Efficient Bottom-up Approaches to Build Variation-aware PLL Behavioral Models

Student: Chin-Cheng Kuo
Advisor: Chien-Nan Jimmy Liu
Department of Electrical Engineering National Central University, Taiwan, ROC
E-mail: casey@ee.ncu.edu.tw, TEL: +886-3-4277092, FAX: +886-3-4255830

In SOC era, analog and mixed-signal (AMS) designs are very popular in ASIC applications. The correct integration between digital and analog designs becomes a serious problem. In traditional design flow, these two parts are assembled at layout level that requires transistor-level simulation to check whole chip performance. For modern large designs, low-level simulations and bug-fixing iterations are timing consuming and become almost infeasible. One of the popular ways to solve these integration issues is to build the behavioral models for both digital and analog circuits and perform system simulation at behavioral level. With those behavioral models, whole chip simulation can be performed quickly to verify the integrated system at earlier design stages and reduce the design iterations.

Using phase-locked loop (PLL) circuits as a study case, an efficient bottom-up extraction approach [1-2] is first presented in this thesis to generate accurate behavioral models of PLL designs efficiently. The main idea is to use a special “characterization mode”, as shown in Fig. 1, to break PLL loop instead of separating the PLL circuit into independent sub-blocks. Therefore, tedious post-layout netlist tracing is avoided and correct loading and parasitic effects can be considered automatically. In this mode, the extracting pattern can be easily controlled to trigger the PLL into different states and acquire critical circuit parameters much faster. Since all parameters in our behavioral models can be measured at the PLL outputs without simulating each block separately, this approach is more suitable to accurately model protected IPs (Intelligent Property) or flattened post-layout netlists.

In this bottom-up behavioral model, the non-ideal properties from PLL circuit itself can be accurately dealt as demonstrated in the experimental results. However, the noises from other circuits also have large impacts on the PLL performance. Therefore, a modified behavioral modeling approach [3-4] is presented in the second part of this thesis to reflect real-time supply noise effects in our PLL behavioral model. Instead of modeling final circuit performance directly as a single complicated equation, the relationship between supply noise and the intermediate parameters in our behavioral models are built in a simple form, as illustrated in Fig. 2. Those parameters can be dynamically adjusted according to the present supply noise induced by other circuits. This approach can accurately handle irregular noise with simple models and still have efficient simulation time, as experimental results show.

Previous analog behavior modeling approaches often treat the noisy V_{DD} waveform as a given input and focus on reflecting such stimuli on circuit performance. However, because the interaction of noise aggressors and victims is not considered, some errors may exist in the simulation. In this thesis, a simple SCORE (state-controlled-resistors) macro-model for PLL designs [5] is presented in the third part. It can be integrated with an arbitrary supply-noise-aware PLL behavioral model to analyze supply noise effects at high level, as illustrated in Fig. 3. In addition to numerical results, the time-varying supply noise waveform and real-time PLL responses can be obtained simultaneously. As demonstrated in the experimental results, the proposed approach can provide more realistic simulation results with noise interaction effects but still keep fast simulation time.

With shrinking device sizes in VLSI technology, device parameter variations have become the major factor limiting circuit performance. Compared with digital circuits, analog circuits are much more sensitive to process variation. Therefore, in the fourth part of this thesis, the approach [6-7] to reflect the process variation effects in our PLL behavioral model is presented, as illustrated in Fig. 4. Without blind regressions, only one input pattern in the extraction mode sufficiently obtains all required parameters in the behavioral model. A quasi-SA approach is also proposed to accurately reflect process variation effects. Considering generic circuit behaviors, the quasi-SA approach saves considerable simulation time for complicated curve fitting but still keeps estimation accuracy. The experimental results demonstrate that the proposed bottom-up modeling flow and quasi-SA equations provide similar accuracy as in the RSM approach, using less extraction cost as in traditional sensitivity analysis approach. Using these pre-characterized quasi-SA equations in the PLL model, Behavioral Monte Carlo Simulation (BMCS) can be performed to fast estimate performance shift under process variation with detailed circuit responses. It can save considerable simulation efforts to check the design yield under process variation.

In summary, this thesis presents an efficient bottom-up modeling approach for PLL circuits. Considering the variation sources in real environment, such as supply noise, noise interaction, and process variation, different characterization equations are built to extend this behavioral model to be variation-aware. This accurate behavioral model can verify PLL performance under noisy environment quickly. In the future, we will try to find out more applications of PLL behavioral models, such as the DFY (design for yield) topics.
Fig. 1. Proposed characterization mode of PLL

Fig. 2. Proposed flow to generated noise-aware behavioral models

Fig. 3. Proposed platform with parasitic power supply line

Fig. 4. Proposed quasi-SA approach and bottom-up BMCS

PUBLICATION LIST


Fast Statistical Analysis of Process Variation Effects Using Accurate PLL Behavioral Models

Chin-Cheng Kuo, Student Member, IEEE, Meng-Jung Lee, Chien-Nan Jimmy Liu, Member, IEEE, and Ching-Ji Huang

Abstract—Using the behavioral model of a circuit to perform Behavioral Monte Carlo Simulation (BMCS) is a fast approach to estimate performance shift under process variation with detailed circuit responses. However, accurate Monte Carlo analysis results are difficult to obtain if the behavioral model is not accurate enough. Therefore, this paper proposes to use an efficient bottom-up approach to generate accurate process-variation-aware behavioral models of CPPLL circuits. Without blind regressions, only one input pattern in the extraction mode sufficiently obtains all required parameters in the behavioral model. A quasi-SA approach is also proposed to accurately reflect process variation effects. Considering generic circuit behaviors, the quasi-SA approach saves considerable simulation time for process variation effects. Considering generic circuit behaviors, modeling flow and quasi-SA equations provide similar accuracy as complicated curve fitting but still keeps estimation accuracy. The experimental results demonstrate that the proposed bottom-up modeling flow and quasi-SA equations provide similar accuracy as in the RSM approach, using less extraction cost as in the traditional sensitivity analysis approach.

Index Terms—process variation, behavioral model, Monte Carlo simulation, quasi-SA.

I. INTRODUCTION

Device parameter variations have become the major factor limiting circuit performance with shrinking device sizes in VLSI technology. Many digital circuit techniques have been proposed to deal with this effect, such as the well-known statistical static timing analysis (SSTA). The key concept is to compute circuit properties under process variation, such as path delay, interconnect delay and slew analysis, using mathematical analysis or regression equations [1-3, 30-31]. Results have shown that such variation-aware techniques help alleviate process variation impacts.

Analog circuits are much more sensitive to process variation, compared with digital circuits. Monte Carlo (MC) analysis is a common approach analyzing process variation impact to analog circuits. Many device variation samples are randomly generated to perform transistor-level simulations and observe the corresponding output performance shift, according to the statistical models of transistor parameters, such as transistor width (W), channel length (L), threshold voltage (Vth), thickness of oxide (Tox), etc. This flatten MC approach accurately estimates the statistical results of process variation effects if sample numbers are large enough. However, repeating the simulation hundreds of times is often too expensive for modern large designs [4]. Although symbolic analysis [5] was proposed to estimate the statistical results much faster, there is no evidence that such approach is still applicable on other complicated designs, such as PLL circuits.

The hierarchical statistical analysis [6-11] is a popular way to solve the speed issue of traditional MC analysis. This approach often uses regression equations, such as prevalent response surface methodology (RSM) techniques [6, 9-11], to replace the whole circuit netlist in statistical analysis. System-level performance of analog circuits is difficult to model as a device variation function directly, therefore the regression process is often divided into two levels as illustrated in Fig. 1. The device-level variation models obtained from IC foundries can be used to form regression equations representing variation values of some intermediate-level circuit properties, such as timing, current, and frequency information. Different regression equations are then made to model system-level circuit performance under process variation, such as jitter and locking time of a Phase Lock Loop (PLL) circuit, according to the variation of those intermediate-level circuit properties.

Fig. 1. Hierarchical statistical analysis flow
Regression-based approaches can only provide certain statistical numbers, since circuit performance has been modeled as a parameter variation function. Detailed circuit response information, such as a PLL locking waveform under process variation, cannot be provided to designers to improve their circuits if necessary. Besides poor observability, building a regression equation for circuit performance requires many samples of system-level information, implying numerous time-consuming system simulations.

Therefore, the intermediate-level parameters in previous approaches [6-9], are used to build a corresponding behavioral model. Then, the Behavioral Monte Carlo Simulation (BMCS) is performed to generate corresponding output waveforms and estimate performance shift under process variation. Monte Carlo analysis results are obtained in a short time with detailed circuit behaviors due to very fast behavioral simulation. However, behavioral model accuracy is the most critical issue in BMCS-based approaches. Accurate MC analysis results are difficult to obtain if the behavioral model is not accurate enough, even if high-order regression equations are used to reflect process variation effects.

This paper proposes an efficient BMCS approach to analyze PLL designs under process variation. An efficient bottom-up approach is used to generate accurate behavioral models, which deal with the complicated relationship between intermediate-level and system-level. The key idea is using a special “characterization mode” to extract the required circuit parameters. Only one input pattern in the extraction mode sufficiently obtains actual circuit properties with parasitic and loading effects. Using such an accurate modeling approach, the comprehensive system performance analysis can be obtained accurately without blind regressions.

Device-level to intermediate-level models in typical RSM approaches still require considerable simulation data to do regression. Digital behaviors, such as delay time, can be well modeled by the sensitivity analysis (SA) approach as demonstrated in previous researches [15-19]. Using SA methodology greatly reduces modeling cost. However, linear SA may induce significant errors in estimating behavior variations of analog blocks, such as the charge pump (CP) and voltage-controlled oscillator (VCO) block in a PLL circuit. Therefore, this paper proposes a Quasi-SA approach to estimate analog behavioral parameters under process variation. This structure-independent approach uses only generic PLL behaviors and keeps traditional SA efficiency, but increases modeling accuracy for analog blocks. After the sensitivity functions are obtained, our behavioral model can be dynamically adjusted to perform fast behavioral simulation with process variation effects when device variation values are randomly generated in MC analysis, as illustrated in Fig. 2.

The remainder of this paper is organized as follows. Hierarchical statistical analysis is described in Section II. Our efficient bottom-up PLL behavioral model is introduced in Section III. The strategy for extending the PLL model to handle process variation is explained in Section IV. Section V uses experiments to demonstrate the efficiency and accuracy of our BMCS approach to deal with process variation. Finally, conclusions are drawn in Section VI.

II. HIERARCHICAL STATISTICAL ANALYSIS

Hierarchical statistical analysis is often used to improve a flattened MC simulation speed. Hierarchically separating the analysis process into three levels can avoid solving the whole flatten netlist. Variation sources of transistor parameters are classified as device level. Electrical properties of each sub-circuit are at intermediate level, such as timing information and current values. The key step in hierarchical analysis is building suitable equations to model the relationships of any adjacent two levels.

A. Device Level to Intermediate Level

In this work, four transistor parameters, $\Delta W$, $\Delta L$, $\Delta V_t$ and $\Delta T_{off}$, are chosen as the MC analysis random variables because they are considered to have more performance shift contributions according to previous researches [12-13]. This work uses the same variation models as in the SPICE MC model provided by IC foundry during our experiments, to make the experiments more realistic. These four chosen parameters in the provided MC model are independently described by different random generators. Therefore, first order Taylor series expansions can be used to approximate circuit property values under process variation [12-15] like timing delay, current values, frequency responses, etc., as shown in (1).

$$B = a_0 + \sum_{j=1}^{n} a_j x_j$$

$B$ are expressed as the functions of device parameters $x_j$ to represent behavioral parameter values. And $n$ is the number of considered variation parameters. The constants, $a_0$ and $a_j$, are unknown coefficients to be estimated. In deep-submicron process, some parameter variations may have correlation to other parameters. For example, the well-known short channel effects will make $V_t$ become a function of $L_{eff}$. Supposedly, such effects have been included in foundry’s MC models because those values in the models are measured after manufacturing. Therefore, most papers still use
independent random variables to generate the effective \( V \) value and other transistor parameters without losing too much accuracy. If foundry’s models are not accurate enough to capture the correlation between parameters in the future, correlated random generators \([14, 32-33]\) can be used instead to generate reasonable combinations of transistor parameters. Then, equation (1) is still applicable based on those parameter sets. Because equation (1) reflects the relationship between performance changes and effective parameter changes, the prediction will still be accurate if the effective parameter values are accurate.

A conventional approach to obtain those coefficients in (1) uses Response Surface Methodology (RSM) techniques \([6, 9-11]\). The Taylor series expansions in (1) are the same form as in the 1\(^{st}\) order RSM approach such that these coefficient values can be found out by RSM techniques. However, accurately fitting regression equations requires considerable training samples, implying the numerous transistor-level simulations to consider device variation effects. The number of training samples should be about four times greater than the number of unknown coefficients from the summary in realistic experimental results \([10]\). Therefore, the number of training samples is at least \(4(n+1)\) for the 1\(^{st}\) order RSM to regress (1).

Sensitivity analysis (SA) is another intuitive methodology to estimate unknown coefficients for the relationships of behavioral parameters under process variation. Considering small variations of device parameters \( \Delta x' \), circuit behaviors will vary from their nominal values \( B_0 \) and the sensitivity \( S_{E_{E}} \) of behavioral parameters to process variations can be obtained, as shown in (2).

\[
S_{E_{E}} = \frac{\partial E}{\partial x} = \frac{\Delta B}{\Delta x'} \Rightarrow B = B_0 + \sum_i S_{E_{E}} \Delta x_i \tag{2}
\]

where \( \Delta B \) is the behavior variations under given device variation \( \Delta x' \). Then, \( S_{E_{E}} \) can be directly extracted through HSPICE simulation with constant device parameter variations so that the number of training samples is only \((n+1)\).

Previous researches \([15-19]\) apply linear sensitivity to show its effectiveness with low extracting cost, especially for modeling statistical gate delay variation. Even if variability is about eight percent to the nominal value, path delay variation is still approximately a linear equation with process variation \([19]\). Therefore, the SA method is useful to model timing parameter variations. In \([9]\), 2\(^{nd}\) order quadratic RSM techniques can be applied to link these two levels and express circuit performance \( P \) as a function of behavioral parameters, \( B_i \) and \( B_j \), shown in (3).

\[
P = c_0 + \sum_{i=1}^{k} c_i B_i + \sum_{j=1}^{k} \sum_{j=1}^{k} c_{ij} B_i B_j \tag{3}
\]

The constants, \( c_0, c_i \) and \( c_{ij} \), are the regression coefficients. \( k \) is the amount of considered behavioral parameters.

Such high order equations provide more accurate results, but require much more regression cost. The number of training samples increases exponentially to find out the unknown coefficients. In (3), the number of coefficients equals to \(1+(k^2-3k)/2\) such that the number of training samples is at least \(4[1+(k^2-3k)/2]\) even using some techniques \([10]\) to reduce regression complexity. Moreover, choosing sufficient regressors is the other serious problem because considering all behavioral parameters requires an extremely large amount of simulation results for curve-fitting. The simulation cost for each sample also increases exponentially because the whole system simulation at transistor level is very time-consuming, unlike the training samples for the equations from device level to intermediate level.

However, final statistical results of system performance shift are still not accurate enough even using expensive second-order RSM polynomial functions \([9]\). Actually, system performance, such as PLL output jitters, are difficult to be expressed as polynomial functions of behavioral parameters \([20]\), implying that the RSM-based approach may not be suitable in the PLL case. Moreover, since circuit performance has been modeled as a mathematical equation, only some statistical numbers can be provided in the analysis. Detailed information of circuit responses, such as the locking waveform of a PLL under process variation, cannot be provided to designers to improve their circuits if necessary.

In fact, the bridge from behavioral level to system level has been built by the behavioral model in the early design stage. This kind of top-down model for a new analog design can be established using Simulink/Matlab or analog hardware description language (AHDL) describing circuit behavior with mathematic formulas. Using such behavioral models, a fast check of design functionality can be performed in advance. Using the PLL model as an example, simulation waveforms of the lock-in process and output frequency can estimate lock-in time and jitter values. These detailed circuit responses are important information for circuit designers.
Although the top-down behavioral modeling approach avoids the luxurious regression cost with better observability, those modeling results are not accurate enough for hierarchical statistical analysis due to lack of actual circuit-level information. Since behavioral model accuracy directly affects the final statistical results, a bottom-up strategy with back-annotation of realistic circuit properties is necessary to improve the modeling accuracy.

This paper uses an efficient bottom-up characterization flow for charge-pump PLL (CPPLL) to extract actual circuit properties as intermediate-level parameters in our behavioral model. For \( n \) device variations, only \( n+1 \) runs of parameter extraction is enough to build the pre-characterized equation for each behavioral parameter such that its corresponding value can be dynamically adjusted during the MC analysis. Overall system responses can be analyzed accurately in fast behavioral-level simulation because non-ideal effects are considered automatically. Then, bottom-up Behavioral Monte Carlo Simulation (BMCS) can be performed in hierarchical statistical analysis to generate the corresponding output waveforms and estimate performance shift under process variation as shown in Fig. 3. As shown in the experimental results, this approach provides more accurate statistical results than the pure regression-based approach [9], and also includes detailed waveforms with less regression costs.

This section briefly introduces the bottom-up extraction flow to generate accurate behavioral models for prevalent CPPLL designs. Any behavioral modeling approach with enough accuracy, such as the models in [21–25], might be used in the BMCS step of hierarchical statistical analysis. For easier integration, we adopt our previous work [21] to generate the required behavioral models of CPPLL designs. The key concept is to develop a special “characterization mode” to extract required circuit parameters from circuit behaviors and back-annotate to the behavioral model. The PLL design under extraction in this mode is not operated in a normal situation, which helps obtain the required parameters faster. In this way, time-consuming correlation analysis can be avoided for building such accurate models. Previous experimental results show that this bottom-up extraction flow can efficiently and accurately analyze PLL performance. Therefore, we try to extend this efficient modeling approach to handle process variation and use resultant models in the BMCS step.

A. Characterization Mode

In the proposed characterization mode, the PLL loop is broken without separating into independent blocks as shown in Fig. 4. The broken connection helps send special patterns to quickly trigger the PLL into different situations. Moreover, simulating all PLL blocks together allows automatic parasitic and loading effects consideration. This methodology is more suitable for existing IP to avoid tedious layout-tracing steps.

Only one input pattern in this mode, as shown in Fig. 4, can trigger the PLL design and extract all required characteristic parameters from simulation results. Major factors affecting PLL performance include timing information of the phase frequency detector (PFD) and frequency divider (FD), current mismatch information and equivalent switch on/off time of the charge pump (CP) and loop filter (LF), and frequency information of the voltage-controlled oscillator (VCO), etc. These factors can be obtained using this approach without detailed circuit structure and device size information.

B. PFD & FD

The PFD circuit is treated as a three-state machine to detect the phase shift. And the FD function divides the frequency. These two blocks are often treated as digital blocks. Timing information, such as delay time, reset time, and transition time, is the major concern of PLL designers. These characteristic parameters are also the primary sources of non-ideal effects, such as the PFD dead zone, and contribute to PLL performance. Timing parameters in our approach can be easily measured from the simulation results in the characterization mode, as shown in Fig. 5.
C. CP & LF

The common CP structure with a second-order low-pass filter (LPF) is shown in Fig. 6 as an example. The transfer function in (4), including the information of $I_{cp}$ and RC values, is described as an S-domain function in our behavior model. Exact internal current values are difficult to observe in protected IP and existing flattened designs. Therefore, a voltage-domain behavioral model is proposed to combine CP and LF and measure voltage changes instead of actual current values. The parameters are extracted from output responses under our extraction pattern, regardless of the implemented structure of bias circuit and current mirror. This approach automatically considers parasitic and loading effects without actual values of equivalent RC components or pole/zero positions.

$$V_{cvi} = I_{cp}Z_{LF} = \frac{s(I_{cp}R) + I_{cp}}{s^2 \left( \frac{RC_1}{\beta} \right) + s \left( 1 + \frac{1}{\beta} \right)} \beta = \frac{C_1}{C_2}$$ (4)

The extraction pattern propagated from PFD is shown in Fig. 6. The positive phase error makes the “Up” signal of PFD remain at logic high so that the source current of CP ($I_{cp}$) charges $V_{cvi}$. Observing the responses of $V_{cvi}$ waveform can extract approximate source current value and equivalent output impedance information. The zero-phase-error operation mimics the lock-in situation. Even when the PLL is locked, the non-ideal effect of CP, such as current mismatch ($\text{Mis} = |I_{cp} - I_{bl}|$) and non-zero switching on/off time, charges or discharges $V_{cvi}$ a little bit. In this way, realistic circuit behaviors can be captured without detailed circuit analysis from voltage waveform responses. The sink current value ($I_{sn}$) at negative-phase-error operation can at last be obtained from $I_{cp}$ and $\text{Mis}$ without simulation. These behavioral parameters provide the critical contribution to system performance, especially the PLL output jitter.

D. VCO

The critical concerns of VCO circuits are the gain of VCO ($K_{VCO}$), the range of input control voltage, and output frequency. Curve fitting is used in previous approaches [25-26] to model VCO behavior accurately. The sample data must be large enough however, to have accurate coefficients for those fitting equations, which may increase required simulation time for building the VCO model.

Rebuilding the completed VCO response may not be necessary for the trade-off between modeling efficiency and accuracy, because designers often prefer to operate VCO in the linear region as much as possible. Actually, the linear VCO model predicts more than 90 percent of real VCO characteristics, especially in normal operation range, according to related studies [6, 20, 27-28]. Therefore, we adopt the linear VCO model to simplify modeling complexity. The linear relationship is helpful in greatly reducing regressor numbers, while extending process-variation-awareness in VCO model.

In the characterization mode, the CP current charges $V_{cvi}$ up to $V_{DD}$ gradually undergo a long positive-phase-error operation. During the charging process, we can measure any two voltage values on $V_{cvi}$ ($V_1$ and $V_2$) within the operation range and obtain the corresponding output frequencies ($f_1$ and $f_2$) to calculate the $K_{VCO}$ value, as shown in Fig. 7. Then, the frequencies at $V_{cvi} = 0$ and $V_{cvi} = V_{DD}$ are extracted to derive the $V_{min}$ and $V_{max}$ of this VCO according to the relationship of the calculated $K_{VCO}$.

E. Behavioral Model Accuracy

Instead of directly modeling the jitter values as stochastic forms, the major non-ideal properties of each sub-circuit are bottom-up extracted respectively in our approach. Then, accurate circuit responses can be obtained automatically during the behavioral-level simulation, such as the settling process and output jitters. Using a PLL circuit designed with TSMC 0.18um process as an example, the locking waveform of our behavioral model is compared with HSPICE results as shown in Fig. 8 and 9. Fig. 10 and 11 show the peak-to-peak jitter values of PLL output, which demonstrate that our model can also predict correct jitter values.
This behavioral model can also provide the FFT waveforms of the PLL output, as shown in Fig. 12, to observe frequency-domain responses if necessary. In this case, the phase noise error compared to HSPICE results is 0.8dBc/Hz at 10MHz frequency offset. In TABLE I, the locking voltage (V_{lock}), locking time (T_{lock}) and peak-to-peak (pk-pk) jitter are selected as the PLL system characteristics for comparison. All results show that our PLL model can provide similar responses very quickly.

### IV. EXTENDED TO PROCESS-VARIATION-AWARE MODEL

The nominal value of behavioral parameters in our PLL model can be extracted from simulation results, according to the characterization flow in Section III. Next, we explain how to extend this bottom-up behavioral model to be process-variation-aware. A pre-characterized equation is built for each behavioral parameter such that its corresponding value can be dynamically adjusted during the MC analysis. According to the different properties of digital and analog sub-circuits, the SA and quasi-SA approaches are used to build equations for different behavioral parameters.

#### A. Sensitivity Analysis (SA)

The benefit of linear sensitivity analysis is low regression cost. The effectiveness of SA has been demonstrated in previous researches [15-19, 31], especially for modeling statistical gate delay variation. Therefore, the SA method is useful to model timing parameter relationships from device level to intermediate level. These behavioral parameters are modeled as the function of process parameters with independent sensitivity values. Taking the delay time (T_d) as an example, timing change (∆T_d) under process variation can be simply modeled by sensitivity analysis, as shown in (5),

\[
\Delta T_d = T_d(\Delta x_i) - T_{d0} \approx \frac{\partial T_d}{\partial x_i} \times \Delta x_i \quad (5)
\]

where \(T_{d0}\) is the nominal delay without process variation, \(\frac{\partial T_d}{\partial x_i}\) is delay sensitivity to the process parameter \(x_i\).

Only one extra extraction process run is required to consider process-level variations, besides the original extraction process to build the behavioral model without process variation. Then, by comparing each behavioral parameter value under device variation to its nominal value, sensitivity value can be obtained. Taking the delay change under width variation as an example, the relationship using a sensitivity value (\(S_{E,T_d,W}\)) can be modeled as shown in (6).

\[
S_{E,T_d,W} = \frac{\partial T_d}{\partial W} \approx \frac{\Delta T_d}{\Delta W} = \text{constant} \quad (6)
\]

\(\Delta W\) represents the chosen value of width variation in sensitivity analysis. And \(\Delta T_d\) is the corresponding delay variation value.

While performing MC analysis using our behavioral models, the changes of behavioral model parameters can be calculated according to their sensitivity when device variation values are
randomly generated. Then, the individual effect of each parameter variation is sum up to obtain the final value of each behavioral model parameter. Because each device variation is treated as independent in the foundry-provided 0.18um MC model, independent random variables are used in this work to generate the effective transistor parameters. If foundry’s models are not accurate enough to capture the correlation between parameters in more advanced process, correlated random generators can be used instead to generate reasonable combinations of transistor parameters. Four transistor parameters, $\Delta W$, $\Delta L$, $\Delta V$, and $\Delta T_{ox}$, are considered in this paper such that delay time under process variation can be expressed as the equation shown in (7), where $T_{nom}$ is the nominal delay. Since our approach focuses on predicting the performance changes with respect to the given parameter values, the assumption of specific parameter distribution is not required. Any probability distribution can be used in the BMCS approach to obtain accurate statistical results.

As an example, the PFD block delay time in our PLL circuit is chosen to check modeling accuracy and verify sensitivity analysis effectiveness for our timing parameters. Compared with the 100-run HSPICE MC simulation, delay values are calculated by the SA model, like the form in (7), with the same variation values of device parameters. The scatter plots between them are shown in Fig. 13. Although the maximal delay change achieves $\pm 10$ percent, estimated values are still very accurate. The error values of mean and standard deviation are both less than 1 percent.

$$T_d (\Delta W, \Delta L, \Delta V, \Delta T_{ox}) = T_{nom} + \Delta W \times S_{E_{t_{cp}}} + \Delta L \times S_{E_{t_{cp}}} + \Delta V \times S_{E_{t_{cp}}} + \Delta T_{ox} \times S_{E_{t_{cp}}}$$

(7)

![Delay Variations of PFD](image)

Fig. 13. SA model v.s. HSPICE (nominal delay is 447ps)

B. Quasi-SA

Traditional SA can avoid expensive regression cost and accurately predict variation values of timing parameters in the PLL case. However, linear static sensitivity values may induce significant errors in estimating analog block behavior variations, such as the CP and VCO circuit. Considering the generic behavior of CPPLL without detailed circuit structures, a quasi-SA approach is proposed to increase modeling accuracy with the same characterization cost in traditional SA. The next section introduces the proposed strategies for CP and VCO blocks.

1) CP & LF

In our behavioral model, the behaviors of CP and LF are combined together and modeled as an S-domain equation. Therefore, the variation of the resistors and capacitors in LF will be reflected in the changes of the coefficients in (4) during MC simulation. However, since the current change of CP under device variation is not a linear function, the coefficient changes cannot be accurately predicted by linear sensitivity. In this paper, a quasi-SA approach is proposed to alleviate this problem.

In our quasi-SA approach, a variable $\text{ratio}$ is defined as the changed current ($I_{cp}'$) to the ideal current ($I_{cp}$), in order to reflect process variation effects. Using the traditional $S_k$ analysis, $\text{ratio}$ would be expressed as a pure linear function like (7). However, actual current variation ratio is not a linear relationship with threshold voltage change, especially with large fluctuation. The 2nd order RSM can be used as a black-box model to represent this relationship. But blind regression increases simulation cost exponentially. The other approach analyzes internal CP circuit structure. Considering the detailed conditions of the bias circuit and current mirror, the $\Delta V$ effects on current change can be calculated. However, this structure-dependent approach is not general for different CP circuits.

The proposed approach would like to find an approximated model according to only generic circuit behavior, not detailed analysis of the CP circuit. Therefore, a quasi-SA model is proposed to improve traditional SA accuracy without extra effort in order to reduce blind regression cost. A single MOS saturation current ($I_{0s}$) is used as an example to observe the relationship between current variation ratio and $\Delta V$, as shown in (8). Since $k$ is a constant value, the $\text{ratio}$ is a 2nd order function of $\Delta V$, if $k$ is known. In this work, this 2nd order form is used as the quasi-SA model instead of a linear function to consider the threshold voltage variation, which requires only two simulation samples.

$$\text{ratio}(\Delta V) = \frac{I_{0s}' - I_{0s}}{I_{0s}} \geq \left[ \frac{V_{GS} - (V_t + \Delta V)}{V_{GS} - V_t} \right]^2 = \left( \frac{V_{GS} - V_t - \Delta V}{V_{GS} - V_t} \right)^2$$

(8)

$$\left( 1 - \frac{\Delta V}{V_{GS} - V_t} \right)^2 = \left( 1 - \frac{\Delta V}{k} \right)^2 \equiv \frac{I_{cp}' - I_{cp}}{I_{cp}}$$

A simple experiment is performed using our PLL circuit to observe the current variation ratio of CP circuit under different $V_t$. After extracting the $S_k$ value for the SA model and the $k$ value for our approach, the predicted results of traditional $S_k$ analysis and our quasi-SA model are compared with HSPICE simulation, as shown in Fig. 14. When the absolute value of $\Delta V$ is rising, our quasi-SA model still estimates behavior accurately, even if the current doubles nominal value. It shows that our quasi-SA can effectively improve the accuracy without extra cost.
As to the other three device-level parameters, $\Delta W$, $\Delta L$, and $\Delta T_{\text{ox}}$, linear sensitivity models still accurately model current variation ratio information because of their linear relationships to the MOS current. Therefore, the ratio parameter in our process-variation-aware CP model can be expressed as the equation shown in (9). When each set of device variation values is randomly generated in MC analysis, the current change ratio is dynamically calculated to determine the corresponding value.

$$
\text{ratio}(\Delta W, \Delta L, \Delta V, \Delta T_{\text{ox}}) = \Delta W \times S_{E,\text{ratio-}\Delta W} + \Delta L \times S_{E,\text{ratio-}\Delta L} + \left(1 - \frac{\Delta V}{k}\right) \times \Delta T_{\text{ox}} \times S_{E,\text{ratio-}\Delta T_{\text{ox}}}
$$

(9)

2) VCO

The transfer curve in the behavior-level between input control voltage ($V_{\text{ctrl}}$) and output frequency ($f_{\text{out}}$) is the critical concern for a VCO circuit, especially the segment at operation range. A structure-independent extraction flow in the characterization mode is developed to build up the approximated transfer curve using a piece-wise-linear function. This approach does not use complicated regression equations such that extending this model to be process-variation-aware does not require too many regressors, implying a great reduction in regression cost.

In order to observe those process variation effects on the VCO block, an oscillator in TSMC 0.18um process with an operation range of 0.8V to 1.2V is implemented. Considering different $\Delta L$ as an example, the relationship between $V_{\text{ctrl}}$ and $f_{\text{out}}$ is obtained from HSPICE simulation as shown in Fig. 15. Curves are quite linear in the operation range such that using linear VCO model would not incur too many errors. Traditional sensitivity analysis for such a linear VCO model uses a constant. The discussion of other process variation effects on VCO can be found in [29].

$$
S_{E,f,\Delta L} = \frac{\Delta f}{\Delta L} = \text{constant}
$$

(10)

Actually, the $f_{\text{out}}$ value of a voltage-controlled oscillator is a function of input control voltage whatever the implemented structure. The transfer curve of the VCO circuit suffering from process variation should be influenced by $V_{\text{ctrl}}$ also, not only device fluctuation effects. Therefore, this work also proposes a quasi-SA model for VCO to solve the problem of static frequency sensitivity using generic circuit behavior only.

Another experiment is conducted to observe the effects of $V_{\text{ctrl}}$ values. Three different $V_{\text{ctrl}}$ values, 0V, 0.8V and 1.2V, are arbitrarily chosen to measure the frequency sensitivity $S_{E,f,\Delta L}$ under different $\Delta L$ values. Experimental results displayed in Fig. 15 show that the slopes, representing frequency sensitivity, are quite different in different $V_{\text{ctrl}}$ values. Therefore, integrating both process variation and $V_{\text{ctrl}}$ effects efficiently is the target in our quasi-SA approach.

The VCO transfer curve under process variation can be translated into a piece-wise linear function shown in Fig. 17. Our characterization mode measures frequencies at $V_{\text{ctrl}} = 0$ and $V_{\text{ctrl}} = V_{\text{DD}}$ under a given $\Delta L$ to find different frequency sensitivity values, $S_{E,f,\Delta L}$ and $S_{E,f,\Delta L}$ defined in (11). Moreover, as illustrated in Fig. 17, we extract the other two values $S_{E,f,\Delta L}$ and $S_{E,f,\Delta L}$ defined in (11) to capture dynamic frequency sensitivity. Then, our approach uses the adjusting term defined in (12) to model dynamic frequency sensitivity with both effects from process variation and different $V_{\text{ctrl}}$ as shown in (13). Finally, the phase variation of the VCO output under length variation can be expressed as (14). In the same way, the phase changes under different device-level variations are found and summed as the entire phase variation in (15).
Please note those sensitivity values in our model are obtained from the same extraction pattern in our characterization mode. No extra simulation cost is required.

In order to verify the accuracy of our quasi-SA approach for VCO model, a 100-run Monte Carlo simulation is performed to observe the VCO frequency changes under various process variation. The calculated results using quasi-SA models are compared with the results of HSPICE simulation, traditional SA, and 1st order RSM regression model. Choosing two arbitrary $V_{ctrl}$ values for observation, the frequency errors of the 100 cases are compared in Fig. 18 and 19. The absolute values of maximum error and average error in this 100-run MC analysis are shown in TABLE II. The quasi-SA model provides almost the same results as the 1st order RSM model, both close to transistor-level simulation (HSPICE) results. Traditional SA approach results have more significant errors. Moreover, Fig. 20 shows one of the 100 cases during MC simulation to illustrate the accuracy of frequency estimation during whole $V_{ctrl}$ operation region. This case suffers from larger device variations than the values in the quasi-SA characterized process ($\Delta W_r, \Delta L_r, \Delta V_t, \Delta T_{ox}$), which helps observe the accuracy to deal with the variation far from characterized values. The results also show that the quasi-SA approach is more accurate than traditional SA methodology. These experimental results can demonstrate that the quasi-SA approach effectively improves VCO model accuracy without extra simulation cost for complicated regressions.

![Fig. 17. Proposed variation-aware linear VCO model with $V_{ctrl}$ effects](image)

![Fig. 18. VCO frequency errors of 100-run MC simulation ($V_{ctrl}=1.0\,\text{V}$)](image)

![Fig. 19. VCO frequency errors of 100-run MC simulation ($V_{ctrl}=1.2\,\text{V}$)](image)

![Fig. 20. VCO frequency curve in operation range](image)

![Fig. 20. Variation: $-2.26\Delta W, 1.51\Delta L, 2.82\Delta V_t, 1.73\Delta T_{ox}$](image)

**TABLE II**

<table>
<thead>
<tr>
<th>100-run Monte Carlo analysis</th>
<th>Error of Quasi-SA</th>
<th>Error of Traditional SA</th>
<th>Error of 1st RSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{ctrl}=1.0,\text{V}$</td>
<td>Max. $</td>
<td>\text{error}</td>
<td>$ of $f_{1.0V}$</td>
</tr>
<tr>
<td></td>
<td>Avg. $</td>
<td>\text{error}</td>
<td>$ of $f_{1.0V}$</td>
</tr>
<tr>
<td>$V_{ctrl}=1.2,\text{V}$</td>
<td>Max. $</td>
<td>\text{error}</td>
<td>$ of $f_{1.2V}$</td>
</tr>
<tr>
<td></td>
<td>Avg. $</td>
<td>\text{error}</td>
<td>$ of $f_{1.2V}$</td>
</tr>
</tbody>
</table>
V. EXPERIMENTAL RESULTS

In this section, a CPPLL circuit implemented with TSMC 0.18um process is used to perform experiments to show overall BMCS approach accuracy. The PLL behavioral model is built using Verilog-A language and simulates the model in Cadence’s Virtuoso environment (Analog Artist). Referring to the statistical models of transistor parameters from TSMC, we perform (4+1) runs of behavioral parameter extraction proposed in Section III to find out pre-characterized functions for these four variation parameters, which are W, L, V, and T(csv). Then, we perform 100-run traditional HSPICE MC analysis for this PLL circuit as the golden results. The same device variation values are used in HSPICE and the following BMCS approaches to compare analysis accuracy. Locking voltage (Vlock), locking time (Tlock) and peak-to-peak (pk-pk) jitter are selected as the system characteristics of PLL circuits for comparison. In our experiments, Tlock is defined as the time when PLL output frequency is kept within 3 percent of lock frequency and pk-pk jitter is measured from 1000 clock cycles after PLL locked.

Behavioral model accuracy in BMCS-based approaches is the most critical issue for analysis results. Therefore, we first perform a 100-run BMCS using the proposed PLL behavioral model with accurate non-ideal effects to compare with the results using a top-down behavioral model provided in the Cadence’s AHDL library. Such top-down model is described by some mathematical formulas and usually adopted before implementation to obtain rough simulation results. Therefore, some detailed circuit properties and non-ideal effects are hard to be considered accurately in top-down modeling approach. In order to focus on the effects of different PLL modeling approaches, this experiment uses a 1st order RSM for both approaches to model behavioral parameter variations. Further, these experimental results demonstrate that our bottom-up behavioral modeling approach is indeed helpful in obtaining accurate results in hierarchical statistical analysis.

The MC analysis results are shown in TABLE III, in which the results from transistor-level MC analysis (HSPICE MCS) are used as the golden reference. The bottom-up model generates more accurate results, especially on the non-ideal output property – jitter. Scatter plots in Fig. 21 and 22 also demonstrate that BMCS results are more accurate on both nominal value and correlation coefficient value. Referring to the previous work [9], the accurate PLL model for BMCS improves the correlation coefficient value of Tlock from 0.888 [9] to 0.991 as shown in Fig. 22, although the 2nd order RSM is used in [9] to model behavioral parameter variations. The figures show that a behavioral model with accurate responses to process variation contributes more to accurate BMCS results. Statistical results are not accurate even if high-order regression equations are used for device variations, if the behavioral model is not accurate.

Previous experiments demonstrate that the bottom-up characterization flow helps obtain accurate BMCS results without using high-order regression equations. The following experiments verify that the proposed behavioral model and quasi-SA approach provide accurate BMCS results but significantly reduce regression efforts. With the same behavioral model, the proposed quasi-SA approach and traditional SA approach are used to model behavioral parameter variations for BMCS and compares their results in TABLE IV.

Although the average values (Mean) are accurate in both approaches, our quasi-SA models provide more similar standard deviation (St. Dev.) as in the transistor-level MC simulation, which means that our approach generates more similar variation spread of statistical results. Traditional SA methodology induces larger errors to CP and VCO behavior, therefore the predicted variation degree of Vlock and output jitter is not accurate enough. Scatter plots in Fig. 23 and 24 also demonstrate that our approach efficiently improves the estimation accuracy without extra extraction cost. Compared with results of the RSM-based approach shown in the second column of TABLE III, the proposed quasi-SA models provide similar accuracy, but reduce regression cost significantly. The 1st order RSM requires 4(n+1) extraction runs while the quasi-SA approach requires only (n+1) runs, as shown in the row “Extraction”. Extraction cost in the previous work using 2nd order RSM [9] becomes about 4[(n+1)+(n^2+n)/2]. Since n equals to four in this work, the characterization time for our process-variation-aware models is twelve times less than the 2nd order RSM approach.

The above experimental results have demonstrated that the proposed bottom-up CPPLL modeling flow and quasi-SA equations provide similar accuracy as in the RSM approach, using less extraction cost as in the traditional SA approach. The simulation time for Monte Carlo analysis is significantly reduced from several weeks to several hours, using the proposed BMCS approach. Furthermore, detailed circuit responses in this BMCS approach, such as the locking waveform of a PLL design under process variation, are easily observed. Comparing the typical and extreme cases in MC analysis, the locking waveforms of our behavioral model shown in Fig. 26 are similar to the locking waveforms of HSPICE shown in Fig. 25, which is useful information for designers to improve their designs.

Because the transistor-level simulation of a PLL circuit requires a long simulation time, we only use 100 runs of HSPICE MC simulation results to verify the accuracy of the proposed BMCS approach. However, the worst CPPLL case may not be estimated using such small samples for statistical analysis. Therefore, we use the proposed hierarchical statistical analysis to perform 1000-run MC simulations at the behavioral level to obtain comprehensive statistical analysis. Finally, there are several cases violating the jitter specification seriously to be considered as failed cases (pk-pk jitter >>20ps). The histograms of locking voltage, locking time, and pk-pk jitter of the pass cases are shown in Fig. 27 for readers’ reference to understand the performance shift of this PLL circuit under process variation. However, as shown in TABLE V, the traditional corner analysis (FF, SS, SF, FS in the same foundry model) reports that all cases pass the jitter specification. It shows that our efficient statistical approach for PLL circuits is helpful to quickly analyze the process variation effects in advance.
TABLE III
Comparing the Accuracy of Behavioral Models

<table>
<thead>
<tr>
<th></th>
<th>HSPICE</th>
<th>Our proposed PLL behavioral model</th>
<th>Top-down PLL behavioral model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Value, err. (%)</td>
<td>Value, err. (%)</td>
</tr>
<tr>
<td>$V_{\text{lock}}$ (V)</td>
<td>0.9941</td>
<td>0.9931 -0.1</td>
<td>0.9254 -6.9</td>
</tr>
<tr>
<td>$T_{\text{lock}}$ (us)</td>
<td>3.3743</td>
<td>3.4489 -2.2</td>
<td>3.1641 -6.2</td>
</tr>
<tr>
<td>pk-pk Jitter</td>
<td>St. Dev.</td>
<td>0.5772 -0.6</td>
<td>0.5447 -5.5</td>
</tr>
<tr>
<td>@ 800MHz (ps)</td>
<td>Mean</td>
<td>12.2 -7.6</td>
<td>8.0 -39.4</td>
</tr>
<tr>
<td></td>
<td>St. Dev.</td>
<td>1.40 -2.9</td>
<td>4.61 229.3</td>
</tr>
<tr>
<td></td>
<td>Worst</td>
<td>17.0 -2.4</td>
<td>18.2 7.1</td>
</tr>
<tr>
<td>$T_{\text{extr}}$ (hours)</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$T_{\text{sim}}$ (hours)</td>
<td>598.54</td>
<td>2.95 2.10</td>
<td></td>
</tr>
</tbody>
</table>

$4(n+1) \times 1.71 = 34.2$ ; $n = 4$

TABLE IV
Comparing Different Regression Methods

<table>
<thead>
<tr>
<th></th>
<th>HSPICE</th>
<th>Our Quasi-SA + BMCS</th>
<th>Trad. SA + BMCS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Value, err. (%)</td>
<td>Value, err. (%)</td>
</tr>
<tr>
<td>$V_{\text{lock}}$ (V)</td>
<td>0.9941</td>
<td>0.9947 0.1</td>
<td>0.9934 -0.1</td>
</tr>
<tr>
<td>$T_{\text{lock}}$ (us)</td>
<td>3.3743</td>
<td>3.4383 1.9</td>
<td>3.4409 2.0</td>
</tr>
<tr>
<td>pk-pk Jitter</td>
<td>St. Dev.</td>
<td>0.5764 -0.8</td>
<td>0.5406 -6.2</td>
</tr>
<tr>
<td>@ 800MHz (ps)</td>
<td>Mean</td>
<td>12.4 -6.1</td>
<td>12.4 -6.1</td>
</tr>
<tr>
<td></td>
<td>St. Dev.</td>
<td>1.41 0.7</td>
<td>2.29 63.6</td>
</tr>
<tr>
<td></td>
<td>Worst</td>
<td>16.7 -1.8</td>
<td>16.4 -3.5</td>
</tr>
<tr>
<td>$T_{\text{extr}}$ (hours)</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$T_{\text{sim}}$ (hours)</td>
<td>598.54</td>
<td>3.50 2.93</td>
<td></td>
</tr>
</tbody>
</table>

(n+1) $\times 1.71 = 8.55$ ; $n = 4$

Fig. 21. Top-down behavioral model + 1st RSM
Fig. 22. Proposed behavioral model + 1st RSM
Fig. 23. Proposed behavioral model + trad. SA
Fig. 24. Proposed behavioral model + our quasi-SA in this paper
Fig. 25. Lock waveforms of HSPICE (from 100 runs)
Fig. 26. Lock waveforms of our approach (from 100 runs)
Fig. 27. Histograms of locking voltage, locking time, and pk-pk jitter (from 1000 runs)
TABLE V

<table>
<thead>
<tr>
<th>HSPICE Simulation</th>
<th>FF</th>
<th>SS</th>
<th>FS</th>
<th>SF</th>
</tr>
</thead>
<tbody>
<tr>
<td>V(th) (V)</td>
<td>0.798</td>
<td>1.234</td>
<td>0.9301</td>
<td>1.063</td>
</tr>
<tr>
<td>t嘲 (us)</td>
<td>1.901</td>
<td>7.162</td>
<td>3.862</td>
<td>3.022</td>
</tr>
<tr>
<td>pk-pk Jitter (800MHz) (ps)</td>
<td>17.7</td>
<td>7.2</td>
<td>15.6</td>
<td>11.5</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

This paper proposes to use an accurate PLL behavioral modeling approach for fast statistical analysis of process variation effects. Besides the bottom-up behavioral modeling flow with efficient extraction process, the quasi-SA approach is also proposed to reflect process variation effects for behavioral parameters. Considering generic circuit behaviors, the quasi-SA approach is variation effects. Besides the bottom-up behavioral modeling approach for fast statistical analysis of process parameters, as shown in the experimental results, our approach speeds up analysis time by 171 times and still provides accurate statistical results to HSPICE MC simulation, including similar variation trend (correlation coefficient near 1) and statistical distribution (similar values of mean and standard deviation). These results demonstrate that this approach effectively reduces simulation time for yield analysis without sacrificing estimation accuracy.

REFERENCES


Chin-Cheng Kuo received the M.S. degrees in electronics engineering from National Central University, Taiwan, R.O.C. He is currently a Ph.D. student in the Department of Electrical Engineering at National Central University. His research interests include behavioral modeling approach and applications for analog designs.

Meng-Jung Lee received the M.S. degrees in electronics engineering from National Central University, Taiwan, R.O.C. She is currently a technical staff of Realtek Semiconductor Corp. Her research interests include analog CAD, and analog behavioral models for system verification.

Chien-Nan Jimmy Liu is an associate professor in the Department of Electrical Engineering at National Central University. His research interests include behavioral modeling for analog/mixed-signal designs, high-level power and noise modeling and functional verification for HDL designs. Liu has a BS and a PhD, both in electronics engineering, from National Chiao Tung University, HsinChu City, Taiwan, R.O.C. He is a member of the IEEE and Phi Tau Phi.

Ching-Ji Huang received the M.S. degrees in electronics engineering from National Central University, Taiwan, R.O.C., in 2003. In 2004, he joined Industrial Technology Research Institute (ITRI), Hsinchu, Taiwan, R.O.C. as a Design Engineer. His research interests include analog behavioral models for system verification, ESD protection circuit design, and standard cell library characterization.