/1 (in) | 1.00000 | 0.00000 | 6.00000 | r |

| Clock recon. | 0.00000 | 0.00000 |
| Library hold time 1.00000 | -0.90000 | 0.00000 |
| Data required time | 0.23483 |
| data required time | 0.23483 |
| data arrival time | -0.45662 |

| Slack (MET) | 0.26178 |
| Derate Summary Report |
| total derate : required time | 0.00000 |
| total derate : arrival time | 0.00000 |
| total derate : slack | 0.07452 |
| slack (with derating applied) | 0.3630 |
| slack (with no derating) | 0.30630 |
6. Conclusion

This paper discusses the timing analysis methods along the course of time. Based on the analysis, tool simulations, and industry feedback, the paper can conclude that LVF is the scope for future technologic nodes, which give control to designer to choose between accuracy and execution time. It is going to be mandatory for IP providers to provide the LVF enabled STF file. Also, the time to generate LVF enabled STF can be significantly improved as compared to the classical Monte Carlo simulations. Technical feasibility and analysis with various EDA companies are on-going to further reduce the LVF generation time, keeping the accuracy as close to the classical Monte-Carlo Simulations.

Acknowledgments

The above work is supported by STMicroelectronics, Greater Noida, India. The authors would like to thank the anonymous reviewers whose valuable suggestions improved the overall quality of the paper.

References

[1] A. Agarwal, D. Blaauw, and V. Zolotov 2003 Statistical timing analysis for intra-die process variations with spatial correlations in Proc. ICCAD, pp. 900–907.
[2] S.Nassif 2001 Modeling and forecasting of manufacturing variations (embedded tutorial) in Proc. ASP-DAC pp. 145–149.
[3] R. B. Hitchcock June 1982 Timing Verification and the timing analysis program Proc.of the Design Automation Conference pp. 594-604.
[4] R. Brashear, N. Menezes, C. Oh, L. Pillage, and M. Mercer Mar. 1994 Predicting circuit performance using circuit-level statistical timing analysis,” in Proc. pp. 332–337
[5] C.Forzan, D.Pandini 2009 Statistical static timing analysis: A survey Integration the VLSI journal 42 pp.409-435
[6] D. Blaauw, K. Chopra, A. Srivastava, L.Scheffer April 2008 *Statistical timing analysis: from basic principles to state of the art* IEEE Trans. Computer-Aided Design.

[7] L.Scheffer 2004 *The count of Monte Carlo* in Proc. TAU Int. Workshop on Timing Issues.

[8] I. Nitta, T. Shibiya October 2007 *Statistical static timing analysis technology* Fujitsu Laboratories.

[9] S. Walia April 2009 *Easy-to-adopt, Variation-Aware Timing Analysis for 65nm and below Prime Time Advanced OCV technology*.

[10] P.S. Fuang November 2014 *Implementation study of Path-Based AOCV Model on Pessimism Reduction for 20nm technology* IEEE International Conference on Control System, Computing and Engineering

[11] A. Ray, R. Kheterpal, A. Yadav 2017 *Using Primetime Parametric OCV analysis approach for 28nm FDSOI design signoff* Synopsis User Group.