DIFFERENTIAL EQUATIONS
AND
CONTROL PROCESSES
N. 1, 2019
Electronic Journal,
reg. N ФС77-39410 at 15.04.2010
ISSN 1817-2172

http://diffjournal.spbu.ru/
e-mail: jodiff@mail.ru

Applications to physics, electrotechnics, and electronics

Comments on van Paemel's mathematical model of charge-pump phase-locked loop

Kuznetsov N.V.[1,2], Yuldashev M.V.[1], Yuldashev R.V.[1], Blagov M.V.[1,3], Kudryashova E.V.[1], Kuznetsova O.A.[1], Mokaev T.N.[1]

[1] Saint-Petersburg State University, dep. of Applied Cybernetics, St. Petersburg, Russia, nkuznetsov239@gmail.com

[2]Institute for Problems in Mechanical Engineering of the Russian Academy of Sciences, St.Petersburg, Russia

[3]University of Jyväskylä, Dept. of Mathematical Information Technology, Finland

Abstract

The charge-pump phase-locked loop (CP-PLL) is one of widely used types of the phase-locked loop (PLL). A PLL is essentially nonlinear control system and its nonlinear analysis is a challenging task. Recently, we found some flaws in the well-known and frequently cited article “Analysis of a charge-pump PLL: A new model” published by M. van Paemel in the IEEE Transactions on Communications journal. In the present brief note the corresponding numerical and analytical examples are provided and the ways to correct the flaws are discussed.

Keywords: control of oscillators, synchronization, charge-pump phase-locked loop, nonlinear dynamical model

1 Introduction

The charge-pump phase-locked loop (CP-PLL) is one of widely used types of the phase-locked loop [20, 16, 7]. Nowadays it is widely used for the control of oscillators and frequency synthesis in computer architectures (see, e.g. [18]). A PLL is essentially nonlinear control system and its nonlinear analysis is a challenging task [1, 13, 12]. The first dynamical models of CP-PLL (see, e.g. [6, 8]) were linear and took into account only linearized approximation of the phase detector dynamics in continuous time. M. van Paemel's article [19] was the first one where a nonlinear mathematical model of second-order CP-PLL was derived. While approximate models are useful for analysis of small frequency deviations between voltage controlled oscillator (VCO) and reference (Ref) signals, van Paemel's model is exact and can also be used for studying out-of-lock behavior. In the present brief note we reveal and suggest a way to avoid shortcomings in van Paemel's model and discuss the corresponding numerical examples.

2 Revised discrete-time model of CP-PLL

Below we suggest a revised discrete-time nonlinear mathematical model of CP-PLL in which discovered shortcomings are fixed:

$$\begin{aligned}
\tau(k+1) &= \begin{cases}
\frac{-b+\sqrt{b^2-4ac}}{2a}, & \tau(k)\geq 0, \quad c \leq 0, \\
\frac{1}{\omega_{\text{vco}}^{\text{free}}+K_v v(k)} - T + (\tau(k) \bmod T), & \\
 & \tau(k)\geq 0, \quad c>0, \quad k=0,1,2,... \\
l_b - T, & \tau(k)<0, \quad l_b \leq T, \\
\frac{-b+\sqrt{b^2-4ad}}{2a}, & \tau(k)<0, \quad l_b > T,
\end{cases} \\
v(k+1) &= v(k) + \frac{I_p}{C}\tau(k+1).
\end{aligned} \tag{1}$$

$$
\begin{aligned}
&a = \frac{K_v I_p}{2C},\\
&b = b(v(k)) = \omega_{\text{vco}}^{\text{free}} + K_v v(k) + K_v I_p R_2,\\
&c = c(\tau(k), v(k)) = (T - (\tau(k) \bmod T)) \left(\omega_{\text{vco}}^{\text{free}} + K_v v(k)\right) - 1,\\
&l_b = l_b(\tau(k), v(k)) = \tfrac{1 - S_{l_a}}{K_v v(k) + \omega_{\text{vco}}^{\text{free}}},\\
&S_{l_a} = S_{l_a}(\tau(k), v(k)) = S_{l_k} \bmod 1,\\
&S_{l_k} = S_{l_k}(\tau(k), v(k)) = -\left(K_v v(k) - I_p R_2 K_v + \omega_{\text{vco}}^{\text{free}}\right)\tau(k) + K_v I_p \tfrac{\tau(k)^2}{2C},\\
&d = d(v(k)) = S_{l_a} + T(K_v v(k) + \omega_{\text{vco}}^{\text{free}}) - 1.
\end{aligned}
$$

Here $v(0)$ and $\tau(0)$ are initial conditions (van Paemel's notation is used). VCO frequency is $f_{\text{vco}} = \omega_{\text{vco}}^{\text{free}} + K_v v_c$, and $\omega_{\text{vco}}^{\text{free}}$ is a free-running (quiescent) frequency (in van Paemel's paper $\omega_{\text{vco}}^{\text{free}} = 0$). Meaning of other parameters is explained in [19]. Remark that following the ideas from [19, 2, 5], the number of parameters in (1) can be reduced to two (α and β) by the following change of variables:

$$
\begin{aligned}
&s(k) = \frac{\tau(k)}{T},\ \omega(k) = T\left(\omega_{\text{vco}}^{\text{free}} + K_v v(k)\right) - 1,\\
&\alpha = K_v I_p T R_2,\ \beta = \frac{K_v I_p T^2}{2C}.
\end{aligned}
\tag{2}
$$

If at some point VCO becomes overloaded one should stop simulation or use another set of equations, based on the ideas similar to (34) and (35) in [19]. The overload conditions are the following

$$
\begin{aligned}
&\tau(k) > 0 \text{ and } v(k) + \frac{\omega_{\text{vco}}^{\text{free}}}{K_{\text{vco}}} - \frac{I_p}{C}\tau_k < 0,\\
&\tau(k) < 0 \text{ and } v(k) + \frac{\omega_{\text{vco}}^{\text{free}}}{K_{\text{vco}}} - I_p R_2 < 0.
\end{aligned}
\tag{3}
$$

3 Comparison of the models. Numerical examples

Two following examples demonstrate that algorithm and formulas suggested by M. van Paemel should be used carefully for simulation even inside the allowed area of parameters (see Fig. 18 and Fig. 22 in the original paper [19]). The main idea of Example 1 was discussed by O. Feely [5]. However they considered only near-locked state and, thus, did not notice similar problems with out-of-lock behavior. Example 2 and Example 3 demonstrate problems with out-of-lock behavior, which were not discussed before.

3.1 M. van Paemel's model

3.1.1 Example 1

Consider the following set of parameters and initial state:

$$\begin{aligned} &R_2 = 0.2; C = 0.01; K_v = 20; I_p = 0.1; T = 0.125;\\ &\tau(0) = 0.0125; v(0) = 1. \end{aligned} \tag{4}$$

The calculation of normalized parameters (equations (27)–(28) and (44)–(45) in [19])

$$\begin{aligned} K_N &= I_p R_2 K_v T = 0.05,\\ \tau_{2N} &= \frac{R_2 C}{T} = 0.016,\\ F_N &= \frac{1}{2\pi}\sqrt{\frac{K_N}{\tau_{2N}}} \approx 0.2813,\\ \zeta &= \frac{\sqrt{K_N \tau_{2N}}}{2} \approx 0.0141, \end{aligned} \tag{5}$$

shows that parameters (4) correspond to the allowed area (see equations (46)–(47) in [19]):

$$\begin{aligned} F_N &< \frac{\sqrt{1+\zeta^2}-\zeta}{\pi} \approx 0.3138,\\ F_N &< \frac{1}{4\pi\zeta} \approx 5.6438. \end{aligned} \tag{6}$$

The allowed area and the corresponding parameters are shown in Fig. 1 (see Fig. 18 and Fig. 22 in [19]).

Now we use the flowchart in Fig. 2 (Fig. 10 in [19]) to compute $\tau(1)$ and $v(1)$. Since $\tau(0) > 0$ and $\tau(0) < T$, we proceed to *case 1)*[1] and the corresponding relation for $\tau(k+1)$ (equation (7) in [19]) takes the form

$$\tau(k+1) = \frac{-I_p R_2 - v(k) + \sqrt{(I_p R_2 + v(k))^2 - \frac{2I_p}{C}(v(k)(T-\tau(k)) - \frac{1}{K_v})}}{\frac{I_p}{C}}. \tag{7}$$

However, the expression under the square root in (7) is negative:

$$(I_p R_2 + v(0))^2 - \frac{2I_p}{C}(v(0)(T-\tau(0)) - \frac{1}{K_v}) = -0.2096 < 0. \tag{8}$$

Therefore the algorithm is terminated with an error.

[1] Hereinafter *case 1)–6)* corresponds to original paper by M. van Paemel

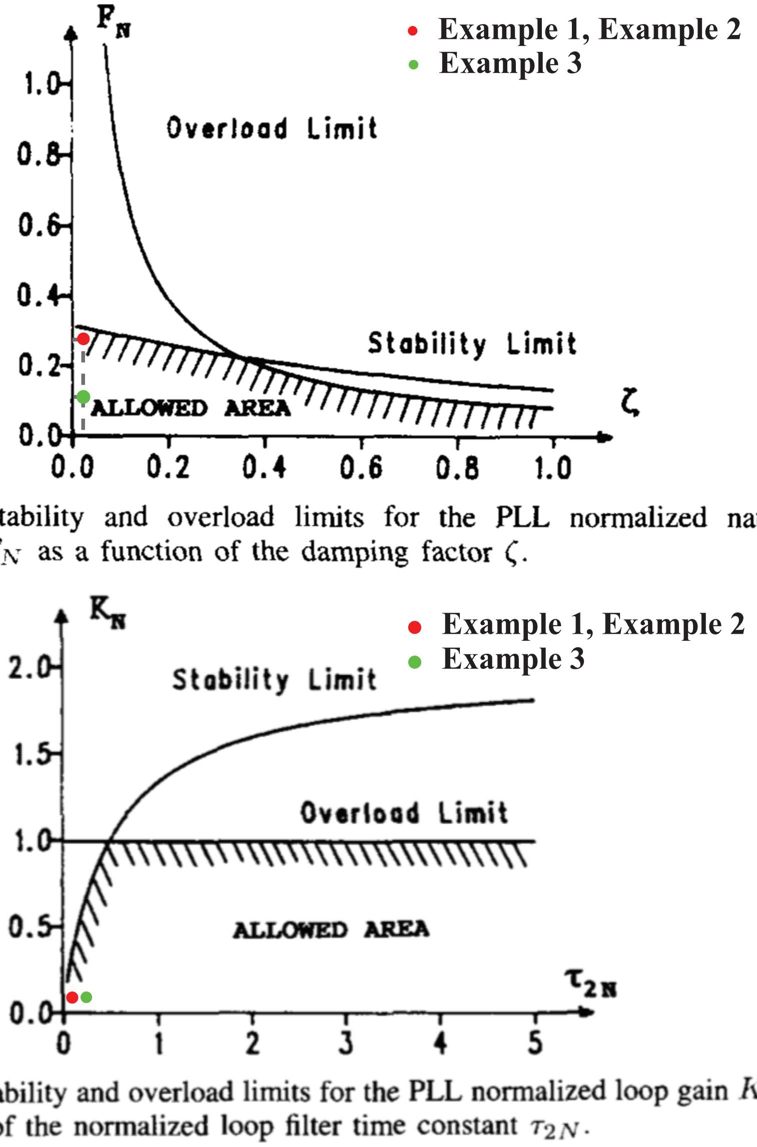

Fig. 22. Stability and overload limits for the PLL normalized natural frequency F_N as a function of the damping factor ζ.

Fig. 18. Stability and overload limits for the PLL normalized loop gain K_N as function of the normalized loop filter time constant τ_{2N}.

Figure 1: Parameters for Example 1, Example 2, and Example 3 correspond to the *allowed area* in van Paemel's paper [19] (see Fig. 18 and Fig. 22).

3.1.2 Example 2

Consider the same parameters as in Example 1, but $\tau(0) = -0.098$:

$$\begin{gathered} R_2 = 0.2; C = 0.01; K_v = 20; I_p = 0.1; T = 0.125; \\ \tau(0) = -0.098; \quad v(0) = 1. \end{gathered} \tag{9}$$

In this example eqautions (5), (6), and Fig. 1 are the same as in Example 1, i.e. we are in the "allowed area". Now we compute $\tau(1)$ and $v(1)$ following the flowchart in Fig. 2. Since $\tau(0) < 0$, we proceed to *case 2)* and the corresponding

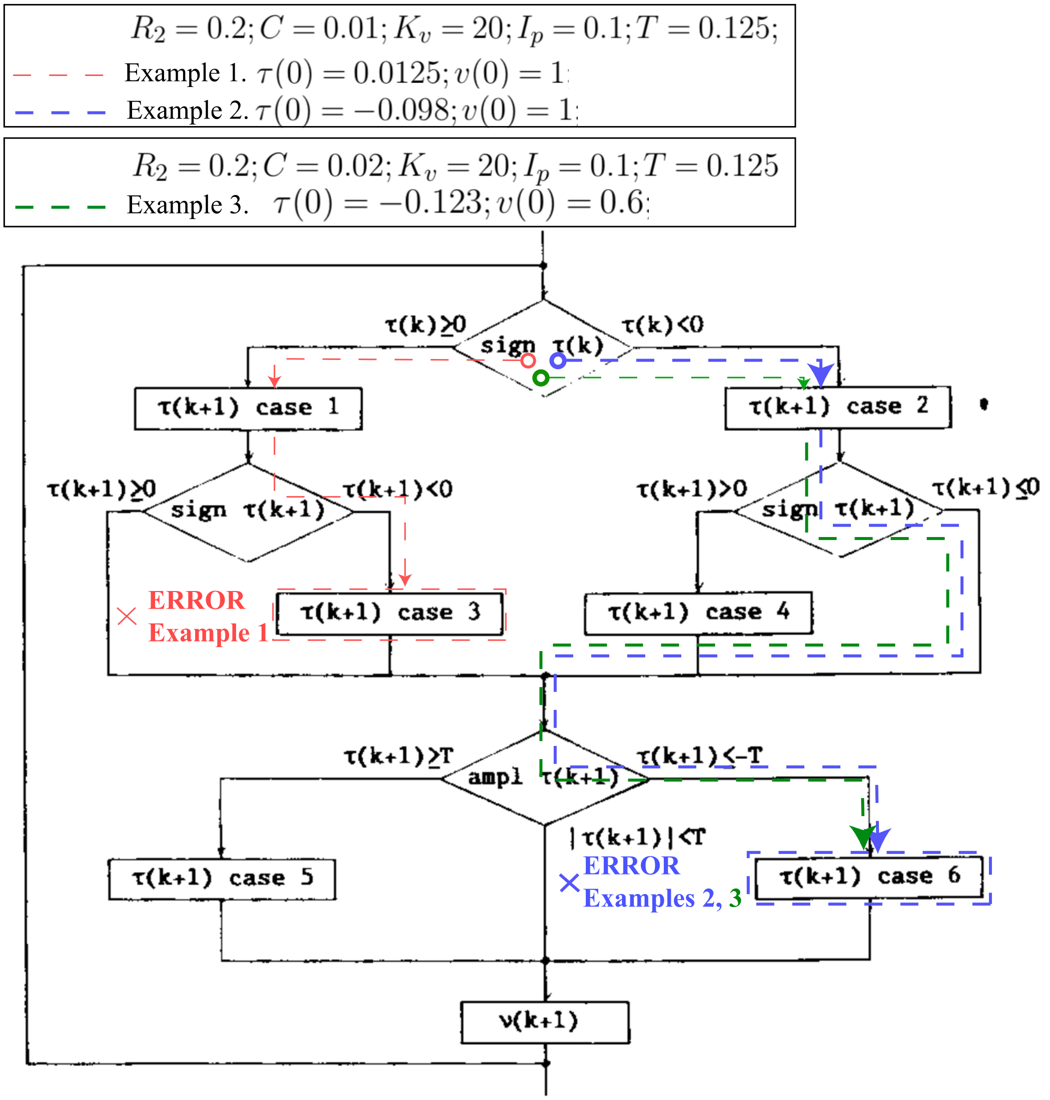

Figure 2: Demonstration of Example 1, Example 2, and Example 3 in the flowchart of the algorithm (see Fig. 10 in [19]).

equation for $\tau(k+1)$ (equation (9) in [19]) is as follows

$$\begin{aligned}&\tau(1) = \frac{\frac{1}{K_v} - I_p R_2 \tau(0) - \frac{I_p \tau(0)^2}{2C}}{v(0)} - T + \tau(0) = -0.21906,\\ &-0.2191 < -T = -0.125.\end{aligned} \tag{10}$$

This shows cycle-slipping (out of lock). According to the flowchart in Fig. 2 (see Fig. 10 in [19]), we should proceed to *case 6)* and recalculate $\tau(1)$. The first step of *case 6)* is to calculate $t_1, t_2, t_3, ...$ (equations (16) and (17) in [19]).

We have

$$\begin{aligned}
t_n &= \frac{v_{n-1} - I_pR_2 - \sqrt{(v_{n-1} - I_pR_2)^2 - 2\frac{I_p}{C} \cdot \frac{1}{K_v}}}{\frac{I_p}{C}}, \\
v_n &= v_{n-1} - \frac{I_p}{C}t_n, \\
v_0 &= v(k-1).
\end{aligned} \tag{11}$$

Since $k = 0$, we have

$$\begin{aligned}
t_1 &= \frac{v_0 - I_pR_2 - \sqrt{(v_0 - I_pR_2)^2 - 2\frac{I_p}{C} \cdot \frac{1}{K_v}}}{\frac{I_p}{C}}, \\
v_1 &= v_0 - \frac{I_p}{C}t_1, \\
v_0 &= v(-1).
\end{aligned} \tag{12}$$

However, $v(-1)$ does not make a sense and the algorithm terminates with an error.[2] Even if we suppose that it is a mistake and $v_0 = v(0)$, then the relation under the square root becomes negative:

$$(v(0) - I_pR_2)^2 - 2\frac{I_p}{CK_v} = -0.0396 < 0. \tag{13}$$

Therefore the algorithm is terminated with an error. Note that modification of *case 2)* corresponding to VCO overload (equation (35) in [19]) can not be applied here since $v(0) > I_pR_2$ (no overload) and $v(1)$ is not computed because of error.

3.1.3 Example 3

Consider parameters

$$\begin{aligned}
&\tau(0) = -0.123; \quad v(0) = 0.6, \\
&R_2 = 0.2; C = 0.02; K_v = 20; I_p = 0.1; T = 0.125.
\end{aligned} \tag{14}$$

Similar to (5) and (6) we have

$$\begin{aligned}
&K_N = 0.05, \quad \tau_{2N} = 0.032, \\
&F_N \approx 0.1989, \quad \zeta = 0.02,
\end{aligned} \tag{15}$$

[2] However, this can be fixed by setting $v(-1) = v(0) - \frac{I_p}{C}\tau(0)$.

$$\begin{aligned} &F_N < \frac{\sqrt{1+\zeta^2}-\zeta}{\pi} \approx 0.3120, \\ &F_N < \frac{1}{4\pi\zeta} \approx 3.9789. \end{aligned} \tag{16}$$

Therefore parameters (14) correspond to the allowed area in Fig. 1 (equations (46)-(47), Fig. 18 and Fig. 22 in [19]).

Now we compute $\tau(1)$ and $v(1)$ following the flowchart in Fig. 2. Since $\tau(0) < 0$, we proceed to *case 2)* and the corresponding equation for computing $\tau(k+1)$ (equation (9) in [19]) is as follows

$$\begin{aligned} &\tau(1) = \frac{\frac{1}{K_v} - I_p R_2 \tau(0) - \frac{I_p \tau(0)^2}{2C}}{v(0)} - T + \tau(0) \approx -0.224, \\ &-0.224 < -T = -0.125. \end{aligned} \tag{17}$$

The last inequality indicates a cycle-slipping (out of lock). According to the flowchart in Fig. 2 (see Fig. 10 in [19]), one proceeds to *case 6)* and recalculates $\tau(1)$. The first step of *case 6)* is to calculate $t_1, t_2, t_3, ...$ using (11) (see equations (16) and (17) in [19]) until $t_1 + t_2 + \ldots + t_n > |\tau(0)|$. Even if we suppose $v(-1) = v(0) - \frac{I_p}{C}\tau(0)$, we have

$$\begin{aligned} &t_1 = 0.0463, \ v_1 = 1.215; \\ &t_2 = 0.0618, \ v_2 = 0.983; \\ &t_1 + t_2 = 0.1081 < |\tau(0)| = 0.123. \end{aligned} \tag{18}$$

However, t_3 can not be computed because the relation under the square root in (11) is negative:

$$(v_2 - I_p R_2)^2 - 2\frac{I_p}{C} \cdot \frac{1}{K_v} \approx -0.0726. \tag{19}$$

Therefore the algorithm is terminated with an error.

3.2 Revised model

Consider the behaviour of revised model (1) for the above examples. In all three examples we have $\omega_{\text{vco}}^{\text{free}} = 0$.

3.2.1 Example 1

By (1) and (4) we can calculate a value of c:

$$c = (T - (\tau(0) \bmod T))K_v v(0) - 1 = 1.2500, \tag{20}$$

and since $\tau(0) \geq 0$ and $c > 0$, we get

$$\begin{aligned} \tau(1) &= \frac{1}{K_v v(0)} - T + (\tau(0) \bmod T) = -0.0625, \\ v(1) &= v(0) + \frac{I_p}{C}\tau(1) = 0.3750. \end{aligned} \tag{21}$$

Note that in this example there is no VCO overload (no saturation) since the filter output (VCO input) is positive.

3.2.2 Example 2

By (1) and (9) we have $l_b \approx 0.0059$. Since $l_b \leq T$, we have $\tau(1) \approx -0.1191, v(1) \approx -0.1906$. In this example the VCO is overloaded. Model (1) correctly detects overload by (3):

$$v(1) + \frac{\omega_{\text{vco}}^{\text{free}}}{K_{\text{vco}}} - I_p R_2 \approx -0.2106 < 0, \tag{22}$$

and stops simulation.

3.2.3 Example 3

Note, that in Example 3 VCO is not overloaded since the filter output (VCO input) is positive. Equations (1) allow one to correctly calculate next step:

$$\tau(1) = 0, \quad v(1) = 10. \tag{23}$$

Therefore it is correct.

4 Conclusions

The problem of flowchart in Fig. 2 is that the sign of $\tau(k+1)$ computed by *case 1)* is used to decide whether *case 3)* can be used or not. Similarly, *case 2)* always precedes *cases 4),5),6)*, which may lead to errors. However, it is possible to use $\tau(k)$ and $v(k)$ to select a correct formula for $\tau(k+1)$ (see (1)). This allows one to avoid computing square roots of negative numbers, to reduce number of cases from 6 to 4, and to apply the methods of the theory of discrete time dynamical systems (see, e.g. [5]).

The correctness of the revised model was verified by simulation in Matlab Simulink. CP-PLL circuit level model in Matlab Simulink was compared with van Paemel's and revised mathematical models. For the following set of parameters ($\tau(0) = 0; v(0) = 10; R_2 = 1000; C = 10^{-6}; K_v = 500; I_p = 10^{-3}; T = 10^{-3}; \tau_{2N} = 1; K_N = 0.5; F_N = 0.1125; \zeta = 0.3536$) all three models show consistent results. Much more detailed analysis of the errors and possible ways of their rectification are included in https://arxiv.org/pdf/1901.01468.pdf as a preliminary version of much larger paper. The revised model can be used for the study of parameters' values for which CP-PLL achieves lock, including analytical estimation of the hold-in range, pull-in ranges, and pull-in time [11, 13, 3].

At present there are several attempts to generalize equations derived in [19] for higher-order CP-PLL (see, e.g.[10, 17, 4, 9, 14, 15]), however the resulting transcendental equations can not be solved analytically without using approximations.

Acknowledgments

The work is supported by the Russian Science Foundation project 19-41-02002. The authors would like to thank Mark van Paemel for valuable comments on this note.

Note that we have notified the editorial board of IEEE TCOM journal (in which the original paper [19] was published) of discovered shortcomings. However, the editorial board did not offer any means of informing the readership of the journal as well as IEEE community.

References

[1] D. Abramovitch. Phase-locked loops: A control centric tutorial. In *American Control Conf. Proc.*, volume 1, pages 1–15. IEEE, 2002.

[2] P. Acco. *Study of the loop ‘a phase lock: Hybrid aspects taken into account.* PhD thesis, Toulouse, INSA, 2003.

[3] R.E. Best, N.V. Kuznetsov, G.A. Leonov, M.V. Yuldashev, and R.V. Yuldashev. Tutorial on dynamic analysis of the Costas loop. *Annual Reviews in Control*, 42:27–49, 2016.

[4] C. Bi, P.F. Curran, and O. Feely. Linearized discrete-time model of higher order charge-pump plls. In *Circuit Theory and Design (ECCTD), 2011 20th European Conference on*, pages 457–460. IEEE, 2011.

[5] P.F. Curran, C. Bi, and O. Feely. Dynamics of charge-pump phase-locked loops. *International Journal of Circuit Theory and Applications*, 41(11):1109–1135, 2013.

[6] F. Gardner. Charge-pump phase-lock loops. *IEEE Transactions on Communications*, 28(11):1849–1858, 1980.

[7] F.M. Gardner. *Phaselock techniques.* John Wiley & Sons, New York, 1966.

[8] F.M. Gardner. *Phaselock techniques.* John Wiley & Sons, 2005.

[9] C. Hangmann, C. Hedayat, and U. Hilleringmann. Stability analysis of a charge pump phase-locked loop using autonomous difference equations. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 61(9):2569–2577, 2014.

[10] P.K. Hanumolu, M. Brownlee, K. Mayaram, and Un-Ku Moon. Analysis of charge-pump phase-locked loops. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 51(9):1665–1674, 2004.

[11] N.V. Kuznetsov, G.A. Leonov, M.V. Yuldashev, and R.V. Yuldashev. Rigorous mathematical definitions of the hold-in and pull-in ranges for phase-locked loops. *IFAC-PapersOnLine*, 48(11):710–713, 2015.

[12] G.A. Leonov and N.V. Kuznetsov. *Nonlinear mathematical models of phase-locked loops. Stability and oscillations.* Cambridge Scientific Publishers, 2014.

[13] G.A. Leonov, N.V. Kuznetsov, M.V. Yuldashev, and R.V. Yuldashev. Hold-in, pull-in, and lock-in ranges of PLL circuits: rigorous mathematical definitions and limitations of classical theory. *IEEE Transactions on Circuits and Systems–I: Regular Papers*, 62(10):2454–2464, 2015.

[14] S. Milicevic and L. MacEachern. Time evolution of the voltage-controlled signal in charge pump pll applications. In *Microelectronics, 2008. ICM 2008. International Conference on*, pages 413–416. IEEE, 2008.

[15] S. Sancho, A. Súarez, and J. Chuan. General envelope-transient formulation of phase-locked loops using three time scales. *IEEE Transactions on Microwave Theory and Techniques*, 52(4):1310–1320, 2004.

[16] V.V. Shakhgil'dyan and A.A. Lyakhovkin. *Fazovaya avtopodstroika chastoty (in Russian)*. Svyaz', Moscow, 1966.

[17] B.I. Shakhtarin, A.A. Timofeev, and V.V. Sizykh. Mathematical model of the phase-locked loop with a current detector. *Journal of Communications Technology and Electronics*, 59(10):1061–1068, 2014.

[18] K. Shu and E. Sanchez-Sinencio. *CMOS PLL synthesizers: analysis and design*. Springer, 2005.

[19] M. van Paemel. Analysis of a charge-pump PLL: A new model. *IEEE Transactions on communications*, 42(7):2490–2498, 1994.

[20] A. Viterbi. *Principles of coherent communications.* McGraw-Hill, New York, 1966.