

A Qualitative Simulation Approach for Verifying PLL Locking Property

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

ABSTRACT

Simulation cannot give a full coverage of Phase Locked Loop (PLL) behavior in presence of process variation, jitter and varying initial conditions. Qualitative Simulation is an attracting method that computes behavior envelopes for dynamical systems over continuous ranges of their parameters. Therefore, this method can be employed to verify PLLs locking property given a model that encompasses their imperfections. Extended System of Recurrence Equations (ESREs) offer a unified modeling language to model analog and digital PLLs components. In this paper, an ESRE model is created for both PLLs and their imperfections. Then, a modified qualitative simulation algorithm is used to guarantee that the PLL locking time is sound for every possible initial condition and parameter value. We used our approach to analyze a Charge Pump-PLL for a $0.18\mu\text{m}$ fabrication process and in the presence of jitter and initial conditions uncertainties. The obtained results show an improvement of simulation coverage by computing the minimum locking time and predicting a non locking case that statistical simulation technique fails to detect.

Categories and Subject Descriptors

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

Keywords

PLL; ESRE; Qualitative Simulation

1. INTRODUCTION

The verification of Analog and Mixed Signal (AMS) designs is challenging and time-consuming because of their infinite state space, the fundamental differences of their digital and analog components operating modes, and their sensitivity to initial conditions uncertainties, process variation, device jitter and noise [5]. Phase Locked Loops (PLLs) are one

of the basic and widely used AMS circuits in modern electronic systems. They are used as modulators and demodulators in wireless systems, frequency synthesizer in communication systems, and clock-acquisition in high-speed links. Several research activities have been done to model PLLs at high abstraction levels [3, 7, 13] in order to identify their functional errors at early design stages. These abstracted models offer a good accuracy and can serve as a reference to check the behavior of the device level PLL implementation. The required time for PLLs to lock is a key property in their verification. Typically, numerical simulation techniques are conducted to check the PLL locking for a finite number of initial conditions and parameters values. However, simulation cannot guarantee that there are no other possible initial conditions and parameters values that can derive the PLL into unlocking. In order to overcome this verification coverage shortcoming, the circuit should be verified for entire intervals of possible initial conditions and parameter values. Qualitative Simulation is a method for the simulation of continuous dynamical systems whose parameters and/or initial conditions are modeled by fuzzy distributions [10]. Interestingly, this method provides a phase space representation of dynamical model sensitivity to parameters which can be used to verify PLL locking property.

In this paper, a new modeling and verification methodology for PLL designs is proposed. The PLL design is described in terms of Extended System of Recurrence Equations (ESREs) (to be mathematically defined later). These equations offer a unified language to describe both analog and digital blocks of PLLs as functions of the preceding state variables terms. Moreover, Extended SREs offer a means of modeling more abstracted designs which significantly speed up the verification execution time. Our ESRE PLL models include the effect of accumulating and synchronous jitter as well as process variation. The Qualitative Simulation algorithm proposed in [10] has been reformulated for the case of discrete models expressed using ESREs to enable the verification of PLL locking behavior over entire ranges of initial conditions and parameter values. To illustrate the use of our methodology, we study the lock time property of a third order dual path charge pump PLL. The obtained results show that we can predict unlocking corner cases which cannot be determined via nominal parameters simulation nor statistical simulation.

The rest of the paper is organized as follows: related work is discussed in Section 2. Then, Section 3 provides an overview of the proposed methodology. After that, we report experimental results for the analysis of the lock time

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
GLSVLSI'14, May 21–23, 2014, Houston, TX, USA.
Copyright 2014 ACM 978-1-4503-2816-6/14/05\$15.00.
<http://dx.doi.org/10.1145/2591513.2591593>.

of a third order dual path charge pump PLL in Section 4. Finally, the concluding Section 5 summarizes the contributions of this paper and provides future work.

2. RELATED WORK

Simulation based techniques provide an insight of the circuit behavior only for particular values of parameters and initial conditions. Therefore, they suffer from uncertainty about verification coverage. Indeed, running multiple simulations cannot guarantee the absence of corner cases where the subject property is not satisfied [4]. To improve the coverage of analog circuits state space, the authors in [12] proposed an algorithm that guides the input stimuli but falls short to address the effect of Process Variation (PV). In [9], the PLL is modeled using Stochastic Differential Equations (SDE). Then, it is verified using a statistical runtime verification approach that combines Monte Carlo/Bootstrap techniques with hypothesis testing. The accuracy of the verification technique is directly related to the confidence level which depends on the number of simulation runs [9]. In addition, this technique introduces randomness in the simulation but fails to cover all possible design behaviors.

The use of formal methods is another paradigm that has been adopted by many researchers [15]. For instance, an equivalence checking formal technique has been used in [11] where the authors verify the equivalence of a Verilog AMS PLL behavioral model against its Spectre netlist electrical implementation for a limited set of input conditions and without imperfections. In [14], the authors proposed a method for reachability analysis of AMS designs, such as sigma delta converters, using Taylor approximations and interval arithmetic. However, this method suffers from dependency and wrapping effect problems. In [3], a reachability analysis technique has been proposed to verify the locking of a PLL circuit (which we intend to also verify in this paper) under parameters and initial condition variations. Taylor approximation was used to model the PLL design and a method called continuization was proposed to over-approximate the reachable state based on a geometrical representation namely polyhedra in the multidimensional space. This technique has the following shortcomings: first, it does not consider jitter which is the major concern in the locking of PLLs [2]. Second, their proposed model and algorithm are not scalable to handle other PLL designs.

3. PROPOSED METHODOLOGY

An overview of our proposed methodology for modeling and verifying PLL designs is depicted in Figure 1. This methodology can be split into two main phases: a modeling phase that results in an ESRE model including process variation and two different jitter models and a Qualitative Simulation based verification which assess the effect of PLLs imperfections on its locking time. We have implemented this methodology using the numerical computation environment MATLAB.

3.1 Modeling Phase

PLL designs contain both analog and digital modules that are interconnected and interrelated. Hence, it is not appropriate to model these modules separately. Therefore, we have to describe the continuous (analog) signal and the discrete (digital) signal using the same language in order to

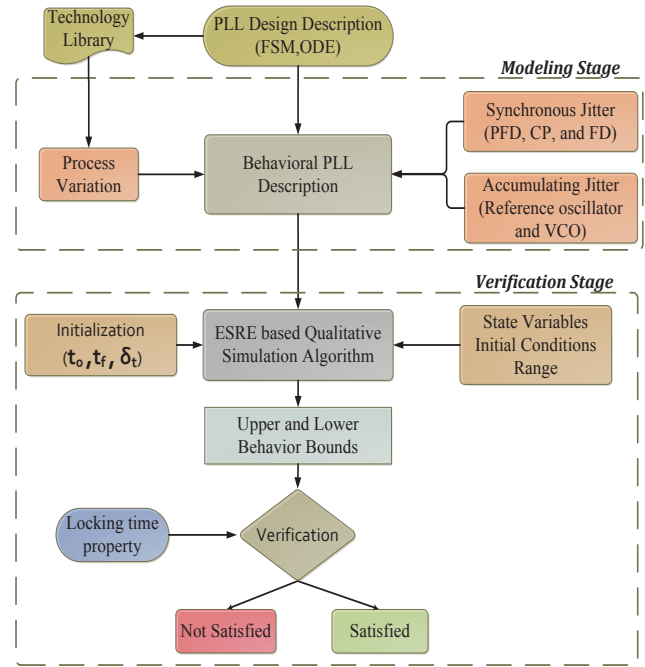


Figure 1: PLL Modeling and Verification Methodology

exhibit the interrelationships between the two. The behavior of analog circuits can be mathematically modeled by Ordinary Differential Equations (ODEs). Since a close-form solution for these ODEs is not always obtained, a numerical approximation is needed. Using System of Recurrence Equations (SREs) [1], it will be possible to handle continuous behaviors like that of currents and voltages in discrete time intervals which can be done for a non-trivial class of analog circuits. An SRE is a set of relations between consecutive elements of a sequence. It is mathematically defined as a system consisting of a set of equations of the form:

$$x_i(n_t) = f_i(x_j(n_t - \delta)), \forall n_t \in \mathbb{Z} \quad (1)$$

where $x_i(n_t) \in \mathbb{R}$ is a state variable with $i, j \in 1, \dots, k$ and $n_t \in \mathbb{Z}$, and $\delta \in \mathbb{N}$ represents the delay. On the other hand, digital designs are described using various frameworks such as Finite State Machines and Petri nets. To alleviate the modeling gap between the digital and analog models, we use the notion of Extended SRE (ESRE) for interleaving the two [1]. ESREs offer a means of modeling more abstracted PLL designs which will significantly speed up the verification execution time. In [1], the authors define the notion of ESREs: “Generalized If-formula is a class of expressions that extend SREs to describe digital systems”. Mathematically, it is defined as follows: let \mathbb{K} be a numerical domain ($\mathbb{B}, \mathbb{N}, \mathbb{Z}, \mathbb{Q}$ or \mathbb{R}), a *generalized If-formula* is one of the followings:

- A variable $x_i(n)$ or a constant $C \in \mathbb{K}$.
- Any arithmetic operation $\diamond \in (+, -, \times, \div)$ between variables $x_i(n) \in \mathbb{K}$.
- A logical formula: any expression constructed using a set of variables $x_i(n) \in \mathbb{B}$ and logical operators : not, or, and, nand, nor, etc.
- A comparison formula: any expression constructed using a set of $x_i(n) \in \mathbb{K}$ and comparison operators $\alpha \in (<, =, >, <>)$.

- An expression $If(x, y, z)$, where x is a logical formula or a comparison formula and y, z are any generalized If-formula. Here, $If(x, y, z) : \mathbb{B} \times \mathbb{K} \times \mathbb{K} \rightarrow \mathbb{K}$ satisfies the axioms:

$$\begin{aligned} If(True, x, y) &= x \\ If(False, x, y) &= y \end{aligned}$$

To accommodate the imprecise nature of PLL designs, design imperfections will be taken into account and will be considered as an integral part of the design modeling method. As a next step, we append to the generated PLL ESRE model two imperfections effect. The process variation due to $0.18\mu\text{m}$ process [8] and the jitter (a.k.a phase noise). Depending on the fabrication technology and the circuit configuration, we include process variation in some or all circuit components. We assume that the parameter values adhere to a Gaussian distribution with mean equal to their nominal values and a standard error σ equal to the deviation percentage extracted from the technology library.

As characterized by Kundert in [6], there are two types of jitter that impact the locking property of PLLs: synchronous jitter exhibited by driven blocks such as the Phase Frequency Detector (PFD), Charge Pump (CP) and Frequency Divider (FD), and accumulating jitter exhibited by autonomous blocks such as the Voltage Controlled Oscillator (VCO) and the reference oscillator (Ref). As a next step, we include the synchronous jitter as a phase modulation. The accumulating jitter is modeled as frequency modulation since in its presence the frequency of a signal fluctuates randomly [6].

3.2 Verification Phase

The behavior of a PLL design cannot be precisely defined in many cases due to inherent initial conditions uncertainty. For that reason, complete ranges of the PLL initial conditions are considered, as done in [3]. They are generated as ranges of values that follow uniform probability distribution as done in [3]. Also, the simulation parameters such as the step-size δ_t , the initial simulation time t_o , and the simulation end time t_f are specified.

Thereafter, the proposed ESRE based Qualitative Simulation algorithm is used to generate a lower and an upper bounds for the transient behavior of each of the ESRE PLL

Algorithm 1 ESRE based Qualitative Simulation

Require: $ESRE(X)$, $ESRE_{property}$, $T = [t_1, t_2, \dots, t_N]$, IC

```

1: for  $n_t \leftarrow t_2$  to  $t_N$  do
2:   for  $n_{t^*} \leftarrow t_1$  to  $n_t$  do
3:      $X_{min}(n_t) \leftarrow +\infty$ 
4:      $X_{max}(n_t) \leftarrow -\infty$ 
5:      $\Delta t \leftarrow n_{t^*} - n_{t^*-1}$ 
6:      $X_{min}(n_t) \leftarrow \min(X(n_{t^*}) \mid X(n_{t^*}) = f(X(n_{t^*} - \Delta t)) \wedge If(LB_{IC} < IC < UB_{IC}, true, false)$ 
7:      $X_{max}(n_t) \leftarrow \max(X(n_{t^*}) \mid X(n_{t^*}) = f(X(n_{t^*} - \Delta t)) \wedge If(LB_{IC} < IC < UB_{IC}, true, false)$ 
8:   end for
9:    $B_L(n_t) \leftarrow X_{min}(n_t)$ 
10:   $B_U(n_t) \leftarrow X_{max}(n_t)$ 
11: end for
12: Evaluate  $\leftarrow If(ESRE_{property}, true, false)$ 

```

model state variables (see Figure 1). These envelopes over-approximate the behavior of the PLL design without using any geometrical representation, as in [3]. The behavior of the circuit is tracked over a range of conditions in one execution. In this paper, we extend the Qualitative Simulation algorithm proposed in [10] to support the verification of PLL designs described with ESREs (see Algorithm 1). Algorithm 1 requires: an ESREs model of the circuit with imperfections denoted by $ESRE(X)$, the desired property to be verified $ESRE_{property}$, which is the locking property for PLL (for example: the required number of cycle for the PLL to lock), the sampling time vector T of size N , and the initial conditions IC , which are m number of intervals for the state variable vector X of size m . The algorithm uses a global optimization technique to compute the minimum X_{min} (line 6) and maximum X_{max} values (line 7) of the state variable X such that they are a solution of the ESRE model for the initial conditions ranges. For each time instant n_t , the computation of X_{max} and X_{min} is done by evaluating all solutions starting from the first sampling instant (see lines 2-8 in Algorithm 1). To do so, the time step Δt is not equidistant; it is adaptatively refined in order to ensure accuracy while optimizing computations. Thereafter, the algorithm affects the obtained values of X_{min} and X_{max} to the lower bound B_L (line 9) and the upper bound B_U (line 10), respectively. Therefore, the behavior of the PLL design is bounded between the B_U and the B_L for each state variable. These envelopes are used in an offline based verification of the locking property $ESRE_{property}$ (line 12). The B_U and the B_L are involved to verify if the PLL locking property is satisfied or not, as depicted in Figure 1.

4. CHARGE PUMP PLL

In this section, we apply the proposed methodology on the dual path third order Charge Pump PLL depicted in Figure 2 [3]. It is comprised of five blocks: a reference signal oscillator, a phase frequency detector, charge pumps, a low pass filter, and a voltage controlled oscillator. By placing a frequency divider in the feedback path, the PLL can be used as a frequency synthesizer that multiplies the low frequency of the reference oscillator to generate a higher output frequency signal. Locking is a key property for all PLL based frequency synthesizers. This property is defined as the synchronization of the VCO frequency f_{VCO} being equal to N_D times the input reference frequency f_{ref} .

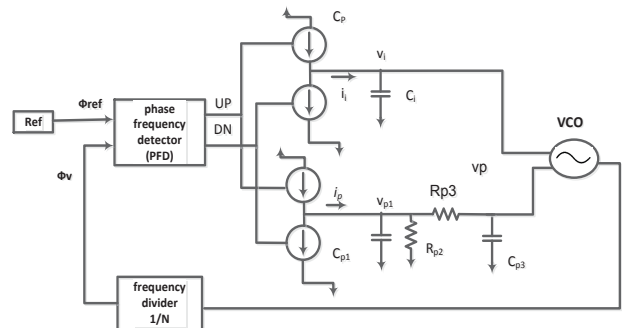


Figure 2: CP-PLL Based Frequency Synthesizer [3]

4.1 Behavioral Model

The CP-PLL based frequency synthesizer schematic, shown in Figure 2, has two different models:

1. Continuous Time Dynamic Model which is a set of first order ODEs (Equation (2)) that describes the behavior of the analog components namely, the Ref, the CPs, the LPF, the VCO, and the FD.

$$\dot{x} = Ax + Bu + c \quad (2)$$

where $x = [v_i \ v_{p1} \ v_p \ \phi_v \ \phi_{ref}]^T$ denotes the continuous state vector and u is the input vector $u = [i_i \ i_p]^T$

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{C_{p1}}\left(\frac{1}{R_{p2}} + \frac{1}{R_{p3}}\right) & \frac{1}{C_{p1}R_{p3}} & 0 & 0 \\ 0 & \frac{1}{C_{p3}R_{p3}} & -\frac{1}{C_{p3}R_{p3}} & 0 & 0 \\ \frac{K_i}{N_D} & 0 & \frac{K_p}{N_D} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \frac{1}{C_i} & 0 \\ 0 & \frac{1}{C_{p1}} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{2\pi}{N_D}f_0 \\ 2\pi f_{ref} \end{bmatrix}$$

We keep the same notations in Figure 2 for the circuit variables (voltage and current nodes) and components (capacitor and resistors). The VCO frequency f_{VCO} can be determined as follows:

$$f_{VCO} = \frac{1}{2\pi}(K_i v_i + K_p v_p) + f_0 \quad (3)$$

where f_0 stands for the offset frequency and K_i and K_p represent the VCO gain.

2. Discrete Event Dynamic Model which characterizes the digital component PFD behavior. Its operation is described by the four states Finite State Machine (FSM) shown in Figure 3. After comparing the VCO phase (ϕ_{VCO}) with the phase of the reference oscillator (ϕ_{ref}), two state sequences are possible:

- *Both_off* \rightarrow *Up_active* \rightarrow *Both_on* \rightarrow *Both_off*: if the reference oscillator leads the feedback signal ($\phi_{ref} = 2\pi$ is reached first), then the CP injects a current to charge the LPF capacitor, which increases the VCO control voltage.
- *Both_off* \rightarrow *Dw_active* \rightarrow *Both_on* \rightarrow *Both_off*: if the reference oscillator lags the feedback signal ($\phi_v = 2\pi$ is reached first), then the CP injects a current to discharge the LPF capacitor, which decreases the VCO control voltage.

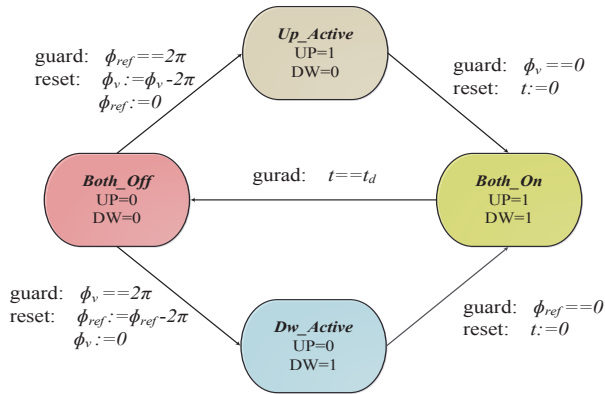


Figure 3: PFD Dynamic Model [3]

As illustrated in Figure 3, the phase values are reset to ($\phi - 2\pi$) after the leading signal, whose phase reaches 2π first, is identified. Therefore, the same transition condition can be used for the next cycle. At the zero crossing of the lagging signal, the FSM passes to the state *Both_On*. After a time delay t_d , which represents the time required to switch off the charge pumps, the FSM enters the *Both_Off* state. The input vector u changes according to the PFD FSM states, as follows:

$$\mathbf{u} = \begin{cases} [0 \ 0]^T & \text{if } \text{Both_Off} \\ [I_i^{DW} \ I_p^{DW}]^T & \text{if } \text{Dw_active} \\ [I_i^{UP} \ I_p^{UP}]^T & \text{if } \text{Up_active} \\ [I_i^{UP} + I_i^{DW} \ I_p^{UP} + I_p^{DW}]^T & \text{if } \text{Both_On} \end{cases}$$

with $I_i^{DW} = -I_i^{UP}$ and $I_p^{DW} = -I_p^{UP}$

Extended System of Recurrence Equations Model

The analog behavior of the PLL (described by the linear continuous dynamic model) and the digital behavior of the PFD (described as FSM) are transformed into ESREs using the approach defined in Section 3.1, as follows :

$$v_i(n+1) = \text{if}((\text{Both_On} \vee \text{Both_Off}), v_i(n) + f_1(n, 0), \text{if}(\text{Up_active}, v_i(n) + f_1(n, I_i^{UP}), v_i(n) + f_1(n, I_i^{DW})))$$

$$v_{p1}(n+1) = \text{if}((\text{Both_On} \vee \text{Both_Off}), v_{p1}(n) - f_2(n, 0), \text{if}(\text{Up_active}, v_{p1}(n) - f_2(n, I_p^{UP}), v_{p1}(n) - f_2(n, I_p^{DW})))$$

$$v_p(n+1) = \text{if}(\text{true}, v_p(n) + f_3(n), 0)$$

$$\phi_v(n+1) = \text{if}(\text{true}, \phi_v(n) + f_4(n), 0)$$

$$\phi_{ref}(n+1) = \text{if}(\text{true}, \phi_{ref}(n) + f_5(n), 0)$$

where

$$f_1(n, I) = \frac{1}{C_i} I \delta_n \quad (4)$$

$$f_2(n, I) = \frac{1}{C_{p1}} \left[\left(\frac{1}{R_{p2}} + \frac{1}{R_{p3}} \right) v_{p1}(n) - \frac{1}{R_{p3}} v_p(n) - I \right] \delta_n \quad (5)$$

$$f_3(n) = \left[\frac{1}{C_{p3}R_{p3}} (v_{p1}(n) - v_p(n)) \right] \delta_n \quad (6)$$

$$f_4(n) = \left[\frac{K_i}{N_D} v_i(n) + \frac{K_p}{N_D} v_p(n) + \frac{2\pi}{N_D} f_0 \right] \delta_n \quad (7)$$

$$f_5(n) = 2\pi f_{ref} \delta_n \quad (8)$$

4.2 Verification of the Locking Property

In this Section, the verification of the locking property of the PLL is discussed in detail. If the phase difference between the reference signal and the feedback one is within the tolerated bound $[-0.2^\circ, 0.2^\circ]$, then the PLL is “locked”, otherwise the PLL is “not locked”. The PLL locking property in ESRE form is given in Equation (9).

$$\text{If}(|\phi_v(n) - \phi_{ref}(n)| < 0.2^\circ, \text{PLL locked}, \text{PLL not locked}) \quad (9)$$

Figures 4 and 5 depict the analysis of the locking property of the CP-PLL without imperfections for the design parameters listed in Table 1. We notice that the VCO frequency stabilizes to 27 MHz after about 756 cycles which correspond also to a zero phase difference between the reference

signal and the feedback signal. However, such analysis can only be performed for one possible initial conditions vector which is in this case $x = [0.35 \ 0.01 \ -0.01 \ -0.3 \ 0]$. We now verify the locking property of the PLL using our ESRE based Qualitative Simulation algorithm. The verification is performed for a range of the voltage nodes initial conditions ($v_i(0) \in [0.34, 0.36]$ and $v_p(0), v_{p1}(0) \in [-0.01, 0.01]$), and the feedback phase ϕ_v . The initial reference phase $\phi_{ref}(0)$ is fixed to zero.

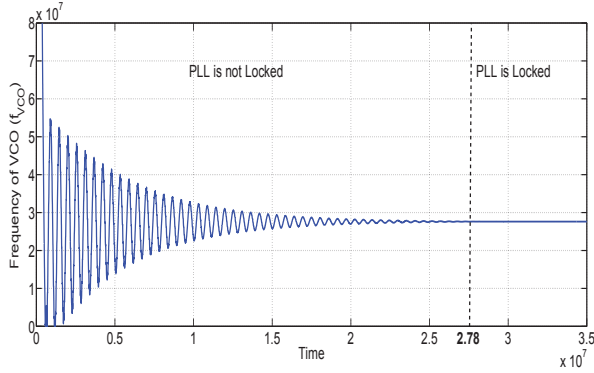


Figure 4: Frequency of the VCO

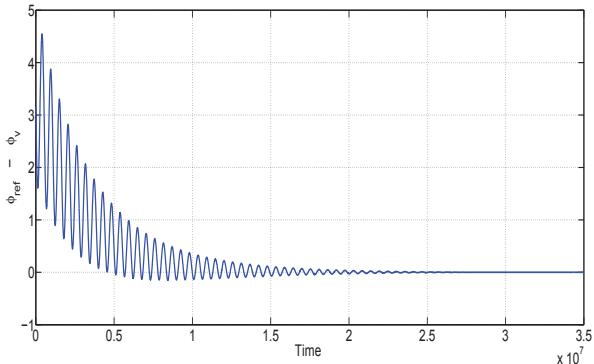


Figure 5: Phase Difference Between the Reference and Feedback Signals

Figure 6 shows the upper and the lower bounds of the VCO phase state variable. We remark that the phase is always between 0 and 2π because the reference and feedback phase are reset to $\phi = \phi - 2\pi$. Thus, the same phase crossing can be used in the next cycle. One of the advantages of our methodology is verifying the PLL locking property based only on the phase difference ($\phi_{ref} - \phi_v$) state variable. In other words, if the upper and the lower bounds of the phase difference are in the tolerated region $[-0.2^\circ, 0.2^\circ]$, then we claim that the PLL is locked. A comparison of the required number of cycles ($t_{cycle} = \frac{1}{f_{ref}}$) for the PLL locking between our Qualitative Simulation (QS) methodology (ESRE-QS) and the Monte Carlo (MC) simulation technique (MC-S) is depicted in Table 2. The simulation was carried out first in the ideal case (row 1), then with process variation (PV) in the VCO (row 2), the CP (row 3), and the LPF (row 4), finally with synchronous jitter (row 5), and accumulating jitter (row 6). Our methodology analyzes the locking property for an entire initial conditions range of the normalized phase ϕ_v . On the other hand, the Monte Carlo technique analyzes the locking property for only one possible initial condition value in each trial. Therefore, MC simulations are performed for 10 times and the maximum

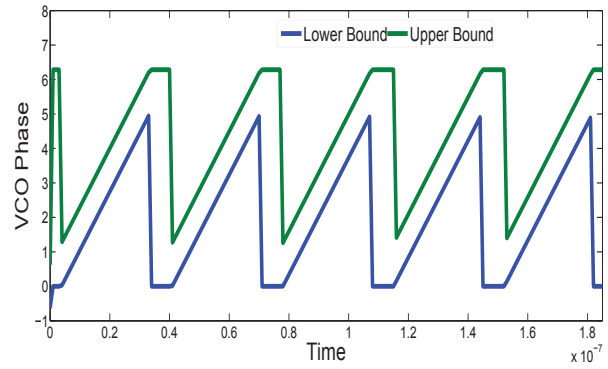


Figure 6: Bounds of the VCO Phase State variable

number of cycles found is compared with our results. In other words, the reported number of cycles for MC technique are performed by randomly choosing the initial value of $\phi_v^i \in 0.1 \times [i, i - 1]$, $i = 1, 2, \dots, 5$.

From Table 2, it can be noticed that our Qualitative Simulation approach provides a greater number of required locking cycles compared to the MC technique. Unless an infinite number of simulations is conducted, the MC simulation technique cannot provide the maximum clock cycles required for the PLL to be locked in an initial condition interval. We also see that the number of lock cycles increases with the initial phase difference ($\phi_{ref}(0) - \phi_v(0)$).

Table 1: Parameters for the CP-PLL Model [3]

Name	Value	Unit
f_{ref}	27	KHz
f_0	26.93	MHz
N_D	1000	-
K_i	200	MHz/V
K_P	25	MHz/V
I_i	[9.9, 10.1]	μA
I_p	[495, 505]	μA
C_i	25	pF
C_{p1}	6.3	pF
C_{p3}	2	pF
R_{p2}	50	K Ω
R_{p3}	8	K Ω
t_d	50	ps

Moreover, we remark that the jitter performance, and more specifically the synchronous jitter in the VCO and the reference oscillator (row 6), are of great concern in PLL locking. For these imperfections, the PLL fails to lock as represented by the “dashed entry” in Table 2. More importantly, when considering phase modulation jitter and for $\phi_v \in [-0.5, -0.4]$, our Qualitative Simulation technique finds values of ϕ_v where the PLL fails to lock, while the MC technique was not able to detect this violation (shaded region).

5. CONCLUSIONS

In this paper, a new methodology to verify the locking property in PLL designs based on Qualitative Simulation technique is proposed. The PLL design is modeled using Extended System of Recurrence Equation in the presence of accumulating jitter, synchronous jitter, and process variation. Thereafter, the verification algorithm based on Qualitative Simulation is conducted to analyze the behavior of

Table 2: Required Cycles for PLL Locking

$\phi_v \in -0.1 \times$	[5,4]		[4,3]		[3,2]		[2,1]		[1,0]	
Experiment	ESRE-QS	MC-S	ESRE-QS	MC-S	ESRE-QS	MC-S	ESRE-QS	MC-S	ESRE-QS	MC-S
No imperfections	1454	883	1389	832	1313	709	1247	676	1169	628
PV in VCO	1827	1228	1763	1117	1718	1001	1652	972	1574	946
PV in CP	1576	1066	1514	913	1479	894	1406	837	1367	779
PV in LPF	1681	1157	1629	1007	1580	994	1516	956	1489	913
Synch. jitter only	–	1763	2513	1546	2337	1481	2284	1405	2162	1379
Accum. jitter only	–	–	–	–	–	–	–	–	–	–

the PLL ESRE model over a range of initial conditions. The algorithm computes the envelopes that enclose all possible behaviors of the PLL state variables. This methodology has been gainfully employed on a third order Charge Pump PLL verification. The required locking cycles of the PLL design were computed over a range of parameters due to 0.18 μ m technology process, and a range of initial conditions of node voltages and feedback phase. Our methodology proved a better coverage compared to simulation methods. Indeed, a locking failure case was found using our approach that was not predicted by 10 Monte Carlo simulations of randomly chosen initial values of the normalized feedback phase $\phi_v \in [-0.5, -0.4]$.

As future work, we are planning to improve the imperfections models by deriving them directly from device level simulations and integrating them in the ESREs behavioral model. Moreover, we are working on the application of the proposed method to larger and more complex AMS designs and the verification of their functional properties. By doing so, we will be able to measure and assess the limitation of the proposed methodology.

6. REFERENCES

- [1] G. Al-Sammame, M. H. Zaki, and S. Tahar. A symbolic methodology for the verification of analog and mixed signal designs. In *IEEE Proceedings of the Conference on Design, Automation and Test in Europe*, pages 249–254, 2007.
- [2] A. Aloisio, R. Giordano, and V. Izzo. Phase noise issues with FPGA-embedded DLLs and PLLs in HEP applications. *IEEE Transactions on Nuclear Science*, 58(4):1664–1671, 2011.
- [3] M. Althoff, A. Rajhans, B. H. Krogh, S. Yaldiz, X. Li, and L. Pileggi. Formal verification of phase-locked loops using reachability analysis and continuization. *Communications of the ACM*, 56(10):97–104, 2013.
- [4] R. Alur. Formal verification of hybrid systems. In *IEEE Proceedings of the International Conference on Embedded Software*, pages 273–278, 2011.
- [5] M. Fulde. *Variation aware analog and mixed-signal circuit design in emerging multi-gate CMOS technologies*, volume 28 of *Springer Series in Advanced Microelectronics*. 2010.
- [6] K. Kundert. Predicting the phase noise and jitter of pll-based frequency synthesizers. Available from: *www.designers-guide.com*, 2003.
- [7] W. Li and J. Meiners. Introduction to phase-locked loop system modeling. *Analog Applications Journal, Texas Instruments Incorporated*, 2:5–10, 2000.
- [8] R. Narayanan, B. Akbarpour, M. H. Zaki, S. Tahar, and L. C. Paulson. Formal verification of analog circuits in the presence of noise and process variation. In *Proceedings of Design, Automation and Test in Europe*, pages 1309–1312, 2010.
- [9] 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 Transactions on Very Large Scale Integration Systems*, 21(10):1811–1822, 2013.
- [10] M. Nikravesh, L. A. Zadeh, and V. Korotkikh. Fuzzy partial differential equations and relational equations. *Reservoir Characterization and Modeling Series: Studies in Fuzziness and Soft Computing*, 142, 2004.
- [11] A. Singh and P. Li. On behavioral model equivalence checking for large analog/mixed signal systems. In *IEEE Proceedings of the International Conference on Computer-Aided Design*, pages 55–61, 2010.
- [12] S. Steinhorst and L. Hedrich. Improving verification coverage of analog circuit blocks by state space-guided transient simulation. In *IEEE Proceedings of International Symposium on Circuits and Systems*, pages 1811–1822, 2010.
- [13] Z. Wang. Runtime verification of analog and mixed signal designs. Master’s thesis, Concordia University, 2009.
- [14] M. H. Zaki, G. Al-Sammame, S. Tahar, and G. Bois. Combining symbolic simulation and interval arithmetic for the verification of ams designs. In *IEEE Proceedings of International Conference on Formal Methods in Computer Aided Design*, pages 207–215, 2007.
- [15] M. H. Zaki, S. Tahar, and G. Bois. Formal verification of analog and mixed signal designs: A survey. *Microelectronics Journal*, 39(12):1395–1404, 2008.