Design for Variability and Signoff Tips

Alexander Tetelbaum
Abelite Design Automation, Walnut Creek, USA
alex@abelite-da.com

ABSTRACT

The paper provides useful design tips and recommendations on how to handle multiple global and local variations in process (transistor, wire and via parameters), voltage and temperatures before and during timing signoff. The presentation will share our experience and knowledge that designers can use in their practice. The paper also teaches how to better use PT-SI or PT-VX, or move to Abelite advanced timing tools.

Table of Contents

Table of Figures

1. Introduction

The corner-based timing signoff approach is a historical and traditional method that has justified a development and enhancements of conventional STA tools and signoff flows. The number of signoff corners exponentially grows along with an increase of variation sources, their magnitude, and timing margins. It becomes a bottleneck in the design flow and leads to an over-margining, over-design, a loss in the System-On-Chip (SoC) performance, timing yield, costs, etc. It causes a timing signoff deadlock and still does not guarantee against a silicon failure. This paper examines the situation and outlines possible solutions.

The corner-based timing signoff methodology and the corner number used in this methodology increase the duration of the timing signoff, make timing closure difficult and worsen most of design metrics. The corner-based timing signoff is a justification for the current design flow and contemporary signoff tools. It has multiple impacts on the design flow, Time-to-Market (TTM), cost, SoC performance F, timing yield Y, etc. It becomes a problem for getting the most benefits from moving to next advanced technology nodes. You can find all the details in white paper [1]. The same paper also discusses the conventional timing signoff methodology in details. It provides a definition of the current timing closure and the timing yield. It shows that the conventional timing signoff does not support the timing yield as a design signoff requirement and it becomes a challenge. Then, timing derating (margins) methods of contemporary STA tools, which should cover for variations, are considered. An increase of variation sources and their magnitude leads to losses in the SoC performance and diminishes other design metrics. Some limitations of current derating methods are considered and, then, it is shown that Statistical STA (SSTA) tools provide a partial solution but are not panacea. Later, in this paper [1], we consider a signoff optimism and conservatism (pessimism), different variability sources and, finally, the timing signoff deadlock.

Chapter 2 provides important design recommendations on selecting and minimizing timing signoff corners.

Chapter 3 provides design and timing signoff recommendations and tips that will minimize delay variations and, in most cases, are the same for the corner-based methodology and new statistical methodologies. They include discussions on corners and minimization of their number, using the useful skew, and variations in paths with zero and a useful skew.

Chapter 3 provides important design recommendations and timing signoff tips on how to minimize delay variations in cells by reducing slew and load.

2. Design for Variability and Signoff Tips

In this chapter, we provide important design recommendations on selecting and minimizing timing signoff corners.

As long as you and your company believe in and use the corner-based signoff, you practically need to close timing at all conceivable corners, at least, at final stage of the timing signoff. Otherwise, there is a risk of missing violations in paths with rare "bad" structures/properties.

When you think if a particular corner is redundant (dominated by other corner), consider the minimum slack as a sum and subtraction of the three sub-path delays (launch, data, and capture) vs. just the maximum or minimum delay in one sub-path.

Let's illustrate why it may be risky to remove any so-called "redundant" (or dominated, or less important, etc.) corners. For example, corner SS + Fast-Metal is needed for the setup check, because a violation may happen in a path where launch and data paths are cell-delay dominated and capture is metal-delay dominated. So, using only corner SS + Slow-Metal, as recommended by [2], may lead to missing violations. The Figure below illustrates this case.

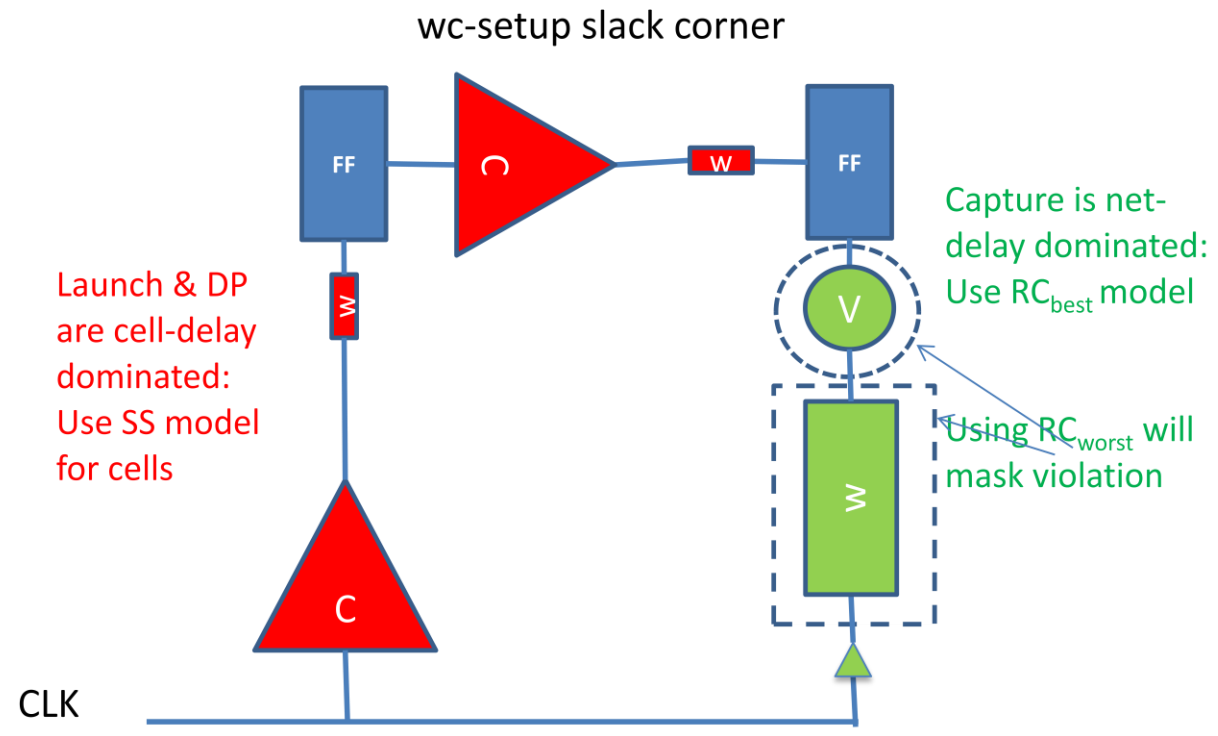

Figure 1 Illustration of a WC-corner for setup slack check

Another example is a corner FF + V_max + Slow-Metal, which is needed for the setup check, because a violation may occur in a path where the launch and data paths are metal-delay dominated and the capture is cell-delay dominated. The Figure below illustrates this case.

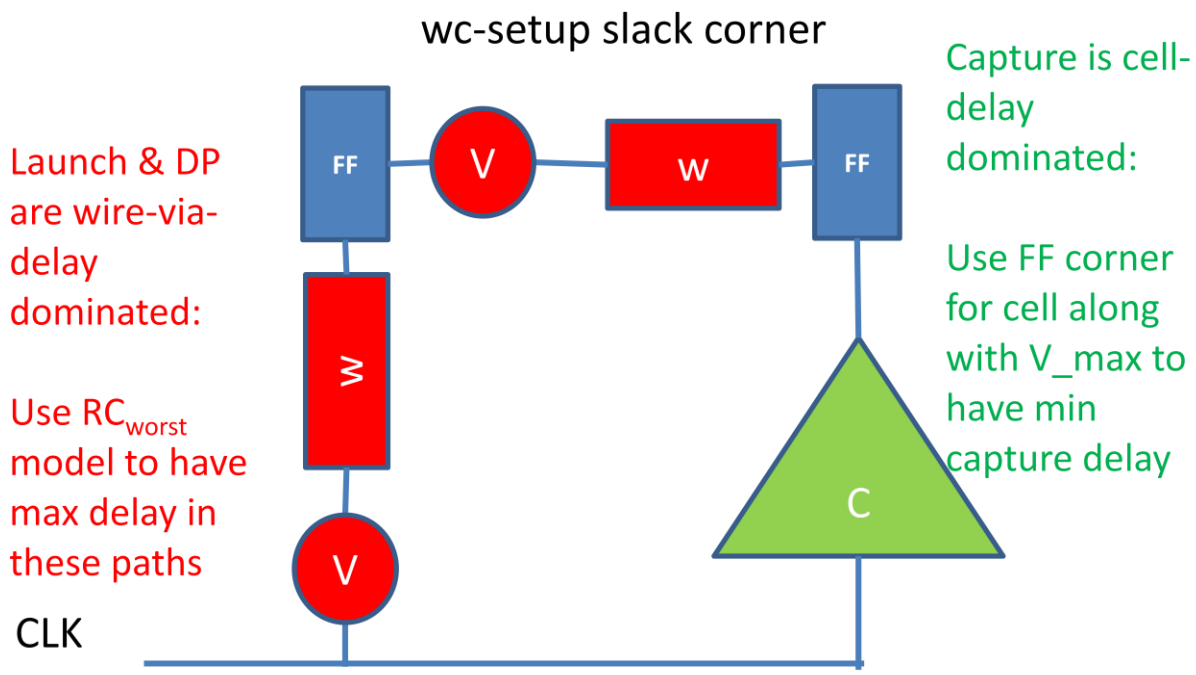

Figure 2 Illustration of a WC-corner for setup slack check

There are other "strange" corners needed for setup and hold checks. It may be important to catch all timing critical paths with not a "typical" structure and investigate them in more details.

At the final stage of the timing signoff, it may be important to add more than 2-3 temperature corners due to T-inversion [3]. It may be not enough to consider only two or even three temperature points T because the extreme stage/sub-path delays (see Figure below) may occur at any temperature T_min < T < T_max due to:

- Temperature inversion effect
- Distribution of delays between cells and nets

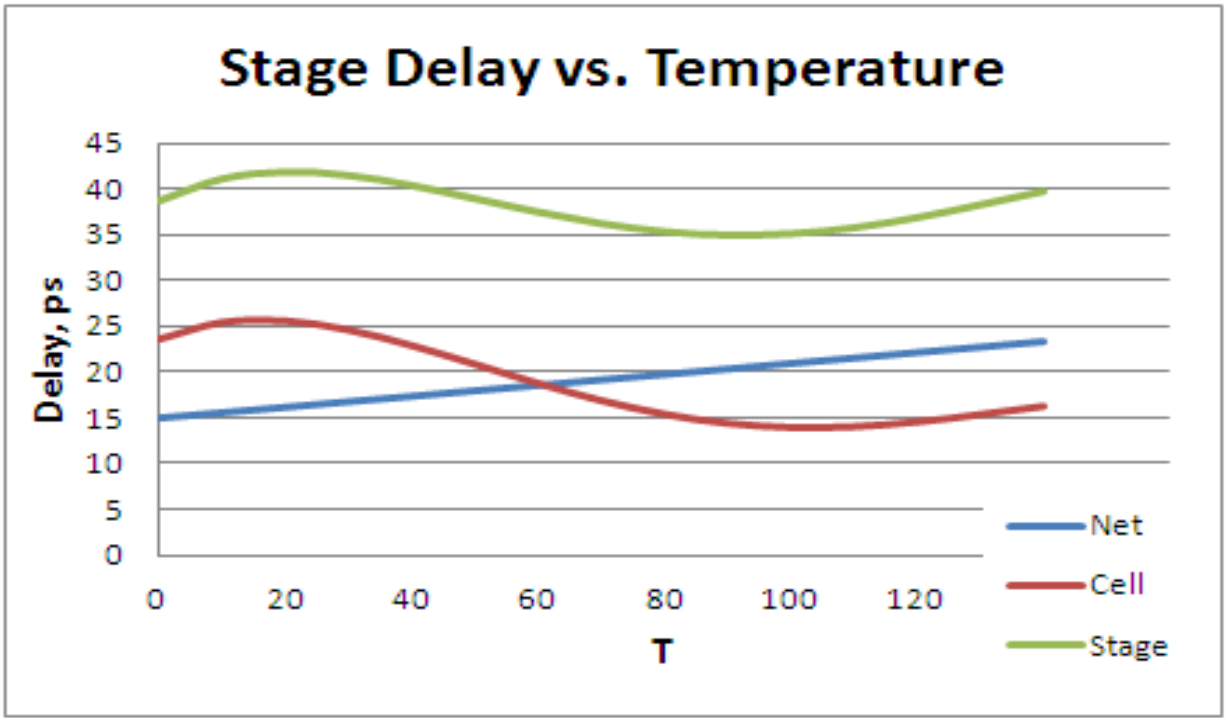

Figure 3 Illustration of delays in one cell, net, and the stage (cell & net)

The delay change is ~1-3% for each 10C temperature change and it increases for each next technology node. The next Figure shows an example of delays in cells C1 and C2 and the total sub-path delay, where the sub-path includes cells C1 and C2. Note that the sub-path delay is not a monotonic function of T.

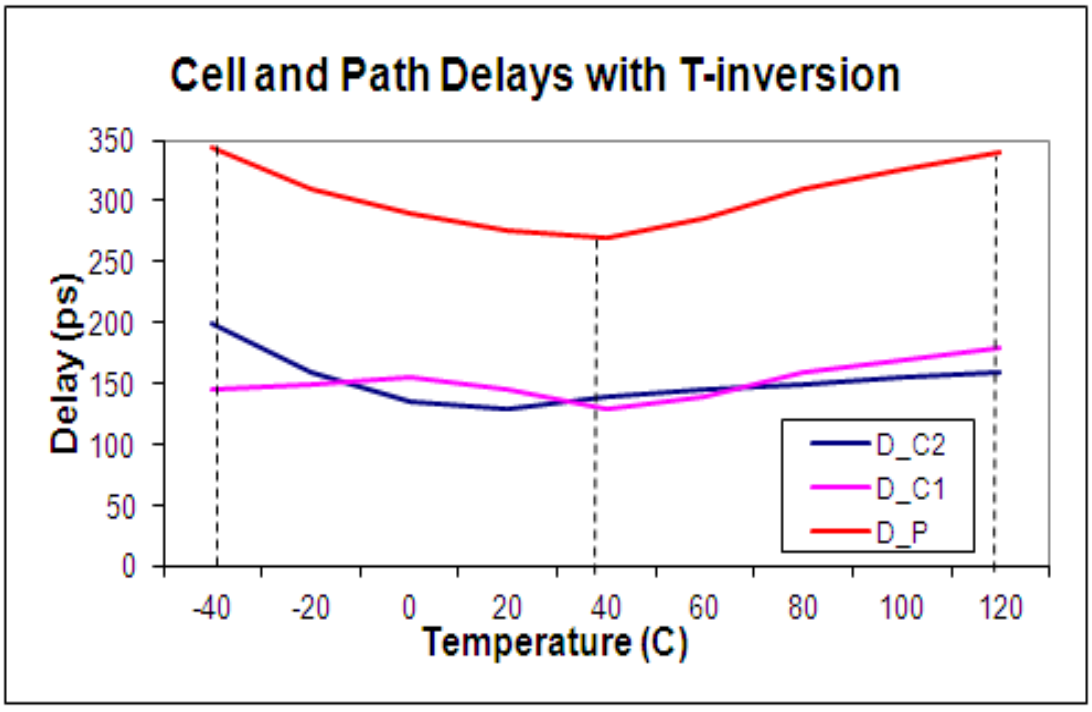

Figure 4 Illustration of delays in cells C1, C2, and sub-path P=C1+C2

If you are using relative small number of corners (at least, during initial stages of the timing signoff), we recommend the following to minimize a risk of silicon failure:

- Initially use increased OCV/AOCV derates to find all potential violators and hopefully not all of them will be the final violators, which are a must to fix
- Re-run PT with your regular OCV/AOCV derates for all found violators at all possible corners—it will likely eliminate some "false" violations found previously

Place and Route tools do not separately balance cell and net delays in clocks. It may lead to optimism in timing. For example, the next Figure shows a pictogram for a path with a "bad" structure: The launch and data paths are fully net delay dominated and the capture is fully cell delay dominated. Note that in this pictogram and follow up pictograms, we use triangles to represent cell delays, rectangles for wire delays and circles for via delays. The size of each shape is proportional to the corresponding delay. Thus, if any shape is absent, it means zero delay of this type. Signoff at many not traditional PVT/RC/VRC corners and sufficiently big margins are needed to avoid a silicon failure. Derate tables cannot take into account an inter-clock correlation properly and it may lead to optimism too.

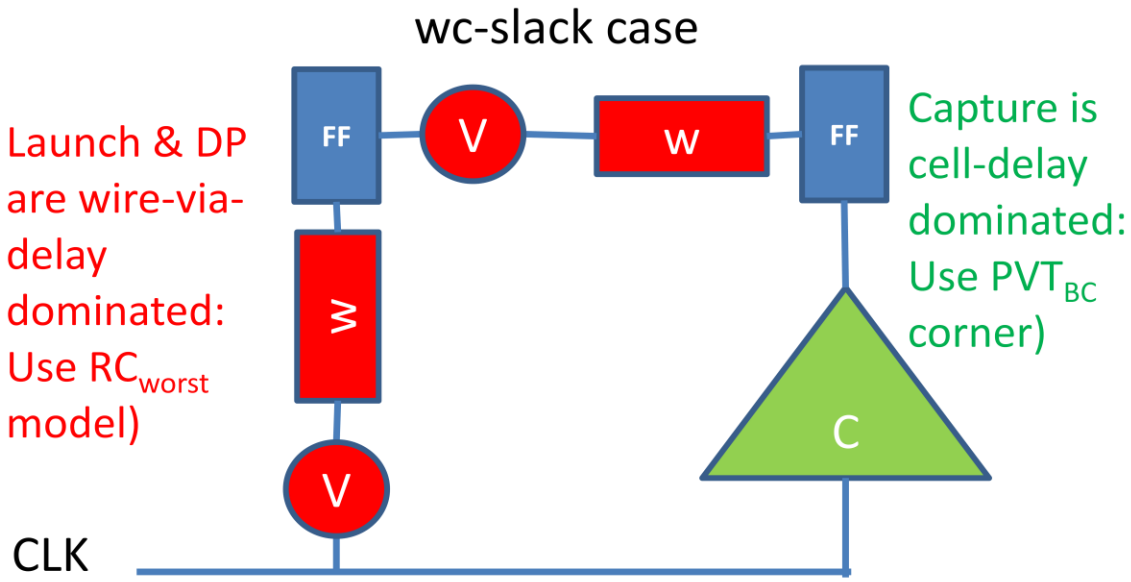

Figure 5 Illustration of possible optimism in timing

Thus, in order to avoid possible optimism, investigate the most timing critical paths at the final stage of the timing signoff use additional "non-traditional" corners and try to properly balance cell and net delays separately.

Hold violations are the most critical and must be never missed. Timing margins (derates) are critical and must be relatively big to avoid hold catastrophic failure illustrated in Figure below. If one ignores EDA tools and libraries inaccuracies, then a risk of a failure increases because the tools are not modeling some analog effects in the digital logic (as an example) with the error on the timing up to ±5% and, thus, " ... *there will be hidden timing violations that will not be dis-*

covered until first silicon." [2]. Note that foundries (like GlobalFoundry) validate quality of libraries and EDA tools, and foundry PDK before releasing reference design flows.

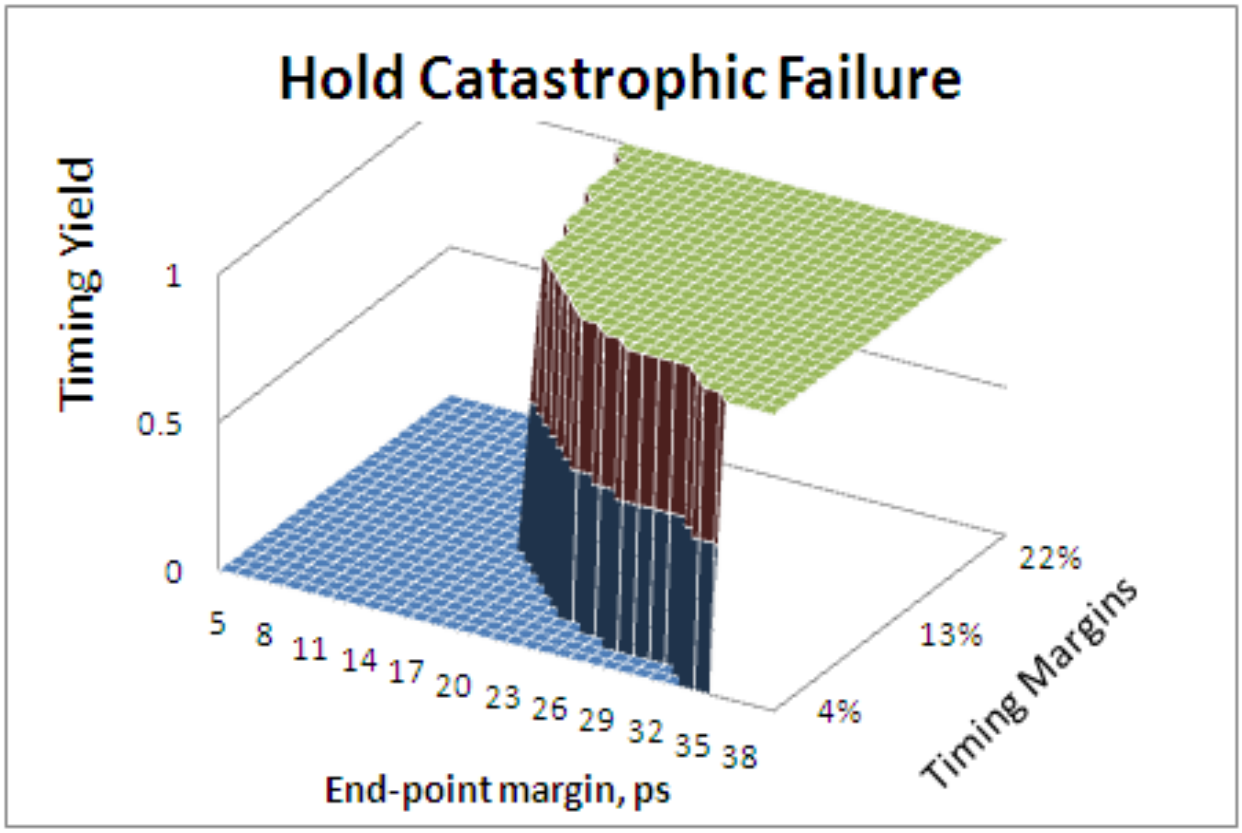

Figure 6 Illustration of hold timing yield vs. margins

Setup violations are also important and must be not underestimated because it leads to:

- Less timing yield
- Failure to meet spec performance F

3. Mitigating Variability in Paths with Zero and Useful Skew

In this chapter, we provide important design recommendations and timing signoff tips that will minimize delay variations for paths with zero or useful skew.

Zero Skew Paths

For paths with zero (or close to zero) skew:

- Try to obtain zero skew for cell and net delays separately—it will minimize the clock skew change (variability) between corners

- Place flip-flops FF1 and FF2 as close to each other as possible—it will minimize the depth of divergent clock paths and make clocks more correlated
- Use as much symmetry in clocks as possible. It includes clock path depths, type of cells and wiring, loads, etc. Note that clock paths with all cells the same (like buffers) have smaller skew variation that paths with the same structure but different cells (like half of them are buffers and the other half is invertors). This is not trivial conclusion.
- Avoid using delay cells in one clock path if the same delay cells are not present in other clock path. Any asymmetric clock design increases variations. Most derating methods have issues when delays are not about the same in paths. Also, clocks cells usually have higher variation.
- Use maximally random cells and nets in the data path (DP)—Use different cell types, VTs, strengths, orientation, voltage supply, increase LDE, etc. and different wire/via features like lengths, width, sizes, layers, orientation, etc.
- Use highly correlated cells and nets in the clocks—the same clock cells (type, VT, strength, orientation, LDE, etc.); the same or close ramptimes and loads; similar wire/via properties like the length, width, size, layers, etc. Figures below illustrates skew variation in ps vs. clock depth n, where $n = n_{c1} = n_{c2}$ (n_{c1} and n_{c2} is the launch and capture clock depth respectively); $v = RSS(v_C, v_R) = 0.1$ is the total relative cell variation and v_C and v_R is the correlated and random cell variation respectively; and all stage delays are 20ps.

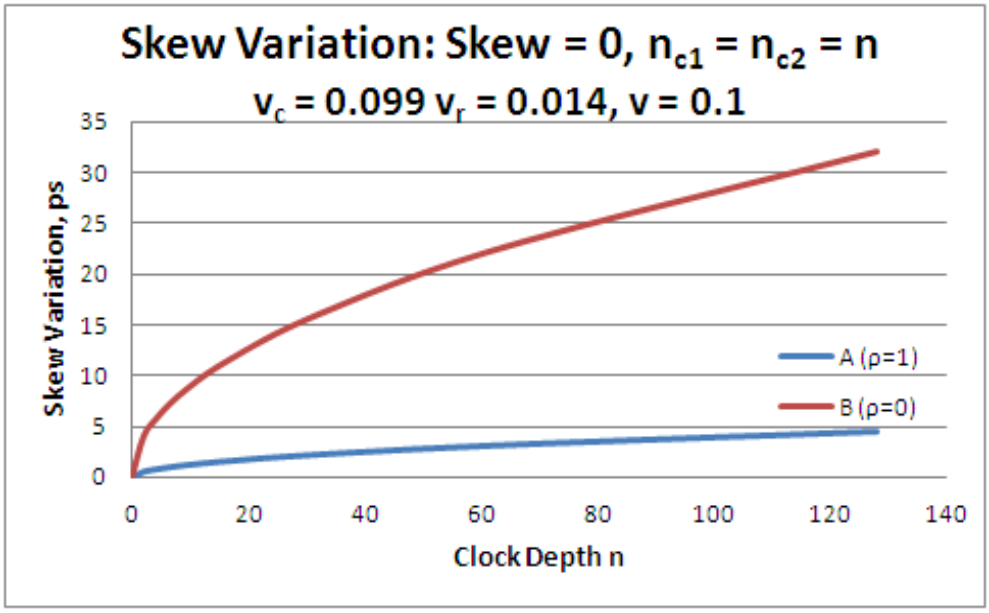

Figure 7 Illustration of skew variation when $v_C >> v_R$

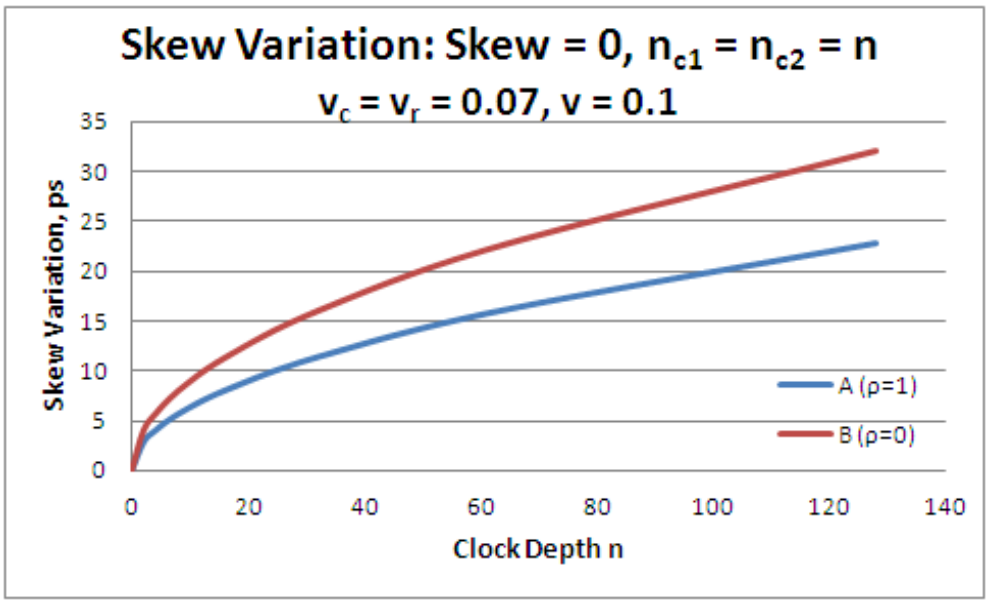

Figure 8 Illustration of skew variation when $v_C = v_R$

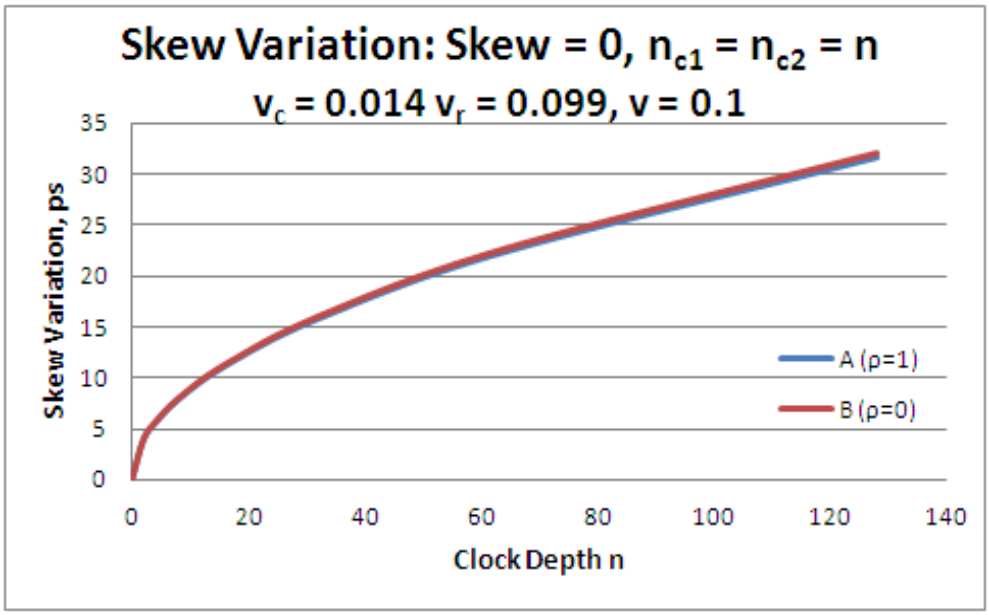

Figure 9 Illustration of skew variation when $v_C << v_R$

As an extended conclusion based on our study, we can mention that:

- Correlated clocks (A-clocks) are better—they produce less skew variation
- The difference in the variation is more significant when correlated variation is greater than random variation
- Clocks with more stages (n) have more skew variation
- If most variations is due random variations (not correlated), then effect is less
- Use:
  - Similar clock cells/nets and place them as close as possible

- Symmetries in wiring and clock structure also improve variation

Avoid using useful skew that increases chances to get into the signoff deadlock. The useful skew approach is having a not zero (or close to zero) clock skew in order to improve the slack (e.g., when the capture sub-path delay is much bigger than the launch sub-path delay to allow the data sub-path have more delay than the clock period).

Useful Skew Paths

If it is still not possible to close timing without useful skew, then:

- Use more signoff corners and additional analysis of such paths, if the corner-based signoff is used.
- Use more clock buffers (or invertors) instead of a delay cell(s) because most derating methods have issues when delays are not uniformly distributed in paths and, also, the magnitude of variations in clocks cells are usually higher.

Use the following solutions in useful skew paths that will minimize clock skew and slack variations.

Solution 1: Use maximum randomness in all components of such paths—use different cell types, VTs, strengths, orientation, voltage supply, increase LDE, etc. and different wire/via features like lengths, width, sizes, layers, orientation, etc. The next Figure shows this type of a path structure, where:

- Data Path (DP) has structure STR_{DP_R} of maximum randomness and delay equal to T_{DP}
- Clock 1 (Launch) has Structure STR_{CLK1_R} of maximum randomness and delay T_{CLK1}
- Clock 2 (Capture) has Structure STR_{CLK2_R} of maximum randomness and delay T_{CLK1} equal to the sum of T_{CLK1} and the useful skew value T_{SKEW_U}

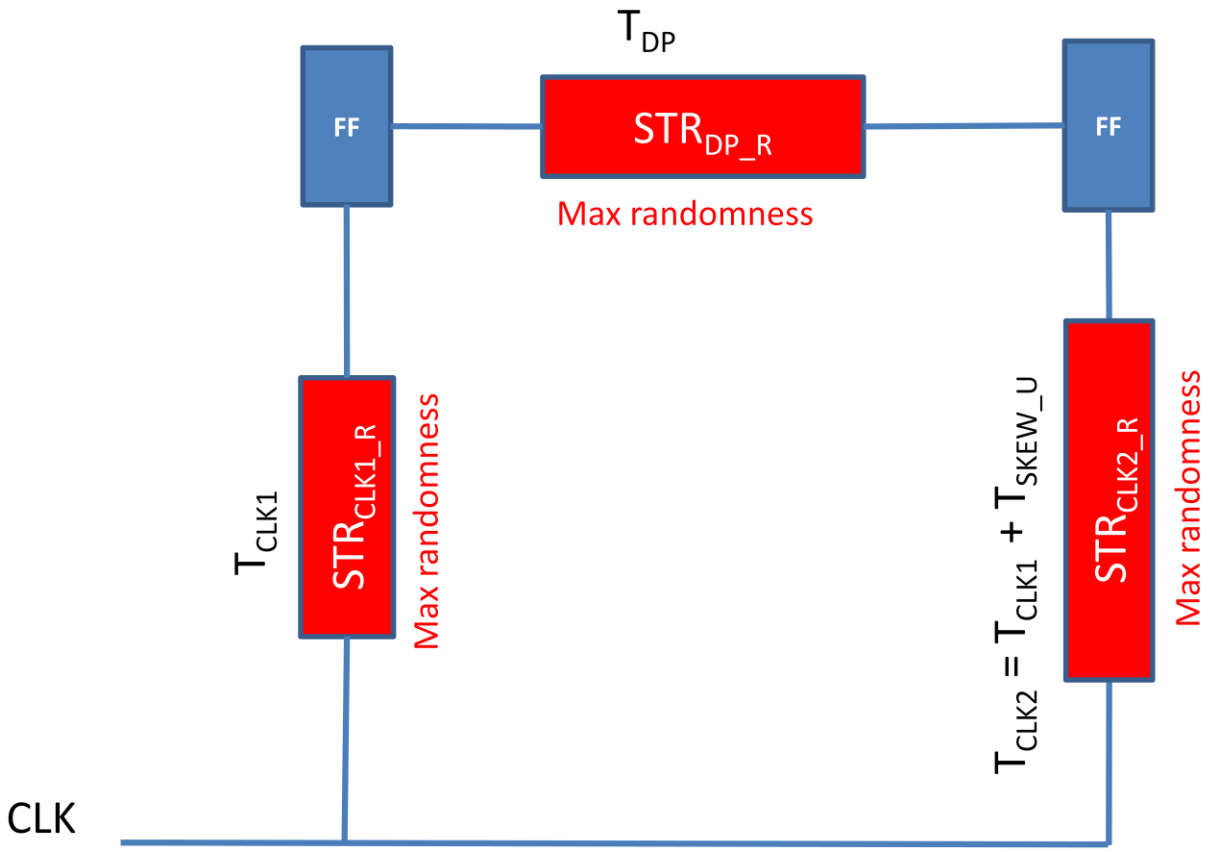

Figure 10 Solution 1 for paths with useful skew

Figures below illustrates the skew variation in ps vs. Δn, where $N = n_{c1} + n_{c2}$ (n_{c1} and n_{c2} is the launch and capture clock depth respectively); $\Delta n = n_{c2} - n_{c1}$; $v = RSS(v_C, v_R) = 0.1$ is the total relative cell variation and v_C and v_R is the correlated and random cell variation (=0.07) respectively; and all stage delays are 20ps.

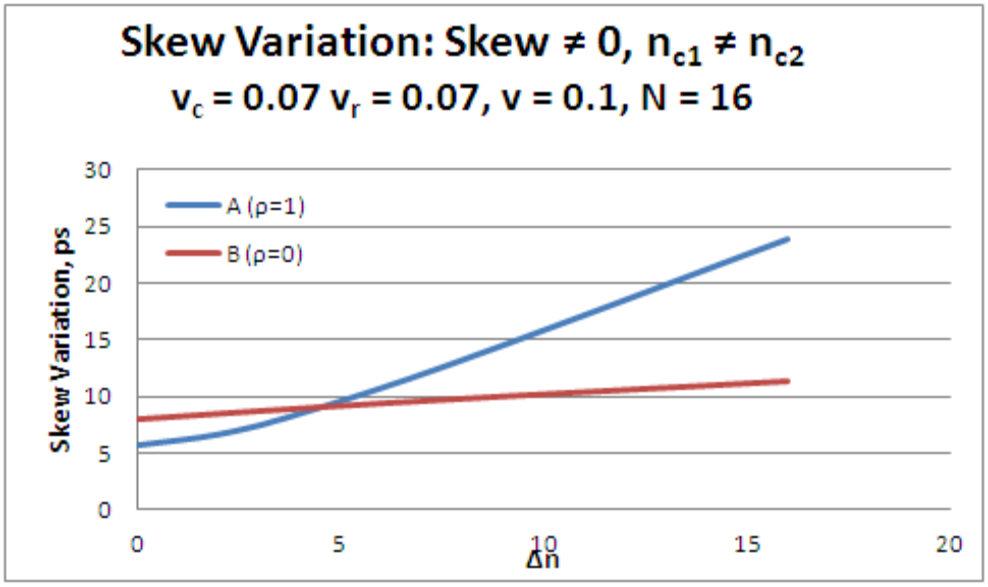

Figure 11 Illustration of skew variation for N = 16

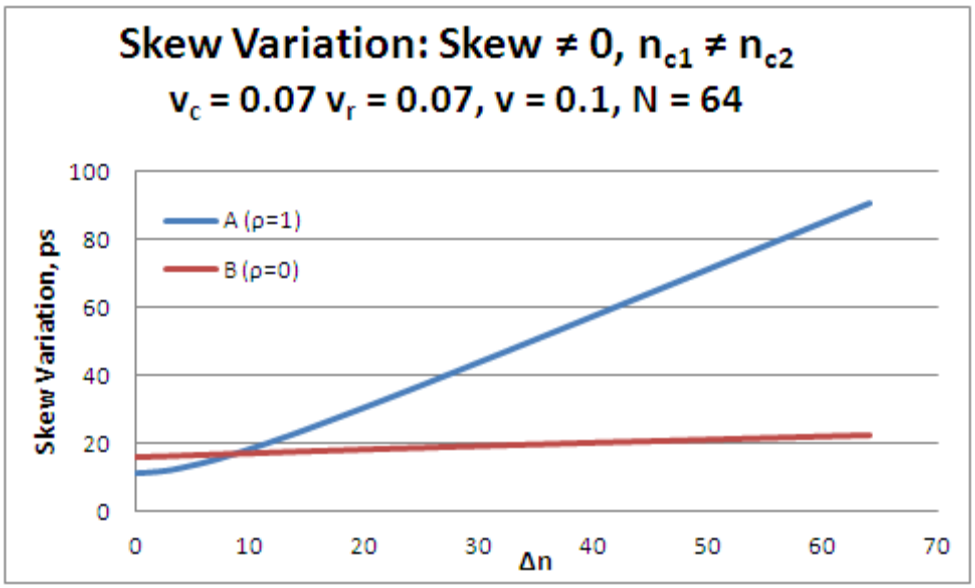

Figure 12 Illustration of skew variation for N = 64

As an extended conclusion based on our study, we can mention that:

- Uncorrelated (random) clocks (B-clocks in the figures) are better—they produce less skew variation
- If Δn is low, it's a case of close to zero skew (will be discussed below)
- The difference between skew variations in B-clocks vs. a-clocks grows with more useful skew (Δn increase)
- Clocks with more stages (depth) N have more skew variation
- A-clocks (highly correlated clocks): Skew variation is:
  - ~30% more for any N when the ration of the correlated variation to random one is >> 1

  - ~50-80% les for any N when the ration of the correlated variation to random one is <<1 (50% is for small N and 80 for big N)
- B-clocks: Skew variation:
  - Does not practically depend on the ration of the correlated variation to the random one
- Use (not trivial conclusions):
  - Different clock cells/nets and place them as far from each other as possible
  - Asymmetries also improve variation
- Use all (cells, wires, vias, etc.) as different as possible in DP for A- and B-clocks and any skew—Any increase in correlations in Data Path will increase the slack variation.

Solution 2: Use max randomness in the DP and a part of the capture clock. The next Figure shows this type of a path structure, where:

- DP has Structure STR_{DP_R} of maximum randomness and delay equal to T_{DP}
- Clock 1 has any Structure STR_{CLK1} and delay T_{CLK1}
- Part 1 of clock 2 has Structure STR_{CLK1} and delay equal to T_{CLK1}
- Part 2 of Clock 2 has Structure STR_{CLK2_R} of maximum randomness and delay equal to the useful skew value T_{SKEW_U}

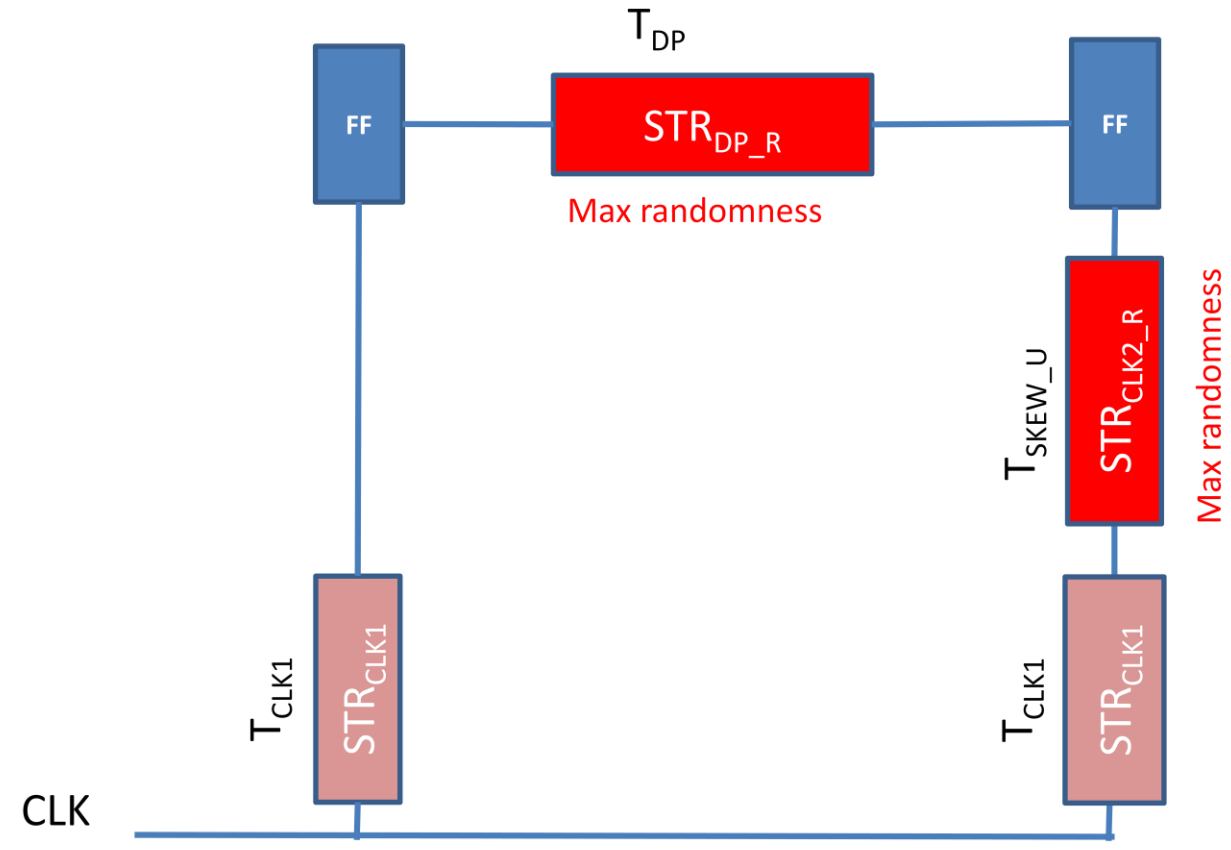

Figure 13 Solution 2 for paths with useful skew

This solution is beneficial when clocks distance is small and there is increased correlation between clocks. Thus, no matter what structure of launch clock is, there is high cancellation of skew variation.

Solution 3: Use all sub-paths structures to minimize skew and slack variations for paths with high correlations between sub-paths. The next Figure shows this type of a path structure, where:

- Part 1 of DP has Structure $STR1_{DP\_R}$ of maximum randomness and delay equal to $T1_{DP}$, which is slightly less than clock period T_{CLK}
- Part 2 of DP has Structure $STR2_{DP\_R}$ of maximum randomness and delay equal to T_{SKEW_U}
- Clock 1 has any Structure any STR_{CLK1} and delay T_{CLK1}
- Part 1 of clock 2 has Structure STR_{CLK1} and delay equal to T_{CLK1}
- Part 2 of Clock 2 has Structure $STR2_{DP\_R}$, which is the same as Part 2 of DP and also has maximum randomness, and delay equal to the useful skew value T_{SKEW_U}

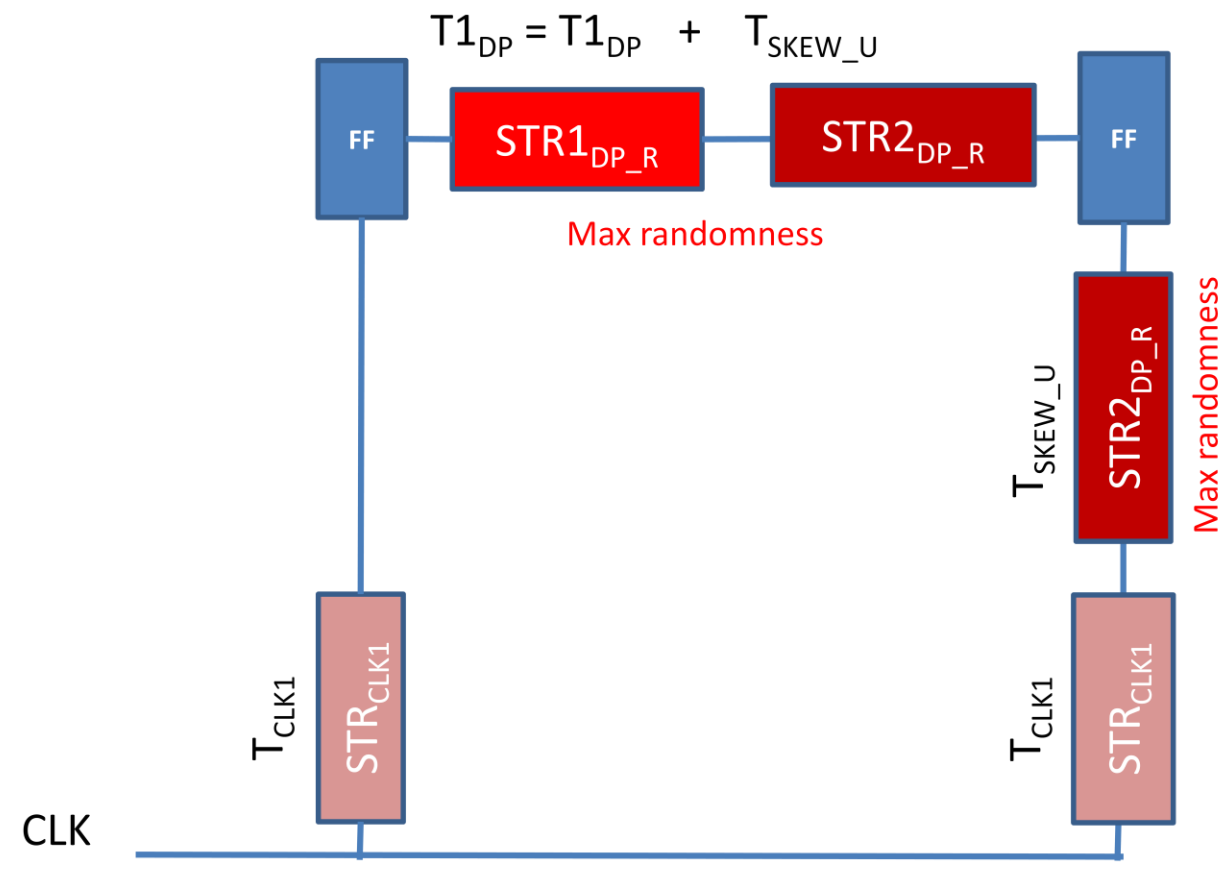

Figure 14 Solution 3 for paths with useful skew

This solution is especially beneficial when clocks and DP distance is very small (all sub-paths are placed close to each other) and there is high correlation between clocks and DP. Thus, no matter what structure of launch clock is, there is high cancellation of skew variation between clocks and partial cancelation between DP and capture clock. It means that almost all variation comes from Part 1 of DP, which has random structure with a reduced variation.

Finally, the following design recommendations for timing critical and failing paths will always minimize variations and improve signal integrity and are common for zero and useful skew:

- Avoid using LVT clock cells because they have the highest variation magnitude
- Minimize all correlations between cells and nets in data paths (as described) above—introduce more randomness everywhere in DPs

4. Minimizing Variability in Critical Paths

In this chapter, we provide important design recommendations and timing signoff tips on how to minimize delay variations in cells by reducing slew and load.

Cell delay variation caused by variations in load and ramptime (slew) depends on specific values of cell load C and input ramptime R. Load C and ramptime R impact cell J variability:

- Big cell load C or/and slew R have more variations caused by C or/and R variations

To minimize variability in cell J belonging to a timing critical path one can use the following two new methods:

- **Method 1**: Minimize the cell J load C by special buffer B insertion that reduces cell J fanout and load. No extra buffers in critical path (assuming relatively short wire between Cell J and J+1).
- **Method 2**: Minimize the cell J slew R by special buffer B insertion that reduces cell J-1 fanout and load .
- See two Figures (Pictograms) below where:
  - Cells and nets in timing critical path are colored red
  - Size of cell is proportional cell driving strength

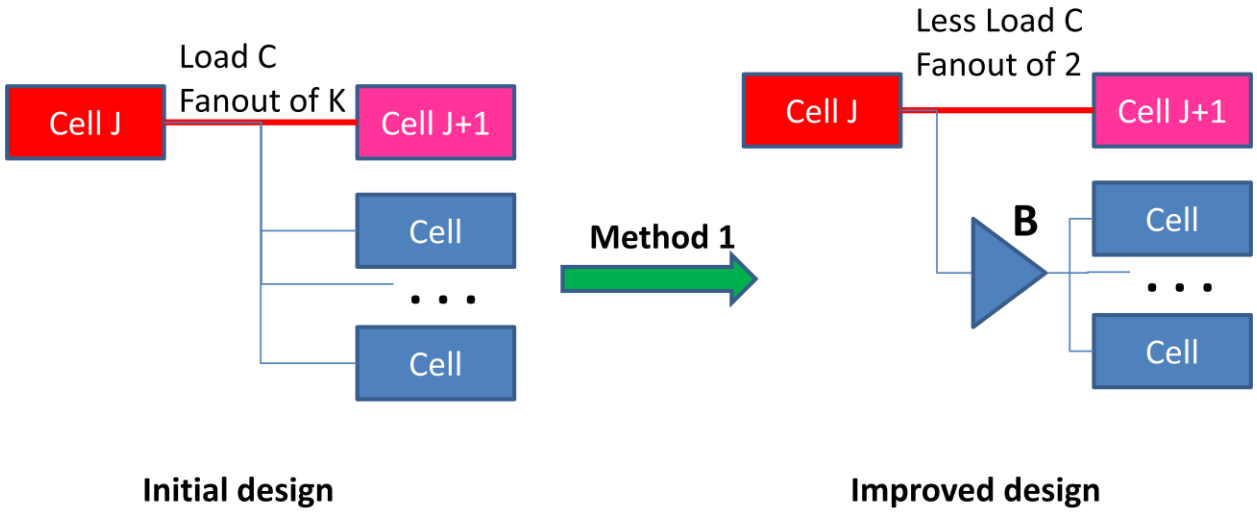

Figure 15 Method 1of minimizing variability in cell J

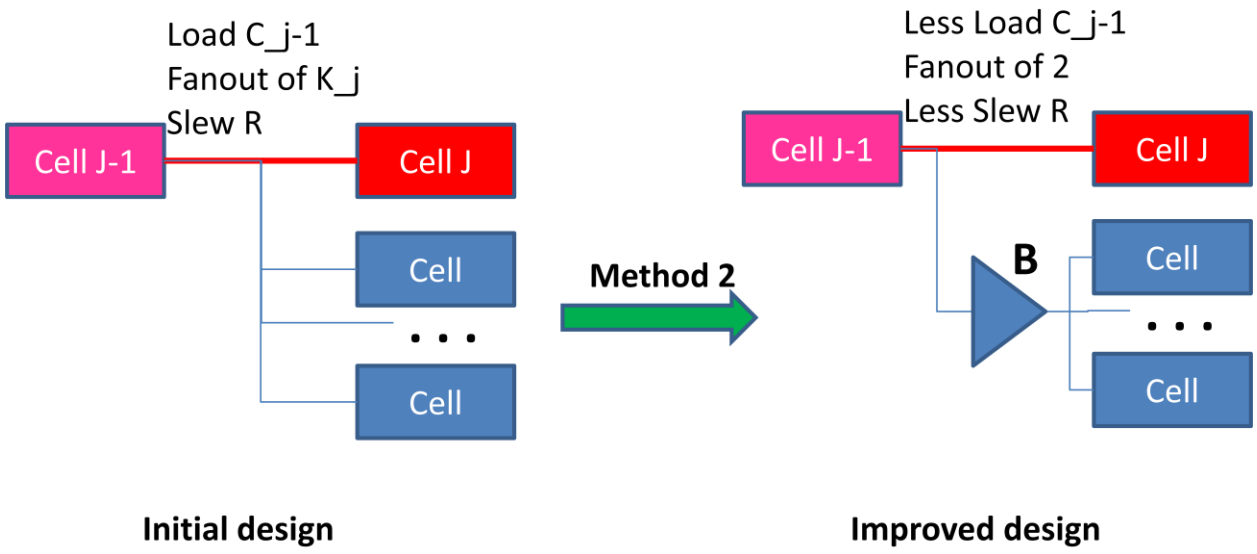

Figure 16 Method 2 of minimizing variability in cell J

Note that methods are listed in order of their effectiveness. Also, we do not provide separate illustrations for known methods, which reduce C or/and R, that may follow up the new two methods. Known methods like:

- Upsizing cell J
- Upsizing cell J-1
- Buffer insertion in critical nets (before and/or after cell J)
- Placement & routing optimization to shorten critical nets

An example of combining 3 methods (Method 1 &2 Plus Cell J Upsizing) is presented below:

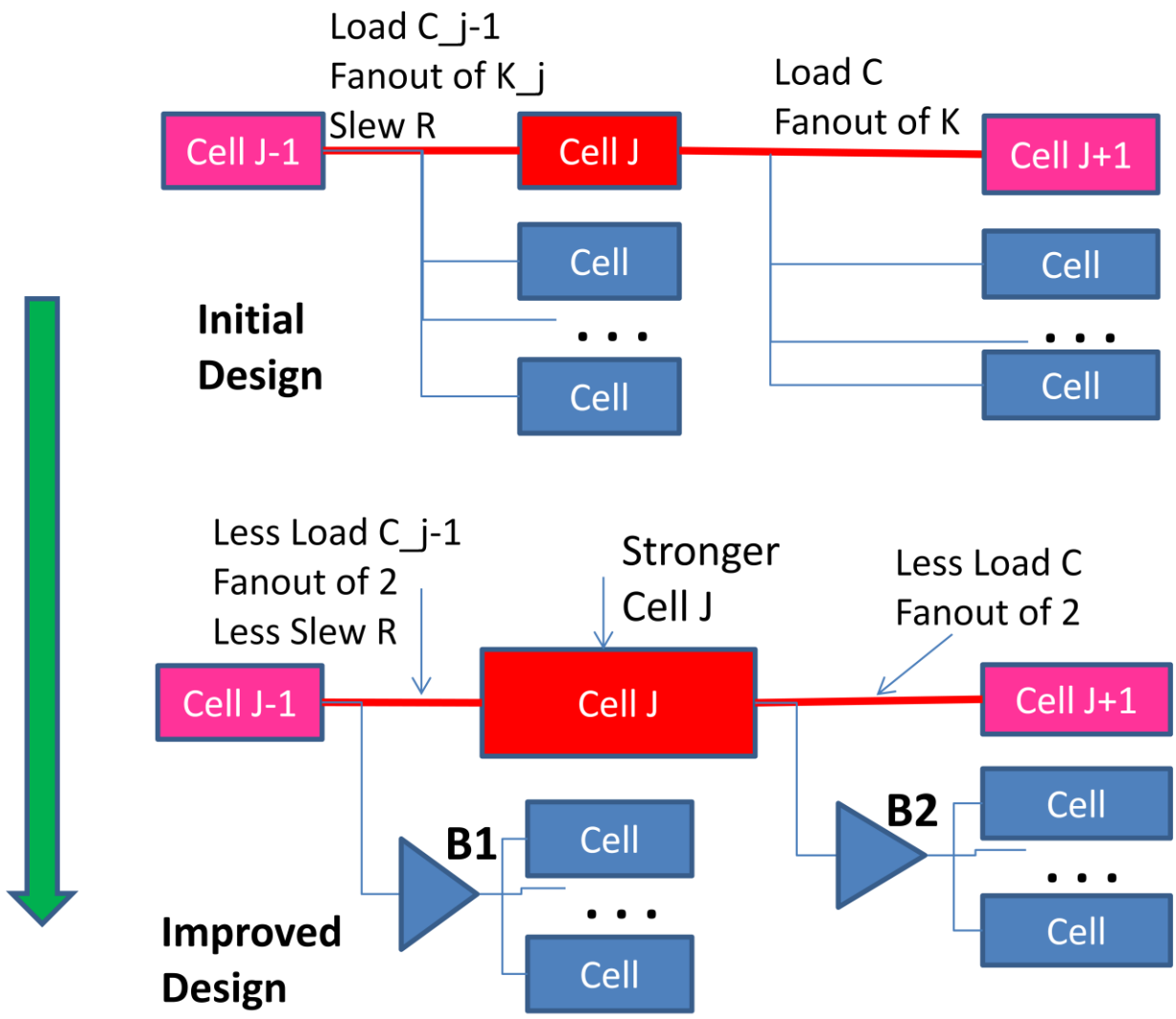

Figure 17 Combining several methods of minimizing variability in cell J

PrimeTime POCV timing derating method has some drawbacks and possibly issues with handling specific cell loads C and slews R. By default, POCV uses one relative variation v (so-called "coefficient" or single-parameter), which is representing all variation sources for all cells no matter their type, VT and driving strength, and this variation v must be provided by user. Optionally [4], user may provide POCV slew-load table for each delay timing arc in Liberty Variation Format (LVF). It theoretically improves accuracy, but it requires a lot of characterization for different cells, VTs, strengths, and PVT corners. Each cell has its own C- and R-values that may be quite different. POCV estimates Delay Variability Scale M in range $1 < M < 7$ [5] for each cell as a function of specific C & R: $\Delta M = \Delta M(C, R)$ using POCV LFT tables if provided and $\Delta M=0$ if no tables. Then, as far as we understand, POCV multiplies the whole cell variation v by scale $M=1+\Delta M$. The examples of scales $M=F(C, R, Cell)$ [5] is shown in Table and two graphs below:

	Delay variability as a function of Slew/Load				
(Slew , Load)	(min , min)	(min , max)	(mid , mid)	(max , max)	(max , min)
INV4X (Max Temp)	1.22X	1.56X	X	1.11X	6.78X
BUF4X (Max Temp)	1.89X	1.33X	1.67X	1.44X	4.33X
INV4X (Min Temp)	1.44X	1.78X	1.22X	1.33X	2.56X
BUF4X (Min Temp)	2.11X	1.67X	1.44X	1.44X	2.78X

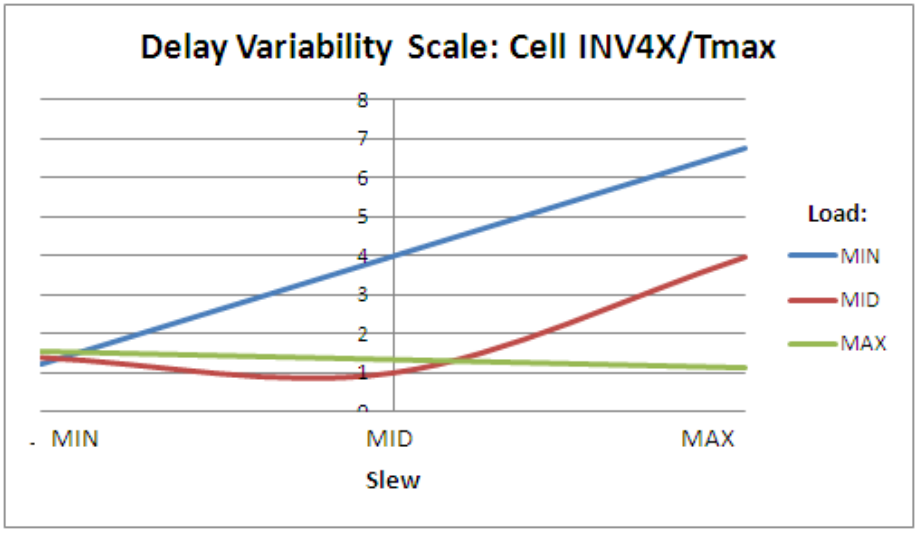

Figure 18 Delay Variability Scale for INV4x/MIN Temp

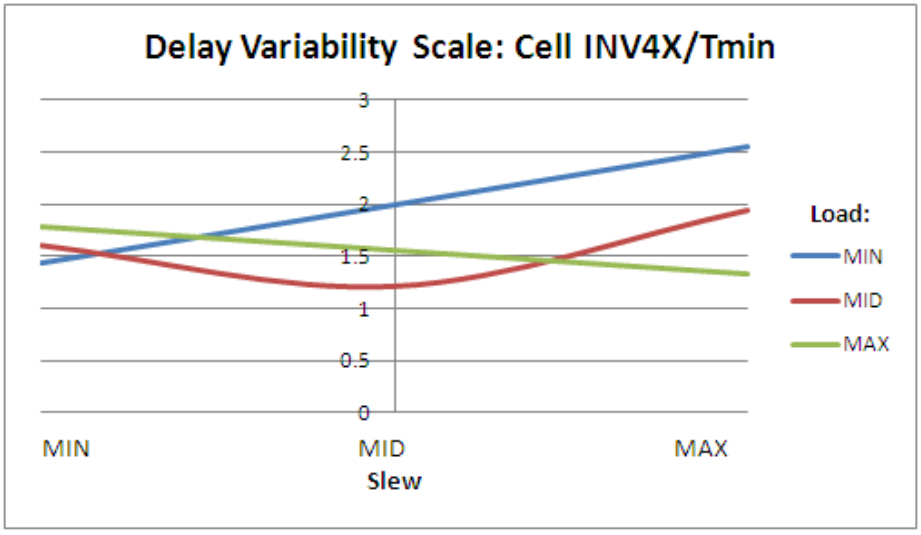

Figure 19 Delay Variability Scale for INV4x/MAX Temp

One can see from the two figures above:

- Strange not monotonic behavior of M as a function of C & R
- There are no scales M < 1
- Figure below illustrates theoretically modeled behavior of M as a function of C & R

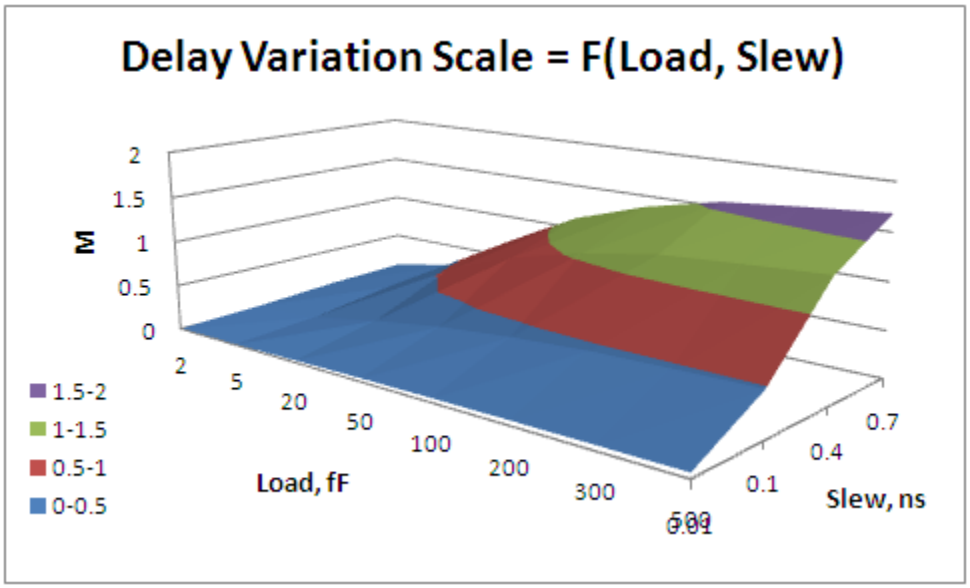

Figure 20 Modeling behavior of M as a function of C & R

If our understanding of POCV is correct (based on [4, 5]), then the final cell local variation may reach 175% (25%*7= 175% if v=25% and 20-25% cell variations are quite reasonable values) and this final variation is definitely too conservative. It would be a more fair method to apply M scale (or M_C & M_R scales) only to "portion" of v corresponding to C- & R-variation contribution to v. But in this case PrimeTime/POCV would need more input information than single v value and LFT tables.

Because M≥1 always, POCV never reduces variation v even when cell (C, R) less than average (Co, Ro) used to obtain v and it would result in pessimism. Conclusions on POCV:

- POCV with scale M=1 (no LFT tables provided) may produce inaccurate derates
- POCV with LFT tables may produce too conservative derates

Conclusions on Design Tips:

- Try to reduce load C to ~Co and slew R to ~Ro, where Co & Ro are the average load and slew that were used to characterize total variation v)
- Run PrimeTime/POCV without LFT tables: M=1 and derating may be accurate enough (within POCV limitations) due to C ≈ Co and slew R ≈ Ro
- Do not spend a lot of time and extra buffers to get min C & R, because POCV will give you no credit for it

5. References

[1] TETELBAUM, A. "Corner-based Timing Signoff and What Is Next": http://abelite-da.com/wp-content/uploads/2012/01/Corner_based_Signoff_paper_Jan_2014a.pdf

[2] KATZ, Isadore, CLK-DA Inc: http://www.deepchip.com/items/0534-03.html

[3] WU, S.H., TETELBAUM, A., WANG, L.C. "How Does Inversed Temperature Dependence Affect Timing Sign-off", in Proc. of 2007 International Conf. on Integrated Circuits Design and Technology, Minatec Grenoble, France, June 2nd – June 4th, 2008, pp. 297-300.

[4] PrimeTime and PrimeTime SI User Guide (Solvnet.com)

[5] GLOBALFOUNDRIES and Synopsys Share Signoff Best Practices For 20/14-nm, 2013 (Solvnet.com)