specification or for conducting inquiries into the legal validity or scope of those patents that are brought to
its attention. By publication of this document, Si2 takes no position with respect to the existence or validity
of any patent rights. A list of any member submitted certificates and exclusions are also available from Si2.

Si2 makes no representation as to the reasonableness or nondiscriminatory nature of the terms and
conditions of the license agreements offered by any such holder(s).

Copyright: This document is subject to protection under Copyright Laws: Copyright (c) 2008 Si2, Inc. All
Rights Reserved Worldwide. Any unauthorized use, reproduction, modification, or distribution of this
document is strictly prohibited. Si2 will make a royalty-free copyright license available upon request.

Trademarks: Where manufacturer or vendor designations claimed as trademarks are used herein, and the
publisher or authors were aware of the trademark claim, such designations have been printed in initial caps
or all caps.

Restricted rights: Use, duplication, or disclosure by the Government is subject to restrictions as set forth
in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor.

Disclaimer: The following description outlines an approach adopted by much of the
industry regarding Statistical Design Techniques. This information is provided only as a
description of the current state of the art, and does not constitute a recommendation or
contribution on the part of Si2 or any Si2 initiative or Member regarding implementation
approaches. This information is not a standard or a specification that has been approved
by Si2 or by any Si2 initiative or Member, including without limitation Si2's Open
Modeling Coalition (the "OMC"), and has not been subjected to a "Call for Patents" by
the OMC. All information in this description is provided "AS IS" without warranty or
representation of any kind including without limitation any warranty or representation of
merchantability, fitness for particular purpose or non-infringement. No information in this
description shall obligate Si2 or any Si2 Member to grant licenses under any patent rights
controlled by Si2 or such Member with respect to any implementation or other use of this
information.

Restricted Rights: Use, duplication, or disclosure by the Government is subject to
restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor.
Document Status

This section reflects the status of this document at the time of its publication. Other documents may supersede this document. Please contact Si2 for a complete list of current Si2 publications produced by the OMC.

This document was produced under the IP Policy of the OMC. Si2 maintains a public list of patents relevant to this document. Contact Si2 for information about this list. Anyone who has actual knowledge of a patent believed to contain Essential Claim(s) with respect to this specification is requested to contact Si2 at legal@si2.org.

Formal comments and error reports on this document should be sent to omctsg@si2.org (archive).

Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>02-December-2008</td>
<td>Initial Version</td>
</tr>
</tbody>
</table>
Si2 Open Modeling Coalition
Statistical Working Group

Antenor de Carvalho       Virage Logic
Eduardo Flores            Nangate
Nagib Hakim               Intel
Jeffrey Hemmett          IBM
Sachin Idgunji            ARM
Kerim Kalafala            IBM
Patrick McGuinness        Freescale Semiconductor
Ayhan Mutlu               Extreme DA
Mahesh Sharma             AMD
Oscar M. Siguenza         Virage Logic
Ashish Srivastava         Intel
Alex Tetelbaum            LSI
# Table of Contents

1  Need For Statistical Techniques In Design Flow................................................................. 6  
   1.1 Definitions and Acronyms ............................................................................................ 7  
   1.2 Case Study, Technology Trends & Limitations In Lithography...................................... 9  
   1.3 Case Study, Statistical STA vs. Conventional STA.......................................................... 11  
   1.4 Case Study, Chip Timing and Optimization Using Statistical Analysis Methods............ 15  
2  Intended Usage Of Statistical Techniques In Design Flow.................................................. 18  
   2.1 Design Flow Description, Concept To Reality................................................................. 19  
   2.2 Initial Intent of Specification.......................................................................................... 21  
3  Specification Details ............................................................................................................ 25  
   3.1 Physical Variation Effects & Their Approximations ......................................................... 25  
   3.2 Mathematical Modeling for SSTA.................................................................................. 26  
   3.3 Timing Propagation......................................................................................................... 27  
Appendix A:  Process Variation Impact On Timing Characteristics......................................... 29  
Appendix B:  Canonical Delay Propagation Example (presentation format)............................ 30  
Appendix C:  Statistical max/min operations explained (presentation format)......................... 43  
References................................................................................................................................. 53
1 Need For Statistical Techniques In Design Flow

This specification concerns itself with statistical techniques needed to ensure the progression of Moore’s law. Lithographic and other process techniques are becoming more and more limited in what can be accomplished from one generation to another.

In previous technology nodes, the uncertainty caused by the equipment could be controlled to the point where simple rules could be defined to allow designers to sign-off their products with good confidence. Second order effects on those nodes could, for the most part, be ignored.

The industry has progressed to the point where some of those second order effects are becoming more and more pronounced. The traditional sign-off, using worst case and best case models to ensure sufficient margin to account for these second order effects, is no longer feasible due to the fact that these effects are now as large and significant as some of the primary issues encountered before. This has led to an apparent loss of scaling when going from 90nm to smaller geometries.

In addition, other sources of variation in design are being affected by environmental concerns as well as large integration of components. This has led to stronger transient currents and thermal gradients that have no unified modeling in current approaches. These sources of variation have led to inaccuracies in the modeling or a proliferation of libraries to handle multiple corner cases.

To ensure that scaling can continue to provide the semiconductor industry and its customers the desired for improvements in performance, better tools and techniques need to be applied so that design specific requirements can be optimized around technology specific effects.

The following figure (Figure1.2: Target Statistical Framework) represents the expected framework being addressed by this specification.
This specification tries to address those issues by ensuring unity of approach across the industry. The intention of this specification is to address the problems and provide the performance enhancement described in the following case studies. The specification is also trying to ensure that a thriving eco-system can be built around the techniques such that rapid adoption of these techniques can happen while allowing companies to inter-operate to provide a complete solution to the industry as a whole.

1.1 Definitions and Acronyms

The following definitions and acronyms are for normative purposes to make communication of statistical terms consistent across this document. They are meant to define what this document means when talking about each term. These definitions may not exactly reflect the “standard” definitions used for these terms.

In addition to these set of definitions and acronyms, the Si2-OMC S-ECSM specification version 1.3, has additional definitions that may be used throughout this document to describe terms related to the S-ECSM specification.

Class – A class is a collection of sets (or sometimes other objects) that can be unambiguously defined by a property that all its members share.

Correlation – A statistical measure referring to the relationship between two or more variables (events, occurrences etc.). It describes the amount of influence one measurement has on the other. Correlated variables describe the increase or decrease of the probability of one variable occurring when the other one occurs.

ECSM – Acronym that stands for “Effective Current Source Model”. A modeling standard that provides non-linear driver and receiver model for a cell level description of a library. The model captures non-linear output waveforms with respect to loading, input state and input ramp time conditions. It also captures input capacitance as a function of input ramp time, output load and pin state conditions.

Environmental Variation – Rate or magnitude of change due to voltage and/or temperature changes.
Temporal Variation – Changes due to aging and related effects that modify the device performance over time.

Foundry Model – Collection of transistor and interconnect models with parameters derived from one or more foundry fabrication processes.

Global Variation – Variation that affects all transistors/interconnects in the same direction, this component of variation is fully correlated. Global variation specifically defines fully correlated variation within one die. Also known as “Systemic Variation” (not to be confused with “Systematic Variation”).

Liberty – Synopsys’ library format (.lib).

Liberty Extensions – Additional constructs incorporated into a Liberty model, beyond the original constructs as defined by the Liberty syntax. The extensions are expressed per the liberty standard for extensions.

Local Variation – Variations that apply to spatially bounded objects or data sets. Local variation is correlated but the magnitude of the correlation varies across location, distance and/or direction.

Model – Abstracted description of a complex entity or process. Usually a mathematical construct that approximates a physical or electrical system.

Order – Degree of a polynomial. The “Order” is defined as the maximum degree of the degrees of all terms in a polynomial.

Parameter – Variable that captures a feature of a model which describes a degree of freedom for a system or function. Usually used in the context of a “Model”. Parameter, in the context of statistical timing, can be a source of variation.

Pin – Connection construct on the interface list of a library cell.

Polynomial – An expression in which one or more variables and constants are combined using only addition, subtraction, multiplication, and non-negative whole number exponents (raising to a power).

Random – Uncertain but bounded effect, bounded within a prescribed confidence level.

Random Variation – The tendency for the estimated magnitude of a parameter to deviate randomly from the true magnitude of that parameter. Random variation is independent of the effects of systematic biases. Random Variation can be correlated or uncorrelated.

Mismatch Parameter: A parameter uncorrelated to any other parameter or any other instance of itself.

Independent Random Variables: Two or more Random Variables that are uncorrelated relative to each other.

S-ECSM – Statistical library extensions based on ECSM.

Sensitivity – Ratio of change in the property being observed (delay, power etc.) relative to the change of a parameter at some point of interest.

Linear Sensitivity – A sensitivity which is linear across its range of applicability.

SPICE – An acronym that stands for “Simulation Program with Integrated Circuit Emphasis”. A general-purpose circuit simulation program.

Statistical – Collection of methods that can be used to summarize or describe a collection of data, in such a way that accounts for randomness, correlation and uncertainty in the observations.

Systematic Variation - Predictable mean shift caused by the properties of the system being observed.

Taylor Series – Representation or approximation of a function as a sum of terms calculated from the values of its derivatives at a single point.\(^1\)

Unit Gaussian Function - Gaussian distribution where the mean is zero and the standard deviation is one.

Variation – An instance of change, or the rate or magnitude of change.\(^a\)
1.2 Case Study, Technology Trends & Limitations In Lithography

Continued process scaling has lead to semiconductor manufacturing processes that are more complex than ever before. Process control precision is struggling to maintain accuracy and each new process generation is now accompanied with new process steps and further restrictive design rules (RDRs) to enhance manufacturability. Variations in process parameters result from effects in a variety of different process steps. These include effects such as those arising from chemical mechanical polishing (CMP), optical proximity effects (OPE), random dopant effect (RDE), line-edge roughness, dose and focus variation and a host of other process induced variations.

CMP is used to produce isolation between the different metal layers and to planarize the oxide surrounding the metal lines. Since polishing rates vary depending on the separation between the metal lines, and also vary between the oxide and metal itself, it produces metal lines with varying thickness, and planarity and process uniformity suffer. These variations are known as erosion and dishing. As copper etches much faster than the surrounding dielectric, the wire ends up being shorter than the oxide. This is known as dishing, and is defined as the vertical distance between the final oxide level and the lowest point in the copper wire. Constraints are set on the processing equipment (including slurries and pads) to limit the amount of dishing in the widest wire expected in a given process. Oxide erosion is another problem - normally in this case CMP is applied to an array of dense lines. The oxide between wires in a dense array tends to be over-polished compared to nearby areas of wider insulators (that is, oxide between sparse features will be thickness than that between dense features).

Patterning of features smaller than the wavelength of light used in optical lithography results in distortions due to the diffraction of light, and is referred to as optical proximity effects (OPE). Corrections, known as optimal proximity corrections (OPC), are made to the mask layout to account for these variations. However, these effects cannot be completely eliminated due to inaccurate correction algorithms and models, and mask constraints. These effects causes an increase in variation of device and interconnect physical parameters such as gate length (or Critical Dimension - CD), gate oxide thickness, channel doping concentration, interconnect thickness, etc. A shift to extreme ultra-violet lithography is expected to improve these effects. However, shorter wavelength lithography technology is too costly and unstable to be used in current technologies.

Optical effects result in variations in polysilicon lines that vary based on their orientation and distance to the neighboring lines. The edge of a polysilicon line is characterized as being dense if the next line is at the minimum possible distance, denso if the next line is at some intermediate distance, and isolated if the next line is further apart. Based on test-chip measurements, the work in [7] found that proximity CD variation is a strong function of both the orientation and the nearby environment. Controlling these variations has become extremely critical in current technologies and has resulted in an explosion in the number of design rules and transistor models. Polysilicon routing in two orthogonal directions may no longer be allowed in certain technologies, so that better control can be achieved in one single direction.

Statistical quantization effects, such as random dopant and line-edge roughness effects, have also grown with scaling of process dimensions. The number of dopant atoms in the channel region of a device decreases as the critical dimension is scaled down. As the number of dopant atoms becomes less, small variation in their number result in a large variation in device performance. Moreover, the actual location of these atoms also plays a role in determining the threshold voltage of a device, further increasing the variability. These variations are true random variations with no correlation across devices and represent one source of random intra-die random variations.

From the perspective of timing analysis, key variations in process manifest themselves as variation in gate-CD, device threshold voltage and interconnect dimensions. Gate CD variation can be categorized as being inter-die (across-die) variation or intra-die (within-die) variation, as shown in Figure1.2.1. Inter-die variations have the same impact across all devices on a die and result from wafer or lot level non-uniformities. On the other hand, intra-die variations impact each device on the die differently. Depending
on the source of variation intra-die variation may be statistically correlated (as shown in Figure 1.2.2) or
independent across devices on the die. Correlated variations result from effects such as stepper non-
uniformity, lens aberration etc. Thus, the correlation is a function of the distance between the devices and it
is important that these correlations are accurately modeled for accurate timing analysis. Statistically
random intra-die variations result from effects such as line-edge roughness etc.

Inter-die variations in threshold voltage can be attributed to wafer-level non-uniformities in gate-oxide
thickness and doping concentrations. Random dopant fluctuations result in a strong component of random
intra-die variation in devices. However, it is important to note that this variation is a strong function of the
device width and it is important that models are available to capture this dependence to analyze designs,
which typically have a wide range of transistor widths. Since random variations statistically average out
across long timing paths, their impact is reduced in scale for circuits as compared to a single device.
However, it is of utmost importance that such variations are considered while designing analog circuits, SRAMs, register files etc.

In this section we have reviewed lithographic variations. However, it is important to understand that variations also result from modeling and environmental uncertainties. Modeling uncertainties are incurred during the design phase due to inaccuracies in various modeling and analysis techniques. These include effects such as false paths, inaccurate parasitics etc. Environmental uncertainties depend on the conditions and workloads under which each sample of the design is operated. These results in variations in power supply voltage and temperature, which impacts timing. Since worst-case modeling and environmental variation can be true for each sample of the design manufactured, these variations are considered as being worst-case.

1.3 Case Study, Statistical STA vs. Conventional STA

Traditional corner analysis based Statistical Timing Analysis (STA) deals with discreet process voltage and temperature (PVT) points when dealing with timing for a given design. The corner conditions have traditionally included three standard PVT corners:

- Worst case timing (WCT)
- Best case timing (BCT)
- Typical case timing (TCT)

Each corner considered process variations, voltage variation and temperature variation. These corners were well defined and for the most part achieved their respective goals. Newer processes had to deal with additional sources of variations for device variations (combinations of Fast/Slow NMOS, Fast/Slow PMOS), metal variations (combinations of Large/Small, metal width, thickness, ILD thickness, via resistance variations), environmental variations (voltage, temperature). This has caused the WCT and BCT corners to include more and more margin to cover these increasing sources of variation.

This has also led to a need to add additional corners and timing runs to take into account issues such as temperature inversion and low voltage corners. This has led to an exponential increase in the runtime while at the same time not being able to handle within die sources of variation that impact critical signals such as clock where skew effects can lead to increases in area or catastrophic failures such as hold time violations causing untestable silicon.

With Statistical STA, on the other hand, various trade-offs can be made. One such trade-off is performance vs yield. In addition realistic handling of statistical variations can be accounted for. This will lead to prevention of loss of performance due to guard-banding as well as new possibilities for optimization and robust design methodologies. Modeling the cell level information and process level information, not as a fixed set of corners, but as a set of corners with some amount of variability around that point allows the tools to understand not just the fixed values but how well they behave under shifting conditions.

The following example shows how Statistical STA compares with conventional STA analysis. The test case uses a 20K instance design. When using conventional STA, the design is timed for worst case setup timing slack. The conventional STA technique analysis can then be done with a few different assumptions. Here is a table of the STA results.
Timing Type Using STA Technique | Timing Slack (ps)
--- | ---
Nominal Timing | 20.64
Worst Case Timing | -1733.82
Worst Case Timing Including Skew Derating (5% OCV) | -2001.73

Table 1: Timing Type Comparison Using Different STA Techniques

Notice that the nominal results are significantly better than the worst case analysis. In addition since the worst case timing analysis does not understand the possible impact that on die process variation has on the clock skew of a typical balanced clock tree (BCT), additional margin needs to be added to deal with the possible skew optimism in the design. Lack of detail variation information for the conventional analysis results in either too much margin or in some cases not enough margin. This leads to iteration through timing closure analysis making the design potentially bigger due to hold time fixes and in some cases making the design un-workable.

In the case of Statistical STA we can have multiple cases to consider for the analysis. The following analysis will be done as part of this comparison: die to die variation, die to die variation with within the die spatially correlated variation, and die to die variation with within the die spatially correlated variation as well as some pure uncorrelated random variation.

Each of these analyses takes into account that the timing is not a fixed term but varies between die as well as within the die which causes timing paths to have a range of possible timings when the complete batch of parts is considered. In this case instead of thinking of a single die and its performance level, we can think of things as multiple die with slightly different performance levels. So in terms of a “population” of die one can think of the timing on any given path as a probability density function such as this next figure.

![Figure 1.3.1: Timing Slack Distribution Probability Density Function (PDF)](image)

One can see that the population follows a distribution where, depending on the performance required, we can have varying amount of “speed yields”. Therefore one can move to the right of the “zero slack point”
or to the left of that point to get either increasing yield with a performance loss or decreasing yield with a performance gain.

Going back to the Statistical STA analysis. The following table shows the results of each corresponding analysis type.

<table>
<thead>
<tr>
<th>Timing Type Using Statistical STA Technique</th>
<th>Average Timing Slack (ps)</th>
<th>Standard Deviation (ps)</th>
<th>Parametric Yield (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Die to Die Only</td>
<td>20.46</td>
<td>221.82</td>
<td>54%</td>
</tr>
<tr>
<td>Die to Die and Die Level Spatially Correlated Variation</td>
<td>-37.43</td>
<td>116</td>
<td>37%</td>
</tr>
<tr>
<td>Die to Die, Die Level Spatially Correlated Variation and Uncorrelated Random Variation</td>
<td>-39.51</td>
<td>107.12</td>
<td>36%</td>
</tr>
</tbody>
</table>

Table 2: Statistical STA Results Comparison Under Different Variation Assumptions

The above conditions show that compared to the original conventional STA analysis, significant yield can still be achieved for the same parts. As the different sources of variation are brought to bear in the analysis we get corresponding decrease of parametric yield but still can achieve reasonable timing for the design. In addition since the statistical STA can provide a distribution for the timing sign-off we can show how changing the performance vs. yield can achieve specific price points for the design. The following graph shows the changing yield distributions for each of the different conditions described so far.

Figure 1.3.2: Statistical STA Results Comparison Under Different Variation Assumptions
The graph also contains the cut-off point used for the conventional STA design. One can see that we can get significant yield (99% or higher) with much better performance by moving to the right of the “zero slack point” for the design. While for the worst case analysis we end up with a single sign-off point that does not allow us to trade-off performance for yield and can therefore take longer to bring to market.

Even though the worst case analysis can be designed to “guard band” the timing for the design sufficiently, there are cases when traditional worst case timing does not provide sufficient margin to catch outliers that can cause sign-off problems.

The following graph shows a slack distribution comparison between the conventional STA analysis and statistical STA analysis.

![Slack Distribution Comparison Conventional STA vs SSTA (3 sigma worst case)](image)

*Figure 1.3.3: Slack Distribution Comparison Conventional STA vs SSTA (3 sigma worst case)*

One can notice that the number of points to the left of the “zero slack” point is quite large for the conventional STA analysis. A normal timing closure optimization would be to work on all the points to the
left of the line to close timing for the design. In the statistical STA case the number of points below the “zero slack” point is much smaller which will lead to less optimization work for timing closure.

The fact that the timing clusters tend to follow the 45 degree line means that the conventional and statistical techniques track well for the design. The pessimism inherent in the conventional case can therefore be reduced.

1.4 Case Study, Chip Timing and Optimization Using Statistical Analysis Methods

A key argument for leveraging the power of static statistical timing analysis (SSTA) focus on reducing the modeled pessimism in emerging and future technologies, whose extremely small dimensions cause even small perturbations to produce relatively large variations on a percentage basis. With the relative variability increasing, there is a concurrent increase in the required guard banding, and also in the number of timing runs required, to safely and adequately cover the space. As the number of runs required has some finite practical limit to obtain reasonable TAT, the guard banding must be increased to ensure working silicon. The problem of course is that this guard banding can lead to performance figures that are only marginally better than those achieved with the latest generation of the previous technology, making it difficult to rationalize the move (while competitors may be migrating).

In this scenario, the primary benefit that SSTA provides is a statistical representation of the variation, making it no longer necessary to aggressively guard band timing results to account for the exposure to variability. By modeling the variability in a statistical manner, SSTA provides a means for low probability scenarios to be selectively filtered out. SSTA also provides valuable information that may be used for more intelligent test definition, e.g. the ability to identify path dominance across the process space. In addition to reducing pessimism via identifying low probability states, SSTA also provides a more intrinsic suite of mathematical tools that can be leveraged to reduce pessimism, such as via Root-Sum-Squaring (or ‘RSS’ing). During RSSing, which may be applied between uncorrelated variables and accounts for the conditional probabilities in their joint distributions, the square root of the sum of the squares of data is used, as opposed to a straight sum, reducing the magnitude of the final product. At the end of the day, these arguments provide a means to rationalize investment in next generation technologies by reducing pessimism outright (e.g. via RSSing), and also providing a means to trim the statistically insignificant tail of the yield curve, improving delivered performance. Essentially, it rightly points out that there is absolutely no difference between a hardware based performance improvement (e.g. as historically provided by reduced dimensions, and often currently augmented with technology twists such as the use of copper interconnects, or leveraging strained silicon), and software based performance improvements via improved modeling. An “X” percent increase in delivered performance is just that, regardless of its source (although it should be noted that software investments often cost magnitudes of order less than hardware breakthroughs).

While maintaining the benefits of migrating to a new technology is indeed a key benefit, there exists an even more fundamental benefit to SSTA: it also produces more robust chip designs, at any technology node. In some respects, this is looking at the exact same benefits from an opposite perspective, but this perspective provides a strong argument for the use of statistical methods even in current technologies. Specifically, block based statistical timing provides full chip coverage (as opposed to path based methods), and it provides coverage over the entire process space. There is no place for fails to hide!

A very simple example of the power of statistical timing will now be provided. This is a real world example that did result in chip fails. An after-the-fact statistical analysis of the data demonstrated that statistical methods would have caught the fail (the statistical methods were not available at the original time of design), and this experience lead to accelerated production adoption of SSTA methods.

The case study involves a technology containing at least 5 metal layers, and was originally analyzed using a corner based timing methodology. In the timing methodology, all metal layer variations were assumed to track perfectly, requiring only two corners to be timed for all layers: all metal fast, and all metal slow. The
problem arose in that metal layer mistrack was just as probable as was perfect tracking: there was no correlation between layers. This can be seen in “Figure 1.4.1: Metal 4/Metal 5 wire resistance mistrack.” below. In this plot, the vertical axis represents the Metal layer 5 resistance, the horizontal axis represents the Metal 4 resistance, and the points in the plot represent actual resistance samples (the colors indicate lots). Note that the data is totally uncorrelated: it is just as likely to move in a mistrack direction (shown by the red arrow) as it is to track. The corners analyzed during timing analysis are shown by the large green dots, but note the significant volume of mistrack data that does not lie near the line connecting these dots (which indicates the perfect tracking direction).

![Figure 1.4.1: Metal 4/Metal 5 wire resistance mistrack.](image)

The fails experienced were a direct result of this mistrack, or more specifically, the lack of modeling this mistrack during timing sign-off. The problem was traced to a set of hold failures experienced when the clock paths were dominated by one metal layer, and the data path was dominated by another, as shown in Figure 2. Under these conditions, the final slack became overly sensitive to both M4 and M5 variations, and any significant mistracking would result in a fail. This can be seen in “Figure 1.4.2: Clock and data paths leading to latch dominated by different metal layers.” below, where the failure instances correspond exactly to the highest level of metal layer mistrack.
Figure 1.4.2: Clock and data paths leading to latch dominated by different metal layers.

Figure 1.4.3: Chip fails track the metal to metal layer mistrack
While part of the issue here may appear to be the assumption that metal layers would track better than they do in reality, this was not a random or uninformed assumption. The problem is that in the corner based analysis used to time the design, if two boundary corners are assumed for each metal layer (fast, slow), for N metal layers, up to 2N corner analyses would be required to cover the entire process. In this case, with at least 6 metal layers, 64 individual timing runs would be required to fully cover the space, and any gaps roughly leave an equally probable exposure in the absence of any mistrack data. Also, note that in practice there may be more than two metal process corners, e.g. there may be (low-C low-R, high-C low-R, low-C high-R, and high-C high R); in this case 4096 timing runs would be required to fully cover 6 layers, and this does not even start to account for accompanying silicon variability.

Contrast this to SSTA, where a single timing run covers the entire process space. Under this approach, a single timing run would produce metal performance distributions for each metal layer. It is then a trivial operation to identify if any corners fail, as all that must be checked is the single worst corner. This is trivially calculated by subtracting from the slack mean the product of sampling sigma (say +/- 3 sigma) with each metal layer sensitivity, such that the result produces the minimum solution. For example, say there are two metal layers, and the SSTA representation for slack is \(5 + 3 \times M1 - 2 \times M2\), i.e. the mean slack is 5 ps, and there is a sensitivity of +3 ps/sigma to M1, and a sensitivity of -2 ps/sigma to M2. Using this SSTA data, the minimum result is easily identified as the case where M1 is sampled at -3 sigma, and M2 is sampled at +3 sigma (for a final slack in the M1(-3), M2(+3) process corner of \(5 + 3(-3) -2(+3) = -10\)). So, with SSTA this fail particular fail condition can be identified in a single timing run (along with any other fail condition in the modeled process space, including silicon), as opposed to up to 64 runs using traditional STA, for coverage of 2 corner metal layers only.

Not only does SSTA provide inherently improved coverage, it also provides valuable information that is not as easily obtained in traditional STA. The core problem here is not so much the physical mistrack, this may be an unavoidable artifact of the process. The problem is that the clock path was heavily routed on one metal layer, while the data path was heavily routed on another. This produces large sensitivities to both of these metal layers. If, instead, both clock and data had been routed on the same layer(s), the clock and data sensitivities would have canceled during the slack calculation, resulting in a small sensitivity to metal mistrack, and much greater design robustness. These types of large sensitivity conditions are easily identifiable in SSTA, as this sensitivity data is one of the direct results produced. If a design shows high sensitivity to some set of metal layers (or e.g. some set of VT families), one easy way to mitigate this is to re-route the design with more layer (or VT) commonality. While it may be possible to heuristically approximate this in corner based timing (e.g. comparing routing of clock/data paths), SSTA will concisely flag only those instances that actually result in problems, and provide concrete numerical data as to what the actual problem parameters are.

2 Intended Usage Of Statistical Techniques In Design Flow

Statistical techniques can and have been used through the design flow for a very long time. Most of these techniques have been used internal to IDM companies to address areas of the design flow that are not easily analyzed or predicted. For example, it is widely know that there are various types of board level noise in any PCB based systems. Noise components can vary from inductive/capacitive coupling of transmission lines (Vxtalk), reflections caused by discontinuities (Vreflection), resistive voltage drops (Vir) and simultaneously switching outputs of chips (Vssn). These sources of noise can be difficult to simulate and predict. One common way of addressing the combination of these noise sources is to take a root mean square approach to combining each noise source to a total potential noise for the transmission line in question. In this case the total noise can be expressed as a root mean square mostly because it is assumed that each source of noise is independent of the other, so a common design technique to do the summation of terms is to express the total noise by this equation:

\[
V_{\text{noise}} = \sqrt{V_{\text{xtalk}}^2 + V_{\text{reflection}}^2 + V_{\text{ssn}}^2 + V_{\text{ir}}^2}
\]

02-December-2008

Copyright © 2008 Si2, Inc
This approach is statistical in the sense that each term is expected to be independent of the other term and described as approximated by an independent Gaussian distribution for the noise. Unfortunately as it is well known the Vir term is not Gaussian but follows a linear expansion based on the length of the line. In this case the more correct way to approximate the noise for the line is the following equation:

\[ V_{\text{noise}} = V_{\text{ir}} + \sqrt{\left( V_{\text{xtalk}}^2 + V_{\text{reflection}}^2 + V_{\text{ssn}}^2 \right)} \]

**Equation 2.2: Combination Of Noise Terms When One Term Does Not Statistically Vary**

This equation better approximates the noise since we know that Vir will always exist and will not be spatially or temporally independent of the others.

So as a can be seen a statistical approach to modeling noise in simple terms can be used to 1) deal with the complexity of the calculation of the terms and 2) in combination with existing techniques to address non-random systemic effects to make the calculation more predictable.

Likewise other areas of designs can use the results of this simple calculations to quickly aid in optimization of the terms, and where ever needed, in modeling effects that have been hereto not modeled well. So each section of the flow can allow for modification of existing techniques to 1) accommodate randomness in the calculation and 2) ensure that systemic effects can be properly modeled and combined with random effects to generate a more accurate system calculation. This specification will address the mathematics required to approximate the combination of terms. Wherever possible the mathematical equations needed will be described as necessary and/or references will be provided for a more in depth look at specific subjects.

Wherever possible effects will be categorized to define how they should be handled to ensure interoperability between tools.

### 2.1 Design Flow Description, Concept To Reality

Having the capability of calculating delay, or any other parameter, using statistical techniques, allows the design flow to take into consideration different aspects of design that could not be fully considered before. One can view the current design flow as a successive approximation of design intent. One starts out with a well defined mathematical model of a state machine and gradually maps this model into successive realization of this design intent in hardware and in some cases software.

This process can be seen as a way to not only map the mathematical construct but also realize the required design performance. This design performance can be defined as a function of desired frequency, area, dynamic power, leakage power, expected yield, cost, etc or a combination of all. The process for realizing those design targets is fraught with unpredictable terms that a probabilistic statistical model could help bound.

This section describes opportunities for innovation to bring in statistical techniques in aspect of design that would help the overall design flow achieve closure faster and more predictably, while taking advantage of the latest process nodes. The discussion will center around opportunities for timing but can be extended to other areas of design performance (i.e. power, leakage, frequency, manufacturability, etc).

Design goes through a serious of steps from concept inception to actual product. Each design step will have some well defined parameters as well as some not so well defined parameters or implementation choices that need to be considered to implement the design at the required cost point. We will cover the following major steps during this description: high level behavioral design and partitioning, register transfer level design synthesis, physical design placement and optimization, physical design routing and optimization and final physical design sign-off. Initially the design would have some idea of what its timing metrics need to be to fulfill the basic design intent.

During the high level behavioral design and partitioning, one typically takes a look at meeting the timing targets for the design by deciding what algorithms will be fast enough for implementation and whether a hardware or software implementation would be needed. If a hardware implementation is required then the
designer would proceed to approximate technology selection for the design. During this phase of the
design there is a lot of uncertainty on what the target technology would be for the design. Cost
considerations would be driven as a trade-off of partitioning the design across multiple chips or a single
high integration SoC design. Since performance is always a consideration the designer would proceed to
obtain aggregate timing data for the target technology. Typically the designer will look to see what the
number of stages of logic is in the design and will use either an “average” number or a “worst” case
number for technology node selection. Either approach taken will lead to either too much optimism for
some some parts of the design (i.e. sections of the design with large number of stages which tend to average
out) or too much pessimism for others (i.e. datapath intensive sections that tend to have shorter delays).
The aggregate numbers will therefore lead to either a too aggressive, in timing, low cost technology
selection, or an unnecessarily expensive fast technology selection. In either case the design cannot deal
well with ensuring the most optimum technology usage.

With a statistical timing infrastructure, this high level design and partitioning can take advantage of the fact
that a lot of the uncertainties in the design can be modeled as a “variation” relative to the “mean” value for
those average numbers. This will lead to a better calculation of the target performance and would
differentiate significantly between paths in the design that have large number of logic stages (where the
RMS'ing of the variations will lead to smaller than worst case numbers) and paths of the design that have a
low number of logic stages (where the RMS'ing of the variations will not decrease the total delay as
significantly as in the large number of stages case).

For example, let's assume at this level of design there are two types of critical paths that are involved. Path
type 1 has no more than 4 stages of logic and tends to short interconnects. In this case assuming that the
average delay is TD1 and the average standard deviation is dTD1. Path type 2 has no more than 50 stages
of logic and tends to average interconnects. The average delay and average standard deviation for this path
is TD2 and dTD2 respectively.

For Path type 1, we can calculate the expected average cycle time two ways:
Average Case Conventional:
\[ T_{cycle1ac} = 4 \times TD1 \]
Worst Case Conventional:
\[ T_{cycle1wc} = 4 \times TD1 + 4 \times dTD1 \]
Statistical Worst Case:
\[ T_{cycle1sc} = 4 \times TD1 + \sqrt{4 \times dTD1^2} \\
= 4 \times TD1 + \sqrt{4} \times dTD1 \]
The difference between these two paths and therefore the decrease in pessimism using the statistical
technique can be compared between the average and worst case approach:
\[ T_{cycle1sc} - T_{cycle1wc} = (\sqrt{4} - 4) \times dTD1 = -2 \times dTD1 \]
\[ T_{cycle1sc} - T_{cycle1ac} = \sqrt{4} \times dTD1 = +2 \times dTD1 \]
So this is a nice compromise between the worst case and best case for this path.

For Path type 2 the following calculation would reflect its comparison relative to statistical timing:
Average Case Conventional:
\[ T_{cycle2ac} = 50 \times TD2 \]
Worst Case Conventional:
\[ T_{cycle2wc} = 50 \times TD2 + 50 \times dTD2 \]
Statistical Worst Case:
$$T_{cycle2sc} = 50 \times TD^2 + \sqrt{50 \times dTD^2}$$

$$= 50 \times TD^2 + \sqrt{50} \times dTD$$

The difference between these two paths and therefore the decrease in pessimism using the statistical technique can be compared between the average and worst case approach:

$$T_{cycle2sc} - T_{cycle2wc} = (\sqrt{50} - 50) \times dTD = -42.9 \times dTD$$

$$T_{cycle2sc} - T_{cycle2ac} = \sqrt{50} \times dTD = +7.1 \times dTD$$

We can see that for this path the pessimism of using the worst case is too much and that the statistical result is much closer to the average case. Therefore the technology selection will be easier to converge since both types of paths can be handled correctly which will lead to less iterations in the design for the further steps. In addition since the deviation of delays can be made to track the trend at the specific technology node, differences in variation between nodes can be taken into account as part of the design partitioning and technology selection.

Since the analysis is happening at a very high level, there is always the question of whether the independence of terms applies when combining terms. In this case the person doing the analysis can potentially give some more margin to the calculation by adding a correlated term to the timing variation as well as a purely independent term. For example, the delay equation per stage could be model as follows:

$$TD = TDn + dTDn + dTD$$

Where dTDn, represents the correlated variation across the delay and dTDn, represents the independent variation for the path. In this case we could make an assumption of what percentage of the total variation is correlated or not, in which case we can worst case the correlated factor by assuming 100% correlation for that section and only root mean squaring the terms that are random and uncorrelated. In this case assuming that K denotes the ratio of the correlated vs. uncorrelated data. We can write the individual path delay as:

$$TD = (TDn + K \times dTDn) + (1-K) \times dTDn$$

Assuming we have “m” equal stages of delay the total delay can be expressed as:

$$TDm = m \times (TDn + K \times dTDn) + (1-K) \times dTDn \times \sqrt{m}$$

Using the previous 50 and 4 stage paths described before, setting K = 0.5 and comparing to the worst case path calculation the amount of gain in timing relative to the worst case can be derived. First let's review the delay calculation using the described K value for each path under review.

$$T_{cycle1sc \_2} = 4 \times (TD1 + 0.5 \times dTD1) + 0.5 \times dTD1 \times \sqrt{4}$$

$$T_{cycle1sc \_2} = 4 \times TD1 + 3 \times dTD1$$

$$T_{cycle2sc \_2} = 50 \times (TD1 + 0.5 \times dTD1) + 0.5 \times dTD1 \times \sqrt{50}$$

$$T_{cycle2sc \_2} = 50 \times TD1 + 28.54 \times dTD1$$

Compared to the worst case calculation, the amount of recovered timing for each path time is now reduced to the following values:

$$T_{cycle1sc \_2} - T_{cycle1wc} = -dTD1$$

$$T_{cycle2sc \_2} - T_{cycle2wc} = 21.46 \times dTD1$$

Depending on the amount of margin required for design sign-off/technology selection, the correlation factor can then be varied between 0 and 1 to achieve sufficient design margin.

### 2.2 Initial Intent of Specification

The initial specification is meant to cover the timing sign-off part of the design flow. It is intended to provide sufficiently accurate timing sign-off to allow the statistical technique to replace the traditional corner based sign-off. Even though the techniques themselves will be sufficiently accurate to achieve this
the end user must use appropriate caution when signing-off their design since the data provided for their respective process may not be accurate enough to predict the process ranges and variations.

It is also the initial goal of the specification to deal with the providing analysis support for predicting speed grading yield curve as part of the sign-off process. This would allow chip providers to ascertain speed based yield curves for speed binning of parts. The current specification will not cover the test aspect of speed binning for designs.

### 2.2.1 Timing Sign-off

Timing sign-off for design is a make or break process for designers. The tendency has always been to use new standards to supplement the old tried and true methods for design sign-off. Early in the evolution of static timing analysis, annotated simulation analysis was the golden reference for timing sign-off. At the time simulation based sign-off was becoming a burden for designers because of the increasing size of design and the need to simulate more and more vectors to ensure full coverage of the design space. Simulator acceleration as well as other techniques we brought to bear, to try addressing the necessary capacity issue. At the time static timing analysis became a new technique to allow for two improvements. First capacity to deal with larger designs and second better path coverage by concentrating on the propagation of delays across paths by analyzing the functionality and not relying on vectors to drive the propagation of terms.

Early adopters of STA were reluctant to rely completely on the new technique to sign-off the design so it was typical to rely on both simulation and STA based sign-off. Simulation was used to ensure that the design was not fully broken for areas were STA was not as good at covering (i.e. asynchronous paths, analog sections, etc) and STA to cover the bulk of synchronous conventional design. Over the years the industry has moved to strong reliance on STA with more functional verification of the design paths.

As a matter of course it is the expectation that Statistical STA will see a similar sign-off process by early adopters. It is the expectation and recommendation that until the infrastructure for designers matures, a mixed sign-off flow be used where conventional corner based STA is used to take into account predictable behavior such as IR drop analysis, crosstalk analysis and specific environmental conditions (i.e. temperature inversion effects and low voltage operation), while using SSTA to provide better coverage of the variation design space for the design. Over time it is the expectation that SSTA based analysis will be sufficiently mature to be able to eliminate corner based STA analysis completely.

The goal for this section is to define design flows using statistical techniques in conjunction with traditional corner based sign-off to minimize the following concerns around statistical or conventional based sign-off:

- Reduction of pessimism (aka guardbanding)
- Better process space coverage
- Linearity and separability of effects
- Infrastructure robustness, for example
  - Characterization accuracy
- Mathematical approximations in design tools
- Process modeling inaccuracies
- Etc.

Let’s explore possible design flows that take these topics into consideration and deal with the expected maturation of statistical static timing analysis.

Here is an outline of the conventional design flow currently used:
The flow is growing in the number of PVT points and needs a large margin to be able to gain good confidence in sign-off for designs. The problem becomes that the margin may become so large that it can make the design difficult to close timing on, while at the same time not covering the margin cases well enough. In this case most designers get pushed to minimize margin and maximize the number of PVT corners considered to minimize the chance for failure.

The following design flow adopts early usage of statistical analysis to supplement the conventional flow to cover the design space with smaller margins to address the need to have quick convergence in timing. This is usually a significant step relative to the current design flow as it involves a completely new methodology in timing sign-off.
This step buys you increased coverage while helping you to reduce the amount of margin while ensuring quality sign-off. As the flow becomes a much more trusted flow and gains confidence within the design community we can reduce the amount of analysis that is done using conventional methods and proceed to a faster and still robust flow. In this case we would be moving to Design Flow#2.
In this case the number of PVT corners can be reduced based on historical understanding of where the worst case conditions tend to happen and the conventional checks become more of a sanity check for the statistical timing sign-off.

Eventually, as the infrastructure, tools, methods and process control metrology matures designs will use the following, purely statistical flow, to sign-off designs.

Corner based checks may still exist in this design flow, but will be exclusively used for sanity checks or for double checking setups.

3 Specification Details

The following two sections cover the physical effects as well as the mathematical constructs that are commonly used for statistical timing analysis. Although this is not an exhaustive list, the section gives a sense of what the effects are and how they are modeled to help implement the timing sign-off.

3.1 Physical Variation Effects & Their Approximations

The variation of timing characteristics (delay or slew) under variations in process parameters, depends on (i) the distribution of the process parameters themselves, and (ii) the dependence of the timing characteristic on the process parameters.

The variations in the process parameters are generally classified as inter-die and intra-die variations. Inter-die, or across-die variation, includes lot-to-lot, wafer-to-wafer and die-to-die variation, since all these variations manifest themselves as a shift in the process parameters that does not vary across the die. Thus, this variation impacts all devices on the die similarly, and is characterized by a given standard deviation of the process parameters. In addition, some of the process parameters used to model delay may have correlations that need to be characterized for accurate timing analysis. Such correlations can result from process steps that impact multiple parameters. As an example, variations that result in an increase in wire thickness result in an increase in wire capacitance and a decrease in wire resistance.

Intra-die variations, on the other hand, vary across the die. These variations can result from effects such as non-uniformity in photo-resist coating, etching concentration, dose variations, random dopant fluctuations (RDF) etc. Many of the variation such as dose variations sources vary slowly across the locations on the die, and thus the variations of two devices close together in layout is highly correlated. However, variations such as RDF and line edge roughness are statistically independent even for devices next to each other. Thus intra-die variations are further classified as being either spatially correlated or random. Random variations are characterized by a standard deviation. However, the standard deviation may have a dependence on the nominal device parameters themselves and these models are required to estimate the random variations.
across transistors with different widths, lengths or channel doping [1]. As an example, the standard deviation of the variation in threshold voltage ($\sigma_{vt}$) due to random dopant fluctuations is generally modeled with a dependency of the form

$$\sigma_{vt} \propto \frac{1}{\sqrt{WL}}$$

*Equation 3.1.1: Threshold Variation As A Function Of Transistor Width & Length*

where $W$ and $L$ represent the nominal width and length of the device, respectively. Since the random variables for different transistors are independent, the timing characteristic of a timing-arc are generally characterized as a single term representing its sensitivity to random variations in a specific process parameter. Note that if multiple timing-arcs share some of the transistors then the timing characteristics of these timing-arcs will have correlated random variations.

Spatially correlated variations, on the other hand, are modeled using a standard deviation and the correlation function that is generally a distance-based function. However, the correlation coefficient may have directional dependencies as well. Moreover, a straightforward model to capture spatially correlated variations during timing analysis requires a number of random variables that is of the order of the number of transistors in the circuit. However, spatially correlated variations are very strongly correlated at distances that are greater than the dimensions of a typical transistor. Therefore, simplified models (that require significantly smaller number of random variables) for spatial correlations have been developed. These include models such as the rectangular grid [2] and quad-tree [3] models. Both these models are based on dividing the die area into a number of squares and using a single random variable for each of the squares. The process parameter variation is considered to be perfectly correlated within each square on the grid. However, both these models are discontinuous at the grid boundaries. Recently, continuous models based on Karhunen-Loeve [4] expansion have been proposed and are argued to provide better accuracy.

Now, we can express the value of a process parameter (represented by the random variable $P$) as

$$P = P_{nom} + \Delta P_{inter} + \Delta P_{intra}$$

*Equation 3.1.2: Process Variation Canonical Model*

where $P_{nom}$ is the nominal value of the process parameter, $\Delta P_{inter}$ and $\Delta P_{intra}$ are the random variables representing the inter- and intra-die variation in the parameter, respectively. The intra-die variation can be further broken down as

$$\Delta P_{intra} = \Delta P_{spatial} + \Delta P_{random}$$

*Equation 3.1.3: Intra-Die Components Of Process Variation*

where $\Delta P_{spatial}$ and $\Delta P_{random}$ represent the spatially correlated and the random components of the intra-die variation, respectively. Note that modeling variations in process parameters using standard deviations and correlation coefficients alone is based on the implicit assumption of their distribution being Gaussian. Since, process parameters distributions are limited to a section of the real axis, a Gaussian distribution is obviously an approximation to the actual distribution. However, recent investigations into distributions of process parameters have shown that this assumption is generally true, though not universally accurate. Variations in via resistance are known to have non-Gaussian distributions.

### 3.2 Mathematical Modeling for SSTA

The basic mathematical construct used for SSTA consists of a mean (nominal) value, combined with a summation of parameter terms (the parameters are the sources of variation). Each parameter term includes the per-sigma linear sensitivity of the timing quantity of interest to the current parameter, multiplied by a random variable representing the current state of that parameter. Note that this requires all of the parameter terms to be mean-centered and normalized, allowing for simple combination of unlike parameters on a per-sigma basis. This model allows efficient modeling of both nominal timing data, as well as inclusion of
perturbation effects on the timing data from all modeled sources of variation as these parameters deviate from their nominal values. For m parameters, there would be m-1 global sources of variation, and one parameter representing all independently random sources. An example of such a model is shown below (this model is referred to as the canonical form).

$$\tau = \mu + \sum_{i=1}^{m-1} (S_i \cdot N_i) + R \cdot N_m$$

Equation 3.2.1: Canonical Statistical Delay Equation

Where:

- $\tau$ - The timing quantity of interest (e.g. delay, arrival time; measured in units of time), which can be sampled at different points for $N_i/N_m$ (e.g. 3 sigma)
- $\mu$ - The nominal timing value (all parameters at nominal on their respective distributions, measured in units of time)
- $S_i$ - The parameter sensitivities (mean centered and normalized, measured in units of time/sigma)
- $N_i/N_m$ - Unit Gaussian function
- $R$ - Independently random (e.g. mismatch) parameter sensitivity; note that this may be a composite of multiple, independently random variables.

The benefits of using this canonical form include:

- Relative efficiency of manipulation and propagation
- Allowance for trivial combination of unlike sources of variation
- Allowance for the implicit inclusion of correlation as a result of path convergence during block based timing propagation
- The first order parameter or spatial correlations are also accommodated explicitly via standard Principal Component Analysis (PCA) modeling.

The drawbacks to this approach include:

- Assumption of linearity
- Assumption of Gaussian distributions
- Assumption of separability between the sources of variation.

The first two drawbacks may be mitigated by the fact that this relation may be extended to include modeling of nonlinear, non-Gaussian effects (at some runtime and memory cost).

The basis for the above canonical form is a truncated Taylor series, as shown in Appendix A.

3.3 Timing Propagation

The canonical form may be propagated in a fashion almost identical to that used for deterministic timing quantities. In block based propagation, this consists of two stages, forward propagation for the calculation of arrival times (ATs), and backward propagation for the calculation of required arrival times (RATs). The forward propagation process consists of a sequence of addition and max (late mode timing) and min (early mode timing) operations.

For the sake of simplicity, the following will be limited to late mode forward propagation. The mechanics are similar during early mode and backward propagation, with the appropriate replacement of addition/subtraction and max/min as required. It should be noted that the same approach works equally
well for gates and wires (interconnect). For an example, see Appendix B Canonical Delay Propagation Example. Also, reference [8] provides a more detailed explanation of this approach. These sources are only meant as a high level introduction to the dominant approach, and it is recommended that you consult your software vendor to learn the specific techniques they apply.

During statistical timing, the addition and subtraction operations differ only in the handling of the independently random term. The sensitivities for like global sources of variation are summed in a standard fashion, however the independently random terms are combined with an RSS operation (square root of the sum of the squares).

Using the approach described above, AT sensitivities at gates are summed with canonical delay data provided by the delay models. Once the summations on all arcs are complete, a statistical max operation is performed using data from all incident timing arcs. A statistical max differs from a standard deterministic max in that the results of a statistical max are a linear combination of potentially all of the input canonical forms. This differs from deterministic max operation, which is essentially ‘winner-take-all’ (only timing data from a single arc propagates). The weighting factors used during the linear combination of terms in the statistical max operation are called ‘tightness probabilities’, and they reflect the probability with which the current segment will dominate (produce the max output arrival time) over the entire process space. In some instances it is possible for a given timing arc to dominate the max operation.

The statistical max operation may be performed in one of several manners. The most efficient approach is to leverage the so-called “Clark equations” [9], which allows for analytic solution of the statistical quantities (for Gaussian distributions only). Note that this approach is limited to 2-way operations, and so when the number of incident edges is greater than two a series of 2-way operations must be performed. The order of the operations can affect the results, and so this is not recommended when there is a very high number of incident edges containing a wide range of means and deviations. Other approaches include convolution (which is also limited to 2 way operations), and Monte Carlo approaches (which not suffer from the 2 way limitation, but being MC based the accuracy of the result is a function of the number of samples taken).

### 3.4 Available and Pending Standards

The state of the industry at the time this document is being published is one where no clear standard has come to the forefront of market adoption. There are two methods to encapsulate the static timing data for design components. One encapsulates the data in traditional static data formats such as liberty, while the other encompasses dynamic APIs such as the IEEE 1481 standard. Both can be extended for statistical data. Plans exist to develop an IEEE 1481 statistical standard, and at present, two published static format statistical standards exist. The static standards are:

- **Si2 OMC standard:** S-ECSM
- **Liberty L-tab standard:** CCS-Var

In addition there are a number of proprietary formats companies operating in this space have defined to allow for statistical analysis. There also exist a number of ad-hoc techniques to allow for statistical analysis in the absence of explicit statistical data. For example, it is possible to start with base corner sets and assert a statistical variation term. It is also possible to use data from multiple Process / Voltage / Temperature conditions to derive the necessary statistical information.

The expectation is that both static and dynamic standards will evolve as the technology matures, and the hope is that in time a single statistical standard will emerge in both categories. In the meantime, work with your timing vendor to determine which approach will work best in your flow.
Appendix A: Process Variation Impact On Timing Characteristics

Let us now consider the impact of process variations on the timing characteristics of a standard cell. Delay and slew rates associated with a timing-arc have a strong non-linear dependence on a number of process parameters. However, variations in process parameters are generally small as compared to their nominal value. Hence, the impact of variations can be captured by using a first-order Taylor's expansion around the nominal value of the process parameters, which can be expressed as

\[ d \approx d_{nom} + \sum \left( \frac{\partial d}{\partial P_i} \right) \Delta P_i = d_{nom} + c^T \Delta P \]

Equation 3.3.1: Canonical Statistical Delay Equation, First Order Taylor Expansion

where \( d \) is the timing characteristic with a nominal value \( d_{nom} \) and \( \Delta P \) represents the vector of variations in the process parameters, and \( c \) represents the vector of sensitivities of \( d \) to the process parameters evaluated at their nominal values. Note that the \( \Delta P \) vector has different components for random and correlated variations in the same process parameters, since the gate-level sensitivities will differ for the two components. Also, using first-order Taylor's expansion leads to a certain error that will increase with increase of process parameters. This error may lead to underestimation of variations and should be taken into account. There is also a question if process sensitivities stay the same for different corners.

If the process parameters have a multi-normal distribution, then the timing property approximated as a Taylor's expansion has a normal distribution. However, with increasing process variations the assumption regarding small variations may lead to inaccuracy and some of the statistical timing analysis techniques [5] currently support second-order dependencies. The timing characteristic model then becomes a second-order Taylor's expansion, which can be expressed as

\[ d \approx d_{nom} + c^T \Delta P + \Delta P^T A \Delta P \]

Equation 3.3.2: Canonical Statistical Delay Equation, Second Order Taylor Expansion

where the matrix \( A \) captures the second-order dependencies. Note that even though this expansion is more accurate it is not clear if it can be effectively used during variation propagation through paths.

Timing models for gates are generally based on table lookup or analytical equations. In a deterministic scenario, lookup table based gate delay models are generally used where transition time at the input and the output loading are used as indices to find the delay. The delay values for intermediate transition times and output loads are obtained using linear interpolation. In a statistical scenario, in addition to the nominal value of the timing characteristic we need to characterize its sensitivity to various process parameters (assuming a first-order dependence). For the case of spatially correlated data this requires an additional simulation for each of the process parameters.

However, for the case of random-variations we will need \( n \) simulations for each process parameter, in a \( n \)-gate cell. The combined sensitivity to random variation is then expressed as

\[ \Delta d_{i_{rand}} = \sqrt{s_1^2 + s_2^2 + \cdots + s_n^2} \Delta P_{i_{rand}} \]

Equation 3.3.3: Combined Sensitivity To Random Variation

where \( \Delta d_{i_{rand}} \) represents the variation in the timing characteristic, \( S_i's \) represent the sensitivity of \( d \) to random variations in \( P_i \), for each of the \( n \) transistors. However, the number of required simulations result in a significant increase in the time required to characterize a standard-cell library. Hence, simplifying approximations are generally made. Reference [6], proposes a RSM based technique to characterize the standard cells for random variations.
Appendix B: Canonical Delay Propagation
Example (presentation format)

Statistical Timing Propagation

- Simple 2 gate NAND2 example...
  - Keep it simple...
    - Assume late mode worst case propagation (Max AT)
    - Consider single transition (R or F)
Sources of variation

- Define 2 correlated, 1 uncorrelated variable
  - C1, C2: Correlated sources of variation
  - R1: Independently random data

- What they represent is moot in this example
  - Tox, Leff, NPskew, VT shift, or non-physical PCA
    - For ind. rand... dopant fluctuations, line edge roughness, etc.
  - Example displays mathematical development
    - Independent of parameter application

- Assumptions in base model
  - Independent Parameters
  - Linear response
  - Normal Gaussian

Standard canonical form

- $X = \text{Mean} + \text{Sens}_1 \times C_1 + \text{Sens}_2 \times C_2 + \text{Sens}_3 \times R_1$
  - "X" can be any timing quantity... all use same model
    - Gate or wire delay
    - Path delay
    - Arrival Time
    - Required Arrival Time
    - Slack
    - Slew
    - Guard Time (e.g. setup or hold constraint, pulse width, etc.)
    - Assertion
    - Adjust

- Accuracy increases with # canonically modeled quantities
  - To the extent the parameters have non-zero sensitivities

- Deterministic data can be modeled canonically
  - Mean only, all zero sensitivities
Standard canonical form

- $X = \textbf{Mean} + \textbf{Sens1} \times C1 + \textbf{Sens2} \times C2$
  - "Mean" is nominal solution, with no variation (ideal)
    - Zero sigma location in Joint Probability Distribution Function
      - Have highest probability near the mean, most likely values

---

Standard canonical form

- $X = \textbf{Mean} + \textbf{Sens1} \times C1 + \textbf{Sens2} \times C2$
  - "Sens" are parameter sensitivities (time / unit variation)
    - Defined in a normalized fashion... "per sigma"
    - Provides commonality for propagation
      - As opposed to e.g. Sens1 being "per mV" and Sens2 "per uM"
2D Example...
- Say $X = \text{slack}$
- $C1$ sensitivity
  - Slope of lines $BE, CD$
- $C2$ sensitivity
  - Slope of lines $CB, DE$
- Mean slack
  - Where vertical axis pierces plane $\bigcirc$, at center of ellipse
  - Ellipse is intersection of $C1, C2$ JPDF with plane

---

Propagation example

- Initially assume zero uncorrelated variability
  - Keep it simple... we will re-introduce it later
- Assume zero wire delay
  - Treated similar to gate delay
    - Provide example later
- Example useful for describing mechanics of operations
  - More pragmatic than theoretical
  - Note tight parallels to traditional deterministic timing
AT Boundary Assertions

- AT IN_1: \(5.0 + 0.1\times C1 + 0.1\times C2\)
- AT IN_2: \(1.0 - 0.5\times C1 + 1.0\times C2\)
- AT IN_3: \(1.2 + 0.1\times C1 - 0.9\times C2\)

Level 1 solve... NAND1 Delays

- Delay A-Y
  - \(4.0 + 1.5\times C1 + 0.0\times C2\)
  - Note delay A-Y is insensitive to parameter C2
- Delay B-Y
  - \(3.9 + 0.8\times C1 + 1.8\times C2\)
Level 1 solve... NAND1 Max inputs

- AT-A + Delay A-Y
  - AT A: $1.0 + 1.5 \times C1 + 0.0 \times C2$
  - Delay A-Y: $4.0 - 0.5 \times C1 + 1.0 \times C2$
  - Sum: $5.0 + 1.0 \times C1 + 1.0 \times C2$

- AT-B + Delay B-Y
  - Sum: $5.1 + 0.9 \times C1 + 0.9 \times C2$
    - (By a similar operation)

Level 1 solve... NAND1 Max Op

- MAX(AT-A + Delay A-Y, AT-B + Delay B-Y)
  - By Clark Equations, Monte Carlo, Convolution, etc.
    - Find tightness probabilities
      - Probability a given distribution will provide max result
      - See Appendix C, Clark Reference for more details
    - $T(A-Y) = 0.2398$
    - $T(B-Y) = 0.7602$ ( = $1.0 - T(A-Y)$ )
Level 1 solve... NAND1 Max Op

- MAX(AT-A + Delay A-Y, AT-B + Delay B-Y)
  - Result is a linear combination of inputs
    - Weighted by tightness probabilities
      - This is a bit of an oversimplification...
        - But provides a decent general overview
        - Refer to references for more details
  - Max = T(A-Y) * (AT-A + Delay A-Y)
    + T(B-Y) * (AT-B + Delay B-Y)

= 5.08 + 0.924 C1 + 0.924 C2

This is AT-Y on nand1

Aside: Compare Max to Deterministic

- Say A = 9, B = 10
  - B > A
Aside: Compare Max to Deterministic

- Say A = 9, B = 10
  - B > A

- 10 is always > 9, B dominates
  - In whatever process corner timing is occurring in
    - A may dominate in another process corner, who knows?

Aside: Compare Max to Deterministic

- In prior stat example, A and B are distributions
  - B > A 76.02 % of the time
  - A > B 23.98 % of the time

  - This provides full process space coverage!
Back to example: Level 2 solve

- **AT IN_1: 5.0 + 0.1*C1 + 0.1*C2**

- Repeat addition of AT/delays, max op
  - D(A-Y) = 3.80 + 1.500*C1 + 1.000*C2
  - D(B-Y) = 4.02 + 0.676*C1 + 0.176*C2

```
5.08 + 0.924 C1 + 0.924 C2
```

---

Back to example: Level 2 solve

- Say B totally dominates Max op
  - T(A) = 0.0
  - T(B) = 1.0
    - Not an infrequent occurrence
  - **AT(NAND2-Y) = 9.1 + 1.6 C1 + 1.1 C2**

```
5.08 + 0.924 C1 + 0.924 C2
```
All quantities calc’d in similar manner

- **Slacks**
  - RAT slacks (simply AT-RAT early, RAT-AT late)
  - Test slacks
    - Calc’d same as RAT slacks, w/ inclusion of guard times

- **Path Delays...**
  - Sum input AT, delays along desired path

- **Path Slack**
  - Subtract path delay from RAT (late mode)

- **Note path data will not span process space**
  - May choose non-dominant path
    - Can give misleading perspective

---

**Example Slack Calculation**

- **RAT (OUT_1) = 7.2 + 1.7 C1 + 0.5 C2**

- **Late mode... RAT-AT**
  
  \[
  9.1 + 1.6 C1 + 1.1 C2 \\
  -7.2 + 1.7 C1 + 0.5 C2 \\
  1.9 - 0.1 C1 + 0.6 C2 = \text{design slack}
  \]

---

*Diagram of NAND2 gate with timing delays.*
Do I pass timing?

- SLK = 1.9 - 0.1 C1 + 0.6 C2
  - Mean = 1.9 (units of time)
  - Slope C1 = -0.1 time/σ
  - Slope C2 = +0.6 time/σ

- Ignoring parameter independence
  - Worst corner projection
    - 3-sigma
    - Similar to a corner based approach
  - 1.9 - 3⁺(0.1+0.6)
  - = -0.2

- Fail timing under this corner projection
  - Assuming 0.0 cutoff
  - Will pass corner at higher sigma (lower yield)
Do I pass timing?

- SLK = 1.9 - 0.1 C1 + 0.6 C2
- Assuming independence
  - We can RSS results
  - \(1.9 - 3 \times (0.1^2 + 0.6^2)^{0.5}\)
  - = +0.075
- Reduces pessimism
  - Pass when assuming parameter independence
  - Leverage conditional probabilities

- In any case, note portion of process space where fails occurs can be identified

With non-zero uncorrelated term

- AT-A + Delay A-Y
  - AT A: \(1.0 + 1.5 \times C1 + 3.0 \times R1\)
  - Delay A-Y: \(4.0 - 0.5 \times C1 + 4.0 \times R1\)
  - Sum: \(5.0 + 1.0 \times C1 + 5.0 \times R1\)

- Uncorrelated data can be RSSed during propagation
  - RSS: Square root of sum of squares
Extending to Interconnect

- **AT (NAND1/Y):** \(5.0 + 1.0 \cdot C1 + 5.0 \cdot R1\)
- For Interconnect - Define 1 correlated, 1 uncorrelated variable (for simplicity)
  - M1: Correlated sources of variation
  - R2: Independently random data

![NAND gate diagram]

Extending to Interconnect

- **AT-A + Delay A-Y**
  - AT A: \(1.0 + 1.5 \cdot C1 + 3.0 \cdot R1\)
  - Delay A-Y: \(4.0 - 0.5 \cdot C1 + 4.0 \cdot R1\)
  - Sum: \(5.0 + 1.0 \cdot C1 + 5.0 \cdot R1\)

- Delay A-Y + Interconnect Y-B (next stage)
  - AT Y: \(5.0 + 1.0 \cdot C1 + 5.0 \cdot R1\)
  - Delay Y-B: \(0.2 + 0.8 \cdot M1 + 0.2 \cdot R2\)
  - Sum: \(5.2 + 1.0 \cdot C1 + 0.8 \cdot M1 + 5.0 \cdot R1 + 0.2 \cdot R2\)

- Uncorrelated data RSSEs
  - Square root of sum of squares
Appendix C: Statistical max/min operations explained (presentation format)

Deterministic Timing

- Simple NAND3 example
  - Want AT at output Z
Deterministic Timing

"Worst AT" (late mode) is maximum arc arrival time:

\[ AT_Z = \text{MAX} \ (AT_A + d_{AZ}, \ AT_B + d_{BZ}, \ AT_C + d_{CZ}) \]

Deterministic Timing

\[ AT_Z = \text{MAX} \ (AT_A + d_{AZ}, \ AT_B + d_{BZ}, \ AT_C + d_{CZ}) \]
\[ = \text{MAX} \ (9, \ 8.5, \ 8) \]
\[ = 9 \]
Statistical Timing

\[ AT_Z = \text{MAX} \left( AT_A + d_{AZ}, \ AT_B + d_{BZ}, \ AT_C + d_{CZ} \right) \]

- Same basic idea as deterministic case
  - Only now all timing quantities are distributions, which can overlap

Max Operation

- Deterministic Max
  - Pretty obvious, one value is the largest
  - 9 is always > 8, 8.5
Statistical Max Operation

- Must take statistical max of distributions
  - The output is also a distribution

- How this works (in theory)
  - Sample all input distributions
    - At a unique, random sample location per iteration
    - The same sample location per distribution
  - Keep the max value from each sample
    - Give a 'point' to the distribution producing max
  - Repeat for all MC iterations
    - Final set of max values defines max distribution
  - Tightness / Criticality value:
    - Ratio of points per distribution / total samples

- The above description defines a "Monte Carlo" Max
  - Actually sample the input distributions
  - Analytic equivalents exist (Clark eqns)
    - Much higher performance
    - Equivalent for 2-way max
      - Higher orders require a series of 2-way Max ops
      - Solution can be order dependent

Statistical Max Operation

- Sometimes as easy as deterministic case
  - Distribution on right always greater (the max)
    - (for values within a ~ +/-5 sigma region)
    - Tightness for right distribution ~ 1.0, for left distribution ~0.0
  - Max distribution ~ Mean 11, Std. Dev. 0.4

| MEAN = 7, Std. Dev. = 0.5 | MEAN = 11, Std. Dev. = 0.4 |
Statistical Max Operation

- Sample distribution, note max result
  - In sampling shown below, blue > purple
  - In MC max, blue result would be kept
    - And blue gets a ‘point’

\[ \text{MEAN} = 7, \text{Std. Dev.} = 0.7 \]
\[ \text{MEAN} = 8, \text{Std. Dev.} = 0.5 \]

---

Statistical Max Operation

- Sample distribution, note max result
  - In sampling shown below, purple > blue
  - In MC max, purple result would be kept
    - And purple gets a ‘point’

- Repeat this procedure for many iterations
  - Store resultant ‘winning’ samples, index ‘win’ counts

\[ \text{MEAN} = 7, \text{Std. Dev.} = 0.7 \]
\[ \text{MEAN} = 8, \text{Std. Dev.} = 0.5 \]
Statistical Max Operation

- The “Tightness probability” / Criticality
  - Defines how frequently a given distribution will be greater
    - E.g., purple > blue 80% of the time
    - Out of 10000 samples, purple was the max 8000 times
    - Tightness of purple = 0.8, and of blue = 0.2

\[
\text{MEAN} = 7, \text{Std. Dev.} = 0.7
\]
\[
\text{MEAN} = 8, \text{Std. Dev.} = 0.5
\]

Statistical Timing

\[
AT_Z = \text{MAX} \left( AT_A + d_{AZ}, \ AT_B + d_{BZ}, \ AT_C + d_{CZ} \right)
\]
\[
= T_{AZ}^* (AT_A + d_{AZ}) + T_{BZ}^* (AT_B + d_{BZ}) + T_{CZ}^* (AT_C + d_{CZ})
\]

\(T_i\) are the tightness probabilities for each arc, which defines how probable it is for a given arc to provide max \(AT_Z\)
Physical Interpretation

Start with some distribution of $T_{ox}$

---

Physical Interpretation

Assume all arcs (AT+d) sensitive to $T_{ox}$ only, overlay arc sensitivities on $T_{ox}$ distribution.
Physical Interpretation

Break distribution into regions where a given arc dominates.

Sensitivity of $A_{AZ}$ delay to $T_{ox}$
Causes largest delay in this region.

Physical Interpretation

Tightness probabilities are probability a given arc will dominate as a result of process variation.

$T_{AZ} = 0.25$
$T_{BZ} = 0.1$
$T_{CZ} = 0.65$
### Physical Interpretation

\[ AT_Z = \text{MAX} (\ AT_A + d_{AZ}, \ AT_B + d_{BZ}, \ AT_C + d_{CZ}) \]

\[ = 0.25 \times (AT_A + d_{AZ}) + 0.1 \times (AT_B + d_{BZ}) + 0.65 \times (AT_C + d_{CZ}) \]

### Statistical Timing

\[ AT_Z = \text{MAX} (\ AT_A + d_{AZ}, \ AT_B + d_{BZ}, \ AT_C + d_{CZ}) \]

\[ = 0.25 \times (AT_A + d_{AZ}) + 0.1 \times (AT_B + d_{BZ}) + 0.65 \times (AT_C + d_{CZ}) \]
Aside: Multi-way max/min ops (1/2)

Using Clark, Convolution, can only perform 2-way Max/Min

N-way operations performed as series of 2-way ops
First, \(M_1 = \text{Max}(A, B)\), Say \(T = 0.2\) (for \(A, B=(1-T) = 0.8\))
\(M_2 = \text{Max}(M_1, C)\) \(T = 0.3\) (0.7 for \(C\))
\(M_3 = \text{Max}(M_2, D)\) \(T = 0.6\) (0.4 for \(D\))

“Raw” tightness probabilities:

Multi-way max/min ops (2/2)...

Next, use conditional probabilities, from the bottom up.
\(T(D) = 0.4\), last in operation, already considers all others, can be used as-is.
‘Available’ probability is now \(1.0 - 0.4 = 0.6\)
Remaining 3 ‘legs’ have a 60% probability of dominating
Of this 60%, \(C\) will dominate 70% of the time
(C already considers \(\text{Max}(A, B)\)); \(C\) dominates \(\text{Max}(A, B)\) by 0.7
\(T(C) = 0.6 \times 0.7 = 0.42\)

Remaining = 0.6 - 0.42 = 0.18
Max (A,B) will dominate 18% of the time

Of this, B will dominate 80% of the time:
\(T(B) = 0.18 \times 0.8 = 0.144\)

And A will dominate 20% of the time:
\(T(A) = 0.18 \times 0.2 = 0.036\)
References


S-ECSM Link:

http://www.si2.org/opendeda.si2.org/projects/omedistrib