Statistically Validating the Impact of Process Variations on Analog and Mixed Signal Designs

Ibtissem Seghaier, Mohamed H. Zaki and Sofiène Tahar
Dept. of Electrical and Computer Engineering,
Concordia University, Montréal, Québec, Canada
{seghaier, mzaki, tahar}@ece.concordia.ca

ABSTRACT

Process variation presents a practical challenge on the performance of analog and mixed signal (AMS) circuits. This paper proposes a Monte Carlo-Jackknife (MC-JK) technique, a variant of Monte Carlo method, to verify process variation affecting the performance and functionality of AMS designs. We use a behavioral model to which we encompass device variation due to $65nm$ technology process. Next, we conduct hypothesis testing based on the MC-JK technique combined with Latin hypercube sampling in a statistical run-time verification environment. Experimental results demonstrate the robustness of our approach in verifying AMS circuits.

Categories and Subject Descriptors

B.7.2 [**Design Aids**]: [Verification, Simulation]

1. INTRODUCTION

Relentless miniaturization of CMOS technology comes with its own compromises. Although it permits an increase in the level of integration, it results on the other hand in more complex designs. This makes AMS design more challenging and dictates a careful verification. Moreover, the diminutive sizing of transistors leads to an ever substantial percentage deviations from the parameters nominal values [4]. Hence, empowering designers with new tools and techniques in order to tape out designs that withstand process variation while meeting strict specification are highly required. To respond to this raising need during early verification, we investigate in this paper a new statistical verification technique that can reduce the Monte Carlo simulation time while ensuring accuracy of the results. The Monte Carlo method is an attractive technique that has a widespread use [8, 3]. Based on repetitive simulations, it permits to evaluate substantive design properties as well as to statistically estimate circuit parameters. To do so, this approach needs a prespecified underlying distribution, mainly uniform, normal, or log-normal to describe the random variables of process variation effect. Hence, a wrong distribution assumption leads to a possibility of outright wrong results. This issue has been mooted in a recent research work [7].

This paper addresses the shortcomings of conventional sampling based verification method by (1) providing better parameter space coverage with reduced simulation overhead results from the Latin Hypercube sampling [1] as an alternative to Pseudo Random Monte Carlo sampling; and (2) offering more accurate error margins and hence better verification accuracy is achieved using the Jackknife statistical method to estimate the Monte Carlo distribution parameters.

The rest of this paper is organized as follows: Details of the proposed methodology are given in Section 2. Thereafter, we report experimental results for the verification of a ring oscillator and a Charge Pump PLL in Section 3. Finally, we conclude the paper in Section 4.

GLSVLSI'15, May 20 - 22, 2015, Pittsburgh, PA, USA.
 ACM 978-1-4503-3474-7/15/05$15.00.
http://dx.doi.org/10.1145/2742060.2742122.

2. PROPOSED METHODOLOGY

Figure 1 depicts the overall proposed verification methodology. Given an AMS design description, we derive its behavioral model. Using the process variation libraries created by technology vendors, we then choose a range of parameter deviations for 65nm process. Traditional sampling techniques such as Pseudo Random Sampling (PRS) arranges

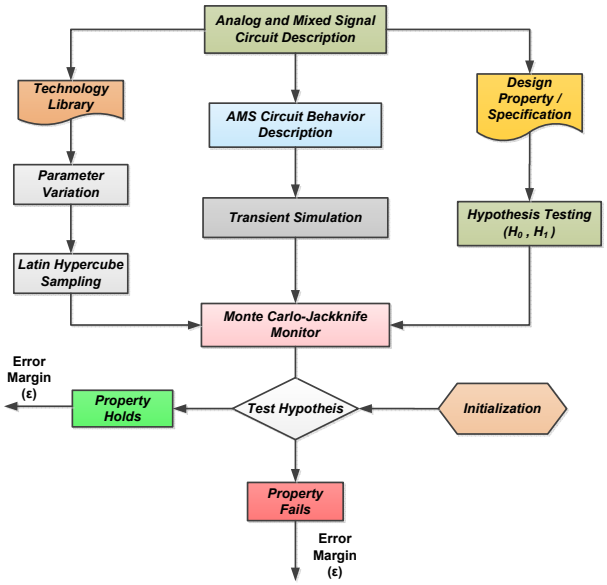

Figure 1: Overview of the proposed methodology

parameter values at some specific corners in the parameter space. When running Monte Carlo simulation, it cannot mimic the system behavior in a global parameter space. However, the Latin Hypercube technique gives samples that could reflect the integral distribution more effectively with a reduced samples variance. In fact, it turns out to be five times more effective than PRS Monte Carlo in yield estimation as shown in [2]. For an efficient sampling procedure from the process variation distribution, the Latin Hypercube Sampling technique is deployed (more details are given in Section 2.1). The AMS circuit is thereafter evaluated for the obtained parameter samples for specific environment constraints, namely the initial values of the voltage and current state variables and simulation parameters (the total simulation time, the simulation step size and so on).
On the other hand, we elucidate a property of interest ($\mathcal{P}$) that the circuit output should comply with. This property is phrased as hypothesis testing problem. It consists of defining a null hypothesis H_0 of $\mathcal{P}$ and an alternative hypothesis H_1 which is naturally the counterexample ($\mathcal{Q}$) opposite to ($\mathcal{P}$). For the chosen parameter variation values and a specified tail test, a critical value based Monte Carlo-Jackknife monitor is carried for a confidence level δ. The property $\mathcal{P}$ is verified if the null hypothesis H_0 is accepted, else the monitor reports a violation. The conclusion of accepting or rejecting H_0 is drawn with an error margin ϵ for a 95% confidence level (see Fig. 1).

2.1 Latin Hypercube Sampling

Latin Hypercube Sampling (LHS) [1] is a an optimized statistical sampling technique to extract parameter values from defined multidimensional distribution. We use LHS to select samples in the process variation space. To construct a Latin hypercube sample of n_p data points from an AMS circuit model with n_d dimensions (state variables) defined from a uniformly distribution $X \sim U(0.1)^{n_d}$, the sampling space is divided in such a way that each of the n_d dimensions are divided into n_b blocks as follows :

$$X_{ij} = \frac{\pi_j(i-1) + U_{ij}}{n_s}, 1 \leq i \leq n_s, 1 \leq j \leq n_d$$

where π_j are uniform random permutations $U_{ij} \sim U[0,1)$ wherein U_{ij} and π_j are independent. LHS has a multiple stratification property:

$$\forall c = 0, ..., n_s - 1, \forall j = 1, ..., n_d$$
$$prob\{1 \leq i \leq n_s | \frac{c}{n_s} \leq X_{ij} < \frac{c+1}{n_s}\} = 1$$

This technique offers variance sampling reduction which results in a better verification coverage.

2.2 Jackknife Technique

The Jackknife technique [6] was originally developed as a nonparametric way to estimate and reduce the bias of an estimator of a population parameter. The bias of an estimator is defined as the difference between the expected value of this estimator and its true value. The Jackknife procedure works as follows: First, remove d data points and calculate the statistic of interest. Second, calculate the pseudo-values according to Equation 1. Then, repeat this process, leaving out d data points at a time to build a distribution of the statistic. Finally, use that distribution to estimate the statistic and its uncertainty. For an estimator S, the i^{th} pseudo-value Jackknife of S was calculated as follows:

$$ps_i = NS - (N-1)S_i \quad (1)$$

where S_i is the estimator value for the sample with the i^{th} data point deleted. The Jackknife Confidence Interval (CI) of this estimate for 95% confidence level is then given by:

$$CI_J = \bar{ps} \pm 2\sqrt{\frac{\sigma_J}{N}} \quad (2)$$

$$\text{where} \quad \sigma_J = \sum \frac{(ps_i - \bar{ps})^2}{N-1}, \quad \bar{ps} = \frac{1}{N}\sum ps_i$$

Hence, the Jackknife reduces the bias of the parameter estimates as well as the variance. The detailed procedure for Monte Carlo-Jackknife (MC-JK) based hypothesis testing technique for AMS circuits is illustrated in Algorithm 1, where V_{out} represents the observed circuit output with process variation, M denotes the number of MC-JK samples, d is a parameter for the *deleted_d_jackknife* method, α a chosen significant level and *test* stands for the type of test to be performed. The algorithm starts with drawing M samples from the circuit output V_{out} of size N by leaving out d samples of the output at a time (line 3). The deviation between the output and H_0 is computed using a test statistic estimation T_{JK} for each Jackknife pseudo-sample. Next, the Monte Carlo quantile procedure [8] is employed to measure the critical by type of test: For an upper tail test (line 5)/lower tail test (line 11), the $1-\alpha/\alpha$ quantiles of the empirical distribution, respectively. In the case of two tailed test, both $1-\frac{1}{\alpha}$ and $\frac{\alpha}{2}$ quantiles define the lower and upper critical values (lines 18-19). Once the critical value is determined, the monitor decides about the satisfaction or violation of H_0.

Algorithm 1 Monte Carlo-Jackknife Verification Algorithm

Require: V_{out}, T_{obs}, α, *test*, M, d
  $N \leftarrow length(V_{out})$
  for $i \leftarrow$ 1 *to* N **do**
3: $\theta \leftarrow delete\_d\_Jacknife$
  $T_{JK}(i) \leftarrow Measure\_test\_statistic(\theta)$
  while *test* = "*upper tail test*" **do**
6: $CV = quantile(T_{JK}, 1-\alpha)$
  if $CV \geq T_{obs}$ **then**
  Accept H_0
9: **else**
  Reject H_0
  while *test* = "*lower tail test*" **do**
12: $CV = quantile(T_{JK}, \alpha)$
  if $CV \leq T_{obs}$ **then**
  Accept H_0
15: **else**
  Reject H_0
  while *test* = "*two tailed test*" **do**
18: $CV_L = quantile(T_{JK}, \frac{\alpha}{2})$
  $CV_U = quantile(T_{JK}, \frac{1-\alpha}{2})$
  if $CV_L \geq T_{obs}$ **or** $CV_U \leq T_{obs}$ **then**
21: *Reject* H_0
  else
  Accept H_0

3. EXPERIMENTAL RESULTS

In this section, we report the results of the application of our methodology to a ring oscillator and a PLL. All computation and circuit models were performed in a MATLAB environment for M=1000 trials and confidence level $\delta = 0.95$. The experiments are run on a 64-bit Windows 7 server with 2.8 GHz processor and 24 GB memory.

3.1 Ring Oscillator

A ring oscillator is a closed-loop chain of an odd number of inverters placed in series with a negative feedback to provide oscillation. Each inverter is composed of a cascaded n-channel and p-channel transistors.

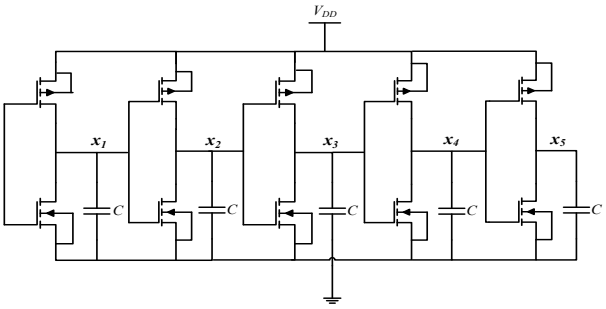

Figure 2: Five-stage CMOS ring oscillator schematic

A five-stage ring oscillator circuit is shown in Fig. 2. To model the influence of the interconnect circuitry, an additional load capacity of C was used. The circuit dynamics are governed by Equation (3):

$$\begin{aligned} \frac{dx_1}{dt} &= -\frac{1}{C}(I_n(x_n, x_1, gnd) + I_p(x_n, x_1, V_{DD}) \quad (3)\\ \frac{dx_i}{dt} &= -\frac{1}{C}(I_n(x_{i+1}, x_i, gnd) + I_p(x_{i+1}, x_i, V_{DD}),\ \forall i \in [2, n]\\ V_{out} &= x_n \end{aligned}$$

where $\{x_i\}_{i=1}^{n}$ and gnd stand for the node and ground voltages, respectively. We model the nonlinear current generated by the n-channel and p-channel transistors as functions I_n and I_p, respectively. The node voltages of each of the 5 inverters has been designed to oscillate between the power V_{dd} and the ground gnd at a frequency of 4.5GHz with a tolerance of ±50MHz. As a result, the null hypothesis H_0 and the alternative hypothesis H_1 can be expressed as:

$$\begin{aligned} H_0 &: 4.450 \text{ GHz} \leq f_{osc} \leq 4.550 \text{ GHz}; \quad (4)\\ H_1 &: f_{osc} \geq 4.550 \text{ GHz} \parallel f_{osc} \leq 4.450 \text{ GHz}; \end{aligned}$$

To study the effect of process variation on the oscillation frequency (f_{osc}), we choose two scenarios:
1) The channel width w_{n_i} of each n-MOS transistor $\{M_{n_i}\}_{i=1}^{5}$ has up to 15%.
2) The power supply voltage V_{DD} is assumed to have 5% variation with Gaussian distribution around the nominal value.

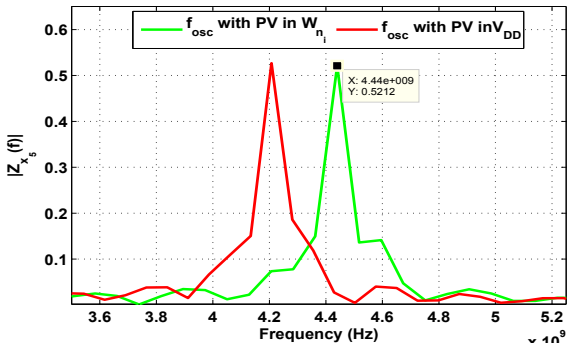

Figure 3: Frequency spectrum of the circuit output

To compute f_{osc}, we employ the Fourier Transform (FT) analysis in MATLAB [5] . A comparison between the Monte Carlo (MC) technique and the proposed approach for the above mentioned scenarios under several process variation factors are summarized in Table 1. Since a two tailed test is chosen, a rejection decision of H_0 is announced if the observed value T_{obs} is within the rejection region ($T_{obs} \in]-\infty, C.V_L] \cup [C.V_L, +\infty[$). It can be noticed from Table 1 that when the percentage of variation increases from 1 to 15% in the case of transistor width deviation (Row 1), both MC and MC-JK techniques report decision changes from acceptance to rejection. However, by sweeping the w_{n_i} up to 10% of variation, an erratic decision between the Monte Carlo and the proposed Jackknife-Monte Carlo is remarked (see Table 1). Moreover, in all cases our approach presents less error margins. This can be explained by the biased estimation of the Monte Carlo technique and the reduced sampling error offered by our approach.

Figure 3 depicts the Single-Sided Amplitude Spectrum of the ring oscillator output x_5 for the two different process variation scenarios. It can be noticed that a great deviation of the tolerated oscillation frequency has been obtained. Therefore, a process variation in the power supply voltage V_{DD} is identified to have a greater effect than a process variation in all 5 n-MOS transistor lengths for 5% trimmed deviation.

3.2 Charge-Pump Phase Locked Loop

We apply the proposed methodology to verify the locking property of a 3^{rd} order dual path Charge Pump PLL (CP-PLL) design shown in Fig. 4. The Phase Frequency Detector (PFD), which is a digital component, operates as follows: The PFD compares the phases of the reference

Table 1: Statistical runtime verification of oscillation frequency of ring oscillator for $\alpha = 0.05$

	$J(ns)$	$C.V_U$		$C.V_L$		T_{obs}		f_{obs} (GHz)		H_0		ϵ (MHz)	
		MC	MC-JK	MC	MC-JK	MC	MC-JK	MC	MC-JK	MC	MC-JK	MC	MC-JK
PV in W_{n_i}	1%	-1.487	-1.326	1.487	1.326	1.0389	0.9754	4.51	4.49	Accept	Accept	34.8	16.9
	5%	-1.533	-1.378	1.533	1.378	1.0773	0.8559	4.48	4.47	Accept	Accept	25.3	20.3
	10%	-1.571	-1.391	1.571	1.391	1.0584	0.8131	4.46	4.47	Accept	**Reject**	32.5	24.9
	15%	-1.652	-1.441	1.652	1.441	2.269	1.938	4.45	4.44	**Reject**	**Reject**	39.0	22.7
PV in V_{DD}	1%	-1.845	-1.722	1.845	1.722	0.741	0.825	4.47	4.49	Accept	Accept	17.2	15.6
	2.5%	-1.883	-1.781	1.883	1.781	-2.179	-1.894	4.46	4.48	Accept	Accept	18.7	16.9
	5%	-1.576	-1.668	1.576	1.668	1.848	1.857	4.25	4.18	**Reject**	**Reject**	25.3	21.7

Table 2: Statistical runtime verification of PLL locking time for $\alpha = 0.05$

case	$J(ns)$	c.v		T_{obs}		N_{obs}		H_0		ϵ	
		MC	MC-JK	MC	MC-JK	MC	MC-JK	MC	MC-JK	MC	MC-JK
P.V in VCO and LPF	0.5%	1.454	1.361	-2.389	-2.771	971	1098	Accept	Accept	154	138
	1%	1.527	1.431	0.768	0.592	1229	1274	Accept	Accept	174	146
	2%	1.676	1.569	13.443	10.782	1479	1583	**Reject**	**Reject**	182	165
	5%	1.541	1.497	162.19	137.08	1847	1643	**Reject**	**Reject**	223	177

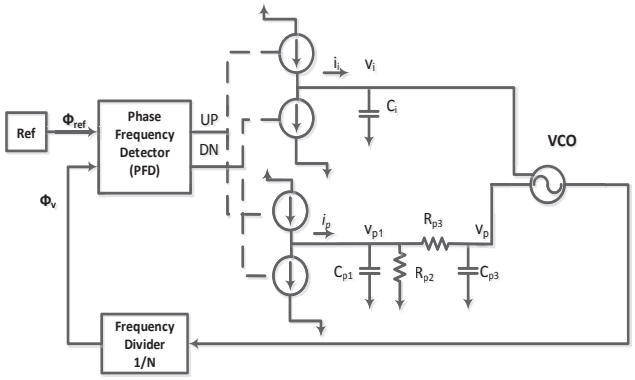

Figure 4: 3^{rd} order CP-PLL macro model

signal ϕ_{ref} and the feedback signal ϕ_v during their rising edges. If the phase difference is within the tolerated margins, no current is injected in the Charge Pumps (CP). However, if the reference oscillator leads/lags the feedback signal (ϕ_{ref}/ϕ_v reaches 2π first), then the CP injects current to charge/discharge the Low Pass Filter (LPF) capacitors which increases the control voltage.

The dynamics of the analog PLL components is described by the following equations:

$$\begin{aligned} \frac{dv_i}{dt} &= \frac{1}{C_i} i_i \\ \frac{dv_{p1}}{dt} &= \frac{1}{C_{p1}}[(\frac{1}{R_{p2}} + \frac{1}{R_{p3}})v_{p1} + \frac{1}{R_{p3}} v_p + i_p] \\ \frac{dv_p}{dt} &= \frac{1}{C_{p3} R_{p3}}(v_{p1} - v_p) \\ \frac{d\phi_i}{dt} &= \frac{1}{N}(K_i v_i + K_p v_p + 2\pi f_0) \\ \frac{d\phi_{ref}}{dt} &= 2\pi f_0 \end{aligned} \quad (5)$$

The locking property we attempt to verify can be expressed as: *The required number of cycles for the PLL to lock is less than a certain value (*$\mid \phi_v - \phi_{ref} \mid \leq 0.2\,^\circ$*)*. This property can be formulated as follows:

$$\begin{aligned} H_0 &: T_{lock} \leq 1500\ cycles; \\ H_1 &: T_{lock} > 1500\ cycles; \end{aligned} \quad (6)$$

Table 2 shows the results of verifying the locking time of the CP-PLL circuit using an upper tail test for a significance level $\alpha = 0.05$. We compared the discrepancy between the results obtained using MC and our MC-JK approach. The obtained results demonstrate a good agreement between the two techniques in verifying the PLL locking (H_0 column in Table 2). Besides, a better verification accuracy is remarked using our approach with an error margins ϵ less than those given by the MC technique. For instance, for 2% process variation in the VCO and LPF the error margin computed using our approach $\epsilon = 165$ cycles while an error estimate of $\epsilon = 182$ cycles has been given by traditional MC.

4. CONCLUSION

In this paper, a Monte Carlo-Jackknife statistical approach is proposed to handle the verification of analog and mixed signal designs under $65nm$ process variation. The proposed technique constructs statistically a sound hypothesis testing approach and hence can be used in lieu of traditional Monte Carlo simulation. For instance, it provides better accuracy with reduced error margin than Monte Carlo. However, MC-JK is more computationally intensive than MC based statistical runtime verification. To alleviate this, we deployed the Latin Hypercube method for a more effective sampling with less trials.

As future work, we plan to extend the proposed approach to online runtime verification fashion for the same accuracy. Moreover, we intend to study more uncertainties in AMS designs such as noise, and initial condition variation.

5. REFERENCES

[1] J. C. Helton et al. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering & System Safety*, 91(10):1175–1209, 2006.

[2] C. Kuo et al. Efficient trimmed-sample monte carlo methodology and yield-aware design flow for analog circuits. In *Design Automation Conference*, pages 1113–1118, 2012.

[3] L. Lin and W. Burleson. Analysis and mitigation of process variation impacts on power-attack tolerance. In *Design Automation Conference*, pages 238–243, 2009.

[4] J. K. Lorenz et al. Hierarchical simulation of process variations and their impact on circuits and systems: Methodology. *IEEE Transactions on Electron Devices*, 58(8):2218–2226, 2011.

[5] MATLAB. Fast Fourier Transform. `http://www.mathworks.com/help/matlab/ref/fft.html`, 2015.

[6] F. Mecatti, P. L. Conti, and M. G. Ranalli. *Contributions to Sampling Statistics*. Springer, 2013.

[7] R. Narayanan, I. Seghaier, M. H. Zaki, and S. Tahar. Statistical run-time verification of analog circuits in presence of noise and process variation. *IEEE Transaction on VLSI*, 21(10):1811–1822, 2013.

[8] Z. Wang, M. H. Zaki, and S. Tahar. Statistical runtime verification of analog and mixed signal designs. In *Signals, Circuits and Systems*, pages 1–6, 2009.