System Level Verification of Phase-Locked Loop using Metamorphic Relations

Muhammad Hassan[1,2] Daniel Große[1,3] Rolf Drechsler[1,2]
[1]Cyber-Physical Systems, DFKI GmbH, 28359 Bremen, Germany
[2]Institute of Computer Science, University of Bremen, 28359 Bremen, Germany
[3]Institute for Complex Systems, Johannes Kepler University, 4040 Linz, Austria
muhammad.hassan@dfki.de daniel.grosse@jku.at drechsle@informatik.uni-bremen.de

***Abstract*—In this paper we build on *Metamorphic Testing* (MT), a verification technique which has been employed very successfully in the software domain. The core idea is to uncover bugs by relating consecutive executions of the program under test. Recently, MT has been applied successfully to the verification of *Radio Frequency* (RF) amplifiers at the system level as well. However, this is clearly not sufficient as the true complexity stems from *Analog/Mixed-Signal* (AMS) systems.**

In this paper, we go beyond pure analog systems, i.e. we expand MT to verify AMS systems. As a challenging AMS system, we consider an industrial PLL. We devise a set of eight generic *Metamorphic Relations* (MRs). Theses MRs allow to verify the PLL behavioral at the component level and at the system level. Therefore, we have created MRs considering analog-to-digital as well as digital-to-digital behavior. We found a critical bug in the industrial PLL which clearly demonstrates the quality and potential of MT for AMS verification.

I. Introduction

The interaction of analog structures and digital logic has increased significantly in modern *Analog/Mixed-Signal* (AMS) systems. For this reason and in combination with the always increasing complexity, their design and verification faces several challenges: (1) Quick and fast modeling to make the right design decisions, (2) fast simulation of application scenarios, (3) cross-level approaches and (4) novel cost-effective verification methods. Some of these issues are addressed by system-level solutions which have made there way into industry. In particular, SystemC AMS-based modeling and verification is heavily used today [1], [2], [3], [4], [5], [6]. However, to perform the verification the expected behavior has to be checked. Since formalizing the analog behavior is non-trivial and very time-consuming, manual (and most often visual) analysis of the waveforms is carried out in industrial practice.

To overcome this problem, we look into *Metamorphic Testing* (MT) in this paper [7], [8]. MT has been first considered in the software domain and a major advantage of this technique is that no reference model/value is needed, which has to be there in classical software testing. The core idea of MT is to relate consecutive executions of the program under test by so-called *Metamorphic Relations* (MRs). Let us consider an example: Assume the goal is to test a program implementing the sine function `prg_sin(x)`. Based on the well known trigonometric properties we can use $sin(x) = sin(180 - x)$ as MR. Instead of checking the expected output for a concrete input to the program, we can run the program for an input x_1 and afterwards for the input $x_2 = 180 - x_1$ (follow up test-case). Now, using the above MR we just have to check if `prg_sin(`x_1`)` $=$ `prg_sin(`x_2`)`. If this is not the case, we have found a bug.

Contribution: Recently, MT has been successfully applied to the verification of *Radio Frequency* (RF) amplifiers at the system level as well [9]. However, this is clearly not sufficient as the true complexity stems from AMS systems. In this paper, we go beyond pure analog systems, i.e. we expand MT to verify AMS systems [1]. We make the following contributions:

1) We consider an industrial *Phase-Locked Loop* (PLL) AMS system and devise a set of eight generic MRs.
2) We show that these MRs allow to verify the PLL behavior at the component level and at the system level. Therefore, we have created MRs considering analog-to-digital, digital-to-analog, as well as digital-to-digital behavior.
3) Besides successful verification of a broad spectrum of tests, the proposed MRs can be easily used to derive follow-up testcases at different levels and hence improve the verification. Please note that these test-cases and the MRs can be re-used at lower abstraction levels.
4) We found a critical bug in the industrial PLL which clearly demonstrates the quality and potential of MT for AMS verification.

II. Related Work

At the heart of MT are MRs – the core properties, however, are different from classical AMS *Assertion Based Verification* (ABV) techniques, such as [10], [11], [12], [13], [14], [15]. All the aforementioned works require a reference model for assertion definition. On the contrary, our proposed MT-based approach enables verification without reference models.

Furthermore, MT has been mainly introduced in the software domain, hence, the closest to our work in hardware context are [16] and [9]. [16] targets the problem of digital hardware fault-tolerance and not AMS verification, and [9] only considers RF amplifiers instead of a complete AMS system.

III. Metamorphic Testing for PLLs

In this section, we first start by introducing the industrial PLL. Afterwards, we briefly introduce the MT principle for mixed-

This work was supported in part by the German Federal Ministry of Education and Research (BMBF) within the project AUTOASSERT under contract no. 16ME0117.

[1]Visit http://www.systemc-verification.org/ams for our most recent approaches.

signal interactions. At the end, we identify the generic MRs for PLLs.

A. Phase Locked Loop

A PLL operates by comparing an input frequency with the system's clock frequency and subsequently adjusting its output to match the input. It comprises of a *Phase Frequency Detector* (PFD), *Charge Pump* (CP), *Loop Filter* (LF), *Voltage Controlled Oscillator* (VCO), and a *Frequency Divider* (FD). The PFD compares the phase of input signals and accordingly sends two signals UP/DN to CP. As a result, the CP generates pulses of positive/negative currents. This current goes to LF which generates a control voltage signal. The control voltage is applied to the VCO which generates the output frequency. The FD takes the output frequency signal and divides it by a factor *N*. The resulting signal goes back to PFD. PLLs are widely used in carrier recovery, clock recovery, frequency synthesis, and clock synchronization etc. In this paper, we use a configurable system-level model of PLL [17], [18] provided by our industrial partner. The high-level PLL block diagram is shown in Fig. 1[2] and has the following specifications:

- Free running frequency (F_{osc}) = 2.39 GHz
- vdd = 3.3 V, vcm = 1.65 V
- CP current_up/dn = 100μA
- VCO Gain = $36.36e^6$
- Frequency Divider N = 2450

The model is implemented in SystemC AMS using *Timed Data Flow* (TDF) and *Electrical Linear Network* (ELN) *Models of Computation* (MoC) for different building blocks. The model also uses *Discreet Event* (DE) simulation. The simulations are carried out using the commercial tool COSIDE [19]. A test stimuli is provided with the model to verify the functional correctness.

B. MT Principle for Mixed Signal Interactions

Mixed signal interactions require understanding of both the analog/digital inputs and the corresponding digital/analog outputs w.r.t the core properties of the DUV. One such property highlighting these interactions is related to CP in PLL where the digital inputs vary the analog current on the output of the CP. More concretely, let's consider a CP with the following behavior:

$$icp = \begin{cases} +100e^{-6} & \text{UP = 1, DN = 0} \\ -100e^{-6} & \text{UP = 0, DN = 1} \end{cases}$$

i.e., CP checks its input and generates a positive/negative pulse of 100e^{-6}. To verify this, a concrete MR can be: *icp[CP(*$1, 0$*)]* $= -$*icp[CP(*$0, 1$*)]* . The MR states that the output current *icp* of the first execution (LHS)[3] should always equal the negative of output current *icp* of the second execution (RHS)[4]. Let's consider this graphically: The base test-case *CP(*$1, 0$*)* is shown in Fig. 2 (top waveform) from time t = 0.0 to 0.5 μs and the corresponding *icp* is shown in Fig. 2 (bottom waveform) which is $+100e^{-6}$ A. The follow-up test-case (inverted inputs) *CP(*$0, 1$*)* is shown in Fig. 2 (middle waveform) from time t = 0.5 to 1.0 μs and the corresponding *icp* is shown in Fig. 2 (bottom waveform) which is $-100e^{-6}$ A. According to the MR above, it should hold that *icp[CP(*$1, 0$*)], i.e.,* $+100e^{-6}$, equals $-(-100e^{-6})$ which is the icp when inputs are inverted, i.e., *icp[CP(*$0, 1$*)]*. This is obviously satisfied here. In case the MR does not hold, the CP is termed buggy.

In the next section we generalize this principle and identify eight MRs for Pll and few of its components.

C. Identification of MRs for PLLs

In this section we describe eight high quality generic MRs we identified from the core properties of PLLs and its components. The analog-to-digital, digital-to-analog, and digital-to-digital interactions are considered for the MRs. As a convention, F_{osc} is the free running frequency of the PLL when either there is no input frequency or the input frequency is out of capture range. F_{DIV} is the PLL output frequency F_o divided by *N*, i.e., $F_{DIV} = \frac{F_o}{N}$.

MR1: The digital output *UP* of PFD is $True$ if the analog input frequency F_1 is higher than F_2. Similarly, the digital output *DN* of PFD is $True$ if the analog input frequency F_1 is lower than F_2. Hence, the following should always be satisfied across 4 executions of PFD with varying input frequencies F_1 and F_2:

$$UP[PFD(F_1, F_2)] + UP[PFD(F_2, F_1)] = DN[PFD(F_1, F_2)] + DN[PFD(F_2, F_1)]$$

MR2: The CP generates pulses of positive or negative currents based on its digital inputs. Let $X_1 = IN_1, IN_2$ be the base test-case, and let $X_2 = IN_2, IN_1$ be the follow-up test-case, then, the following relation should always be satisfied over 2 executions of the CP:

$$ICP[CP(X_1)] = -ICP[CP(X_2)]$$

MR3: When the PLL is not in locked state, F_o is always at free running frequency F_{osc}. Let F_1 be the base test-case and F_2 be the follow-up test-case, such that both F_1 and F_2 are not in lock range, then the following should always hold across 2 executions:

$$F_o[PLL(F_1)] - F_o[PLL(F_2)] = 0$$

MR4: In PLL locked state, F_{DIV} scales by the same constant with which the input frequency is scaled. Let F_1 be the base test-case and $F_2 = C \times F_1$ is a follow-up test case where *C* is the scaling constant, then the following should always hold:

$$C \times F_{DIV}[PLL(F_1)] = F_{DIV}[PLL(F_2)]$$

MR5: If PLL is not locked, F_o equals F_{osc}. Let F_1 be the base test-case when the PLL is in locked state such that $F_1 > Fosc$, and F_2 is a follow-up test-case such that F_2 is very high and outside PLL lock range, then the following should always hold:

$$F_{DIV}[PLL(F_1)] > F_{DIV}[PLL(F_2)]$$

MR6: If PLL is not locked, F_o equals F_{osc}. Let F_1 be the base test-case such that $F_1 < F_{osc}$, and F_2 is the follow-up test-case

[2]PFD is written as PSD in the provided model as shown.
[3]LHS: Left Hand Side.
[4]RHS: Right Hand Side.

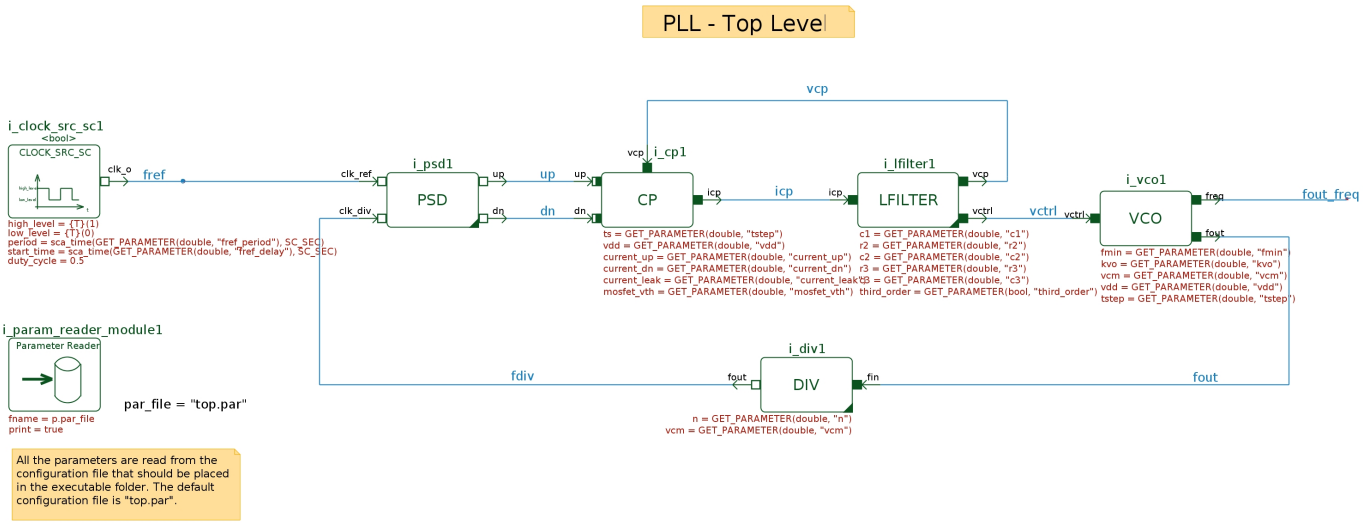

Fig. 1. PLL Top Level Diagram

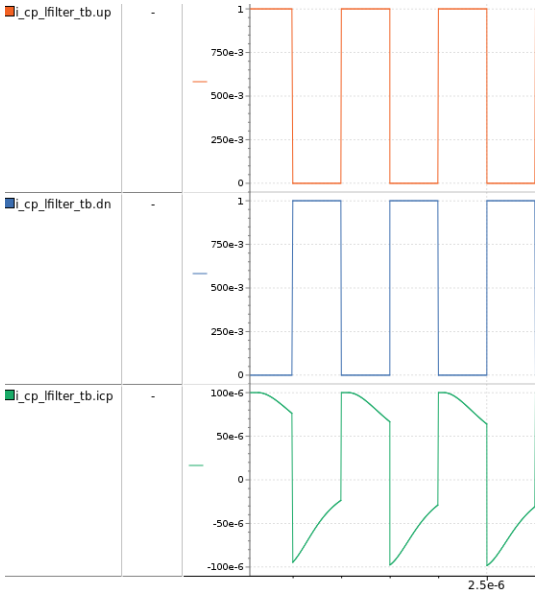

Fig. 2. Graphical illustration of MR for Charge Pump (CP)

such that F_2 is outside lock range (very high or very low), then the following should always hold:

$$F_{DIV}[PLL(F_1)] < F_{DIV}[PLL(F_2)]$$

MR7: The PLL stays in the locked state (indicated by *Lock Detector* (LD)) if the input frequency is varied inside the lock range. Let F_1 be the base test-case, F_2 and F_3 be the follow-up test-cases, such that all the frequencies keep the PLL in locked state, then the following should always hold:

$$LD[PLL(F_1)] \,\&\, LD[PLL(F_2)] \,\&\, LD[PLL(F_3)] \;=\; 1$$

MR8: F_o synchronizes to a new frequency within the lock range in a single beat-note [20]. Let F_1 be the base test-case and F_2 be the follow-up test-case such that the PLL is in locked state at both frequencies, then the following should always hold:

$$LockTime[PLL(F_1)] = LockTime[PLL(F_2)]$$

IV. Experiments

In this section, we present the experiments to demonstrate the quality and potential of MT for system-level AMS verification.

A. Overview

As already mentioned in the introduction, we use a configurable industrial system-level PLL model provided by our industrial collaboration partner. The details and specifications of the PLL have been already described in Section III-A. We use the 8 MRs devised in Section III-C to verify the PLL behavior at component level and at the system level. We have found a critical bug in the PLL design using the introduced MRs, which has escaped during the extensive verification. In the following we provide more details on the proposed MT-based approach.

B. MT-based Verification of PLL

As test stimuli we use the test-cases shipped with the model. As expected, the simulations for the set of shipped test-stimuli passes. As a next step, we employed our proposed MT-approach using the given test-stimuli as the base test-cases. The MRs from Section III-C were used to create the follow-up test-cases. Out of these, 10% follow-up test-cases covered analog-to-digital behavior at the component level and 10% follow-up test-cases covered analog-to-digital behavior at the system level. Furthermore, 10% covered digital-to-analog behavior at the component level and the remaining 70% follow-up test-cases covered the digital-to-digital behavior at the system level.

Running our MT-approach with the proposed MRs resulted in a simulation failure. More precisely, **MR4** was not satisfied: The constant factor $C = 1.01$ increased the input frequency (F_1) of the PLL (RHS of MR4 with $F_2 = C \times F_1$), from 1 MHz to 1.01 MHz, and it was expected that the divided frequency F_{DIV} will also increase by the same factor C (LHS of MR4). However, that was not the case.

Upon close inspection of the waveforms of F_{REF} and F_{DIV} (inputs of PFD), and *Fast Fourier Transform* (FFT) of F_{REF} and F_{DIV} revealed that the PLL was locking to a different very low

frequency of 50 KHz instead of 1.01 MHz. The faulty behavior is shown in Fig. 3 where F_{REF} is a high frequency signal of 1.01 MHz (top waveform) and F_{DIV} is unexpectedly low frequency signal of 50 KHz (bottom waveform). The FFT of F_{REF} and F_{DIV} signals is shown in Fig. 4 (top waveform - F_{REF}, bottom waveform - F_{DIV}). The FFT of F_{REF} shows a peak at 1.01 MHz and the FFT of F_{DIV} shows multiple peaks at various frequencies with the strongest peak at 50 KHz. Upon further investigation, we observed the **Dead-zone** effect, i.e. a **Dead-zone** was occurring in the output behavior of the PFD. A **Dead-zone** occurs when the PLL loop does not respond to small phase errors between F_{REF} and F_{DIV}. As a result, the output of CP is modulated by a signal that is a sub-harmonic of the PFD input reference frequency F_{REF}. Since this could be a low frequency signal, it would not be attenuated by LF [21]. Looking into the design of PFD revealed that there was no delay element between the *AND* gate and the *reset* pins of the Flip-flops. The design without any delay element is shown in Fig. 6. The delay element between the output of *AND* gate and the *reset* inputs of Flip-flops ensures that dead-zone effect does not happen. After insertion of the delay element, we observed the correct output behavior of the PLL and **MR4** was satisfied (cf. Fig. 5). Thf F_{REF} and F_{DIV} signals can be observed at same frequencies.

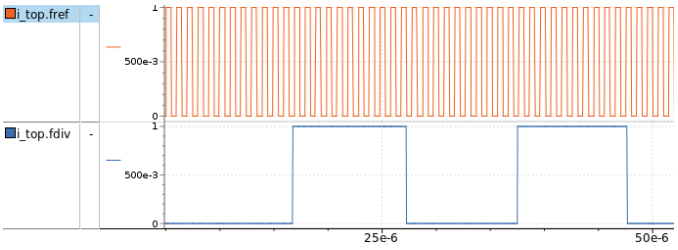

Fig. 3. PLL faulty behavior - dead zone effect revealed by MR4.

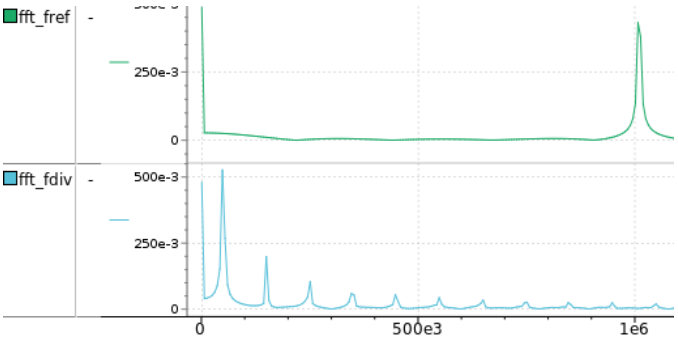

Fig. 4. FFT of PLL faulty behavior

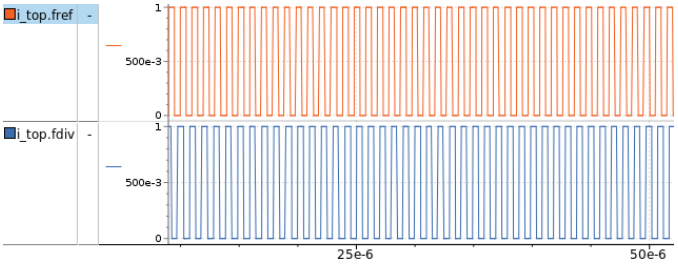

Fig. 5. PLL behavior after addition of delay element in PFD.

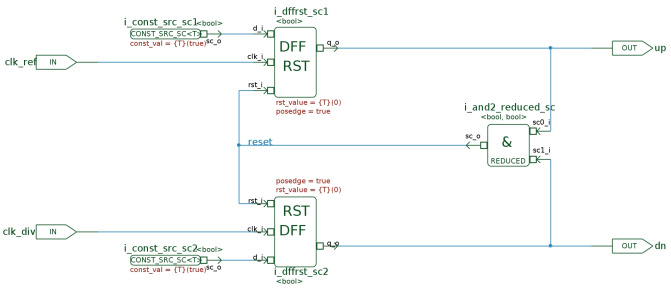

Fig. 6. Phase Frequency Detector (PFD) without a delay element

V. Conclusion

In this paper, we broadened the MT-approach to verify complex AMS systems, in particular an industrial PLL. We identified 8 high quality generic MRs to verify the PLL behavior at the component level and at system level which encompasses analog-to-digital, digital-to-analog, and digital-to-digital behaviors. In preliminary experiments, we found a critical bug in the PLL which demonstrates the quality and potential of MT for AMS verification.

References

[1] C. Grimm, M. Barnasconi, A. Vachoux, and K. Einwich, "An introduction to modeling embedded analog/mixed-signal systems using SystemC AMS extensions," in *DAC*, vol. 23, 2008.

[2] M. Lora, S. Vinco, E. Fraccaroli, D. Quaglia, and F. Fummi, "Analog models manipulation for effective integration in smart system virtual platforms," *TCAD*, vol. 37, no. 2, pp. 378–391, 2018.

[3] M. Hassan, D. Große, H. M. Le, T. Vörtler, K. Einwich, and R. Drechsler, "Testbench qualification for SystemC-AMS timed data flow models," in *DATE*, 2018, pp. 857–860.

[4] T. Vörtler, K. Einwich, M. Hassan, and D. Große, "Using constraints for SystemC AMS design and verification," in *DVCon Europe*, 2018.

[5] M. Hassan, D. Große, H. M. Le, and R. Drechsler, "Data flow testing for SystemC-AMS timed data flow models," in *DATE*, 2019, pp. 366–371.

[6] M. Hassan, D. Große, T. Vörtler, K. Einwich, and R. Drechsler, "Functional coverage-driven characterization of RF amplifiers," in *FDL*, 2019, pp. 1–8.

[7] S. Segura, G. Fraser, A. B. Sanchez, and A. Ruiz-Cortés, "A survey on metamorphic testing," *TSE*, pp. 805–824, 2016.

[8] T. Y. Chen, F.-C. Kuo, H. Liu, P.-L. Poon, D. Towey, T. Tse, and Z. Q. Zhou, "Metamorphic testing: A review of challenges and opportunities," *ACM Computing Surveys (CSUR)*, vol. 51, no. 1, p. 4, 2018.

[9] M. Hassan, D. Große, and R. Drechsler, "System-level verification of linear and non-linear behaviors of RF amplifiers using metamorphic relations," in *ASP-DAC*, 2021.

[10] Z. J. Dong, M. H. Zaki, G. Al Sammane, S. Tahar, and G. Bois, "Checking properties of PLL designs using run-time verification," in *ICM*, 2007, pp. 125–128.

[11] J. Havlicek and S. Little, "Realtime regular expressions for analog and mixed-signal assertions," in *FMCAD*, 2011, pp. 155–162.

[12] O. Maler and D. Ničković, "Monitoring properties of analog and mixed-signal circuits," *STTT*, vol. 15, no. 3, pp. 247–268, 2013.

[13] A. G. Kimura, K.-W. Liu, S. Prabhu, S. B. Bibyk, and G. Creech, "Trusted verification test bench development for phase-locked loop (PLL) hardware insertion," in *MWSCAS*, 2013, pp. 1208–1211.

[14] C. Radojicic, F. Schupfer, M. Rathmair, and C. Grimm, "Assertion-based verification of signal processing systems with affine arithmetic," in *FDL*, 2012, pp. 20–26.

[15] N. Bombieri, F. Fummi, V. Guarnieri, G. Pravadelli, F. Stefanni, T. Ghasempouri, M. Lora, G. Auditore, and M. N. Marcigaglia, "Reusing RTL assertion checkers for verification of SystemC TLM models," *JETTA*, vol. 31, no. 2, pp. 167–180, 2015.

[16] J. Liu, "Metamorphic testing and its application on hardware fault-tolerance," 2011, eCE Project Report at University of Wisconsin.

[17] A. Dias, "Phase locked loop simulator in SystemC-AMS," https://americodias.com/docs/systemc-ams/pll.md.

[18] COSEDA Technologies GmbH, "PLL example for coside® 2.5," https://www.coseda-tech.com/files/coside/user_files/Files/PLL_Example_for_COSIDE_2.5.zip.

[19] "Coside®," http://www.coseda-tech.com.

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

[21] Analog Devices, "Fundamentals of phase locked loops (PLLs)," *MT–086 Tutorial*, 2009.