Buffer Insertion Considering Both Inter-Die and Intra-Die Process Variations

Jinjun Xiong

1. BUFFER VARIATION CHARACTERISTICS

1.1 First-Order Linear Approximation

We characterize a buffer in terms of its gate capacitance (C_b), intrinsic delay (T_b) and output resistance (R_b). Due to process variations, these values will no longer be a fixed value. To simplify the model of process variation effects on buffer characteristics, we lump all variations effects into gate capacitance variation and gate intrinsic delay variation with the output resistance as a constant that is a function of gate sizing only.

In general, devices characteristics are complicated (nonlinear) functions of the underlying physical parameters and sometimes are even hard to described in a closed form. Therefore, to model the devices characteristics variations in the of presence of process variations, we resort to first-order approximation. The rational is that if the underlying parametric variations is small, the nonlinear relationship can be reasonably captured by a first-order approximation. Mathematically it can be described as:

$$C_b = C_{b0} + \sum_{i \in \mathcal{I}} \alpha_i \cdot X_i, \tag{1}$$

$$T_b = T_{b0} + \sum_{i \in \mathcal{I}} \beta_i \cdot X_i, \tag{2}$$

where C_{b0} and T_{b0} are nominal values of C_b and T_b, respectively; and X_i are the underlying parametric variations such as channel length, doping density, and gate oxide thickness. The coefficients α_i and β_i are sensitivity of C_b and T_b to the variation of X_i, respectively.

To verify the accuracy of the above first-order modeling, we run SPICE simulations. For illustration purpose and also because of the lack of access to the real sources of foundry process variations, we only model the random L_{eff} variation using $65nm$ BSIM model in this section. Moreover, we assume the variation of L_{eff} to be a symmetric normal distribution to represent the common wisdom regarding to process variations.

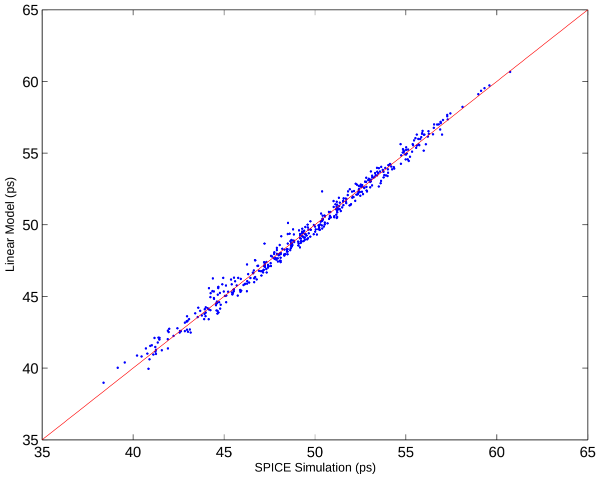

Figure 1: SPICE extracted buffer intrinsic delay versus linear model predication.

In Figure 1 and 2, we show the experiment results for a two-stage buffer with size equal to 128× of the minimum size buffer. The standard deviation of L_{eff} variation is set as 10% of the mean value in our experiment. We run SPICE simulation and extract the characteristics of the buffer when different L_{eff} values are assumed. We then use a least square error curve-fitting technique to obtain (1) and (2). Figure 1 shows the SPICE extracted T_b versus the linear model predicted T_b, while Figure 2 shows the SPICE extracted C_b versus the linear model predicated C_b. According to the figures, it clearly shows that the first-order models for T_b and C_b are very accurate and the largest relative error is less than 5%.

1.2 Normal Distribution Approximation

Because of the nonlinear relationship between parametric variations (like channel length, doping density, gate oxide thickness, etc.) and the device characteristics, the latter's distributions are unlikely to be normal even if the underlying parametric variations are assumed to be normal. However, just as we have discussed above, if the underlying parametric variations is assumed to be small, the nonlinear relationship can be approximated by a first-order linear equation. Therefore, the device characteristics can also be approximated by a normal distribution.

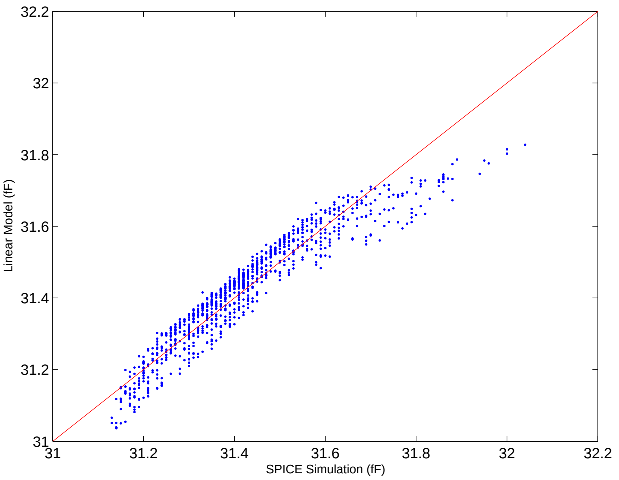

Figure 2: SPICE extracted buffer input capacitance versus linear model predication.

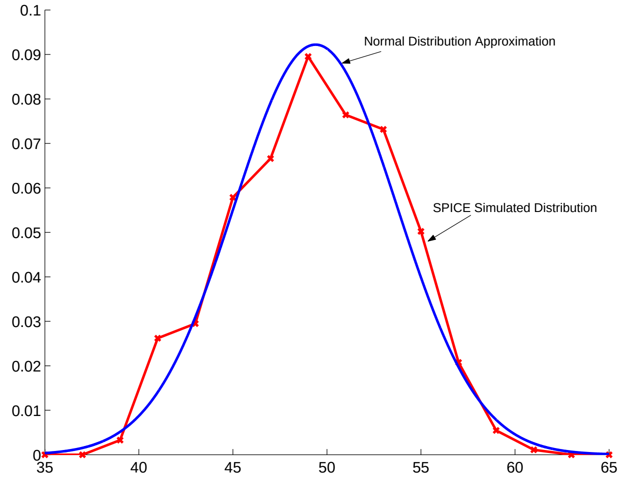

Figure 3: Normal distribution approximation of T_b.

We validate this argument via Monte Carlo simulation. During each run, we use SPICE to extract the device characteristics as discussed above. Figure 3 shows the PDFs of T_b from both Monte Carlo simulation and the normal distribution approximation. It clearly shows that normal distribution is a reasonably good approximation of the real distribution as the two PDFs are very close to each other.

To see how good the normal distribution approximation is, we further compute the accumulated probability errors sampled at some interesting quantiles (like 1%-tile, 5%-tile, 10%-tile, 50%-tile, 90%-tile, 95%-tile, and 99%-tile) from the two CDFs, and show the results in Figure 4. According to Figure 4, we find that the normal approximation incurs no more than 3% error when compared to the Monte Carlo simulation. This further validated our assumption on the first-order approximation of device characteristics. Otherwise, the distribution would very unlikely exhibit a normal distribution if the relation is nonlinear.

2. PROCESS VARIATION MODELING

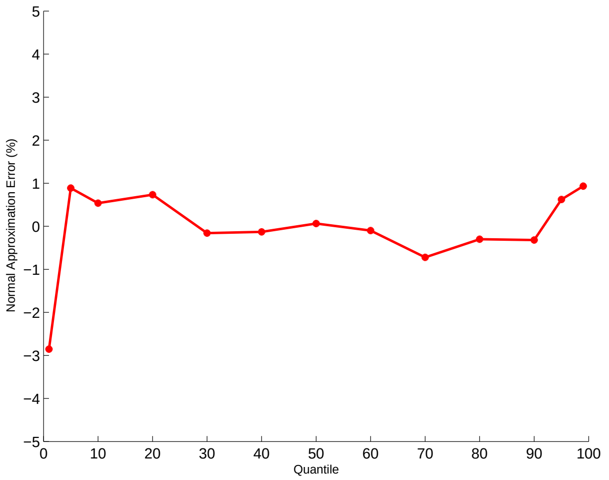

Figure 4: Normal distribution approximation errors measured at some interested quantile points.

2.1 Intra-die Variation

To capture the effect of intra-die spatial variations on device, the characteristics of a device located at a particular region will be affected by its nearby regions variations. In another words, two devices that are physically close should have a higher correlation than two devices that are physically far apart.

We partition the chip area into different regions, and associate each region with two independent random variables X_c and X_d, where X_c models process variation induced device gate capacitance variation; and X_d models process variation induced device intrinsic delay variation. In general, X_c and X_d are correlated within the same region as the underly physical parameters for both gate capacitance and device intrinsic delay are the same. However, as we can always employ the PCA technique to decouple the two correlated terms into two independent terms, for brevity, we will assume X_c and X_d are independent in the following.

For a device located at a particular region R_t with sizing fixed, its output resistance $R_{b,t}$ are constant, input capacitance $C_{b,t}$ and intrinsic delay $T_{b,t}$ are two random variables given by:

$$C_{b,t} = C_{b0} + \sum_{i \in \mathcal{I}_t} \alpha_{t,i} \cdot X_{c,i}, \tag{3}$$

$$T_{b,t} = T_{b0} + \sum_{i \in \mathcal{I}_t} \beta_{t,i} \cdot X_{d,i}. \tag{4}$$

The coefficients $\alpha_{t,i}$ and $\beta_{t,i}$ are two sets of constant numbers that are associated with region R_t. The index set $\mathcal{I}_t$ defines the set of regions that spatial correlations should be considered for devices located at R_t, while the values of coefficient $\alpha_{t,i}$ and $\beta_{t,i}$ determine how strong the correlation is. In general, the larger the coefficients, the larger the correlation.

Because the index set $\mathcal{I}_t$, and the coefficients $\alpha_{t,i}$ and $\beta_{t,i}$ are region-dependent and different regions will have different $\mathcal{I}_t$, $\alpha_{t,i}$, and $\beta_{t,i}$. By properly setting up these values, we can easily capture the spatial correlations between devices at different regions. For example, for two devices located at two nearby regions, they will share more common correlated regions as decided by their common indexes in $\mathcal{I}_{t1}$ and $\mathcal{I}_{t2}$, and have larger corresponding coefficients.

For example, given an example here.

2.2 Inter-die Variation

As the inter-die variation affects all devices within the same die in a similar way, we can model this variation by introducing another two independent random variables, X_{gc} and X_{gd} and modify (3) and (4) as follows:

$$C_{b,t} = C_{b0} + \sum_{i \in \mathcal{I}_t} \alpha_{t,i} \cdot X_{c,i} + \alpha_{t,0} \cdot X_{gc}, \quad (5)$$

$$T_{b,t} = T_{b0} + \sum_{i \in \mathcal{I}_t} \beta_{t,i} \cdot X_{d,i} + \beta_{t,0} \cdot X_{gd}. \quad (6)$$

The coefficients of $\alpha_{t,0}$ and $\beta_{t,0}$ can be different for different regions so that the difference in effects of inter-die variations on devices can be also captured in (5) and (6).

2.3 Layout Dependent Variation

In deep sub-micron design regime, crosstalk-induced timing variation is becoming more prominent. Because this type of timing variation is not only functional dependent, but also layout dependent, it is very hard to model it deterministically.

Therefore, we model this timing variation via a layout dependent random variable. We associate each routing region R_t with a random variable $X_{w,t}$, which models interconnect crosstalk (noise) induced wiring delay variation. In another words, for wires that are routed through R_t, we would expect additional timing delay given by $\gamma_t \cdot X_{w,t}$, where γ_t is a routing region dependent coefficient to capture how good the layout is immune to crosstalk effect. For a layout with less crosstalk effects, γ_t should be smaller than a layout with large crosstalk effects.

3. BUFFER INSERTION CONSIDERING PROCESS VARIATIONS

3.1 Preliminary

For simplicity of presentation, we follow the same argument as [1] by assuming that the routing tree is given as a binary routing tree and the *legal* buffer positions (nodes) are directly after the branching points of the tree[1]. For a given buffered routing tree, we associate every legal buffer position t in the tree with two numbers: the *input loading capacitance* (or *downstream loading capacitance*) L_t and the *required arrival time* T_t. Denote c and r as interconnect's *unit length capacitance* and *sheet resistance*, respectively, we model each interconnect segment in the routing tree with length l_t as a π model, where the resistance is given by $r \times l_t$, and the capacitance is given by $c \times l_t$.

Under the Elmore delay model, the L_t and T_t can be computed as follows.

If node t is obtained by adding a wire of length l_i at its direct downstream node n, then

$$\begin{aligned} L_t &= L_n + c \cdot l_i & (7) \\ T_t &= T_n - r \cdot l_i \cdot L_n - \frac{1}{2} \cdot r \cdot c \cdot l_i^2. & (8) \end{aligned}$$

If node t is obtained by adding a buffer at its direct downstream node n, then

$$\begin{aligned} L_t &= C_b & (9) \\ T_t &= T_n - T_b - R_b \cdot L_n. & (10) \end{aligned}$$

If node t is obtained by merging two nodes m and n, then

$$\begin{aligned} L_t &= L_n + L_m & (11) \\ T_t &= min(T_n, T_m). & (12) \end{aligned}$$

It has been proved in [1] that the buffer insertion problem, without considering process variation, can be solved optimally via dynamic programming. Moreover, by properly define the *dominance relationship* (or *pruning rule*) between two solutions, i.e., solution (L_1, T_1) dominates solution (L_2, T_2) if condition $L_1 < L_2$ and $T_1 > T_2$ are satisfied, [1] proved that by keeping only dominating solutions at every node, the dynamic programming approach can solve the problem in polynomial time without loosing optimality.

4. REFERENCES

[1] L. P. P. P. van Ginneken, "Buffer placement in distributed RC-tree networks for minimal Elmore delay," in *Proc. IEEE Int. Symp. on Circuits and Systems*, pp. 865–868, 1990.

[1] Note that the methodology to be presented in this work does not depend on these assumptions.