



HAL
open science

Low-cost phase noise measurement with digital test resources

Stéphane David-Grignot

► **To cite this version:**

Stéphane David-Grignot. Low-cost phase noise measurement with digital test resources. Micro and nanotechnologies/Microelectronics. Université Montpellier, 2015. English. NNT : 2015MONTTS055 . tel-01590614

HAL Id: tel-01590614

<https://theses.hal.science/tel-01590614>

Submitted on 19 Sep 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de
Docteur

Délivrée par Université Montpellier

Préparée au sein de l'école doctorale Information, Structure
et Systèmes (I2S)

Et de l'unité de recherche Laboratoire d'Informatique, de Robotique et de
Microélectronique de Montpellier (LIRMM)

Spécialité : Systèmes Automatiques et Microélectronique (SYAM)

Présentée par Stéphane DAVID-GRIGNOT

**Mesure de bruit de phase faible coût à l'aide
de ressources de test numériques**

-

**Low-cost phase noise measurement with
digital test resources**

Soutenance prévue le 21 juillet 2015 devant le jury composé de :

Laurent Latorre, Professeur, Université de Montpellier	Directeur de thèse
Florence Azaïs, Chargé de Recherche, CNRS-LIRMM	Co-directrice de thèse
Salvador Mir, Directeur de Recherche, CNRS-TIMA	Rapporteur
Emeric de Foucauld, CEA	Rapporteur
Dominique Dallet, Professeur, IMS, Bordeaux INP	Examineur
François Lefevre, Ingénieur, NXP Semiconductors	Examineur
Pascal Nouet, Professeur, Université de Montpellier	Examineur



Acknowledgments

I would like to thank abundantly my PhD director Laurent Latorre as well as my employer from NXP Semiconductors, François Lefevre, for offering me the opportunity to work on this very entertaining and open subject and guiding my work so adequately. I also thank plentifully Florence Azais, my PhD co-supervisor, for her implication which led to numerous and rich contributions to this work. I would also acknowledge her extensive and impressive writing skills that permitted this study to be fruitfully translated into many publications.

All the team working on this thesis project is also very grateful to the CNFM (Coordination Nationale pour la Formation en Micro électronique et en nanotechnologies) for lending us their Automatic Test Equipment in Montpellier on which many of the most interesting experiments were carried out.

I would also like to thank Salvador Mir, Emeric de Foucauld and Dominique Dallet for their interest on my work and for accepting to be part of my dissertation committee.

I am also grateful to all the team and colleagues from the LIRMM in particular Regis Lorival and Beatrice Pradarelli for their inestimable help on the automated test equipment without which lots of the most interesting hardware results would have been far more difficult to obtain. I also thank Serge Bernard and Pascal Nouet for following the evolution of my thesis work.

I will also thank Alain Jean-Marie for his valuable formation and help on stochastic model, as well as Thierry Gil and Gilles Sassatelli for their advices on FPGA. I equally thank exterior professors Monique Jeanblanc and François Vernotte as well as pseudo-named Norbert from the Stackexchange website for their generous expertise on technical details.

Finally I am eternally grateful to my parents for providing me with a good education path, supporting my choices and generally encouraging me in my life.

Table of content

Acknowledgments	3
General introduction	7
Chapter 1. Context and state of the art	9
1.1 Industrial testing.....	9
1.1.1 Generalities	9
1.1.2 Case of analog/RF circuits	10
1.2 Phase noise testing.....	10
1.2.1 Phase noise definition	10
1.2.2 Prior work on phase noise testing.....	12
1.2.3 Current practice for industrial phase noise testing.....	17
1.3 Thesis objective and positioning	18
Chapter 2. A phase noise model	20
2.1 Phase noise as a Brownian motion.....	20
2.1.1 Phase fluctuation model.....	20
2.1.2 Standardizing factor	21
2.1.3 Hertz: a unit amongst others.....	21
2.1.4 Validation of model in Matlab®	21
2.2 Phase noise on hardware signals	23
2.2.1 Experimental setup.....	23
2.2.2 Validation on practical devices.....	28
2.3 Relationship between injected PN in time domain and measured PN in frequency domain	30
2.4 Conclusion	36
Chapter 3. First method: phase noise estimation based on instantaneous frequency estimation	37
3.1 Background on frequency estimation	37
3.1.1 Frequency estimation from 1-bit capture	37
3.1.2 Discretization quantum	38
3.1.3 Running average filtering	39
3.1.4 Error on threshold level.....	40
3.2 Statistical dispersion on instantaneous frequency estimations.....	40
3.2.1 Standard deviation	40
3.2.2 Mean Absolute Deviation (MAD)	42
3.2.3 Considerations for sampling ratio setting	43
3.2.4 Summary of proposed digital technique	45

3.3	Integrated module.....	46
3.3.1	Integrated module overview and principle	46
3.3.2	Implementation.....	48
3.4	Preliminary experimental results	52
3.4.1	Experimental setup.....	52
3.4.2	Results	53
3.5	Additional Filtering	54
3.5.1	Origin of Digital Technique Limitation.....	54
3.5.2	Jitter or voltage noise?	57
3.5.3	Improved Digital Technique	58
3.6	Experimental results.....	63
3.6.1	Filter validation.....	63
3.6.2	PN Hardware Measurements	65
3.7	Conclusion	65
Chapter 4.	Second method: phase noise estimation based on phase fluctuation reconstruction.	68
4.1	Theoretical background.....	68
4.2	Algorithm.....	69
4.2.1	Phase fluctuation reconstruction	69
4.2.2	Standard deviation of phase fluctuation increments.....	71
4.2.3	Allan deviation of phase fluctuation increments	75
4.2.4	Other approaches.....	78
4.3	Performance of the measurement process based on Allan deviation.....	80
4.3.1	Minimum phase noise level measurable.....	80
4.3.2	Measurement variability	84
4.4	Experimental results.....	88
4.4.1	On synthesized signals.....	88
4.4.2	On industrial devices	89
4.4.3	On industrial production line	90
4.5	Conclusion	92
Chapter 5.	On chip implementation.....	93
5.1	On chip algorithm	93
5.1.1	Processing only integers.....	93
5.1.2	Semi-pipelined implementation.....	94
5.1.3	Modular arithmetic	95

5.1.4	From digital signature to PN value	97
5.1.5	Area estimation	97
5.2	FPGA implementation	98
5.3	Experimental results.....	98
5.3.1	Measurements on synthesized signal	98
5.3.2	Measurement on NXP silicon tuner	101
5.4	Conclusion	103
General conclusion		104
Bibliographic references.....		106
Publications		112
Papers		112
Patent		112
List of Tables.....		113
List of Figures.....		113
Annex.....		118
1.	Stochastic model	118
1.1.	General Behavior	118
1.2.	High level noise.....	120
1.3.	Low level noise without filter	125
1.4.	Low phase noise level with filter	129
2.	Algorithms	134
2.1.	Signal Generation In simulation in Matlab.....	134
2.2.	Allan Variance Measurement In simulation in Matlab.....	135
2.3.	Simulation Example in Matlab.....	136
2.4.	Integrated module in Verilog	136

General introduction

With the development of communication and wireless applications, analog and radio frequency (RF) functions have become an essential part of electronic systems. With the efforts of the last decades, we today benefit from solutions that allow design and fabrication of these analog functions with interesting features concerning size, power and operating frequency. As to answer the mass market demand, the main challenge to which semiconductor founders are confronted is the production cost of these circuits. The cost not only includes the cost of fabrication of the circuit but the cost of testing as well. Indeed this step of the process is essential to guarantee the quality of devices delivered to the client. This context has motivated the study on which focuses this thesis: test cost reduction for analog/RF circuits.

This thesis is a collaboration between the LIRMM (Laboratory of Computer science, Robotics and electronics of Montpellier) and NXP Semiconductors, one of the world leading manufacturers of analog/RF integrated circuits. More precisely the study focuses on complex RF devices, such as silicon tuners, on which a lot of time-consuming RF tests should be performed to check the functionality and the specifications of the device. One of the most important tests is checking the phase noise of the analog/RF output signal. This specification parameter is linked to the product sensitivity, specifically at low reception levels. Measuring phase noise requires expensive digitizers to capture the analog signal and then perform signal analysis (FFT). Moreover the quantity of digitizers is limited per industrial tester, so multi-site efficiency is reduced.

The approach here is to use cheaper digital resources instead of expensive high-performance RF channels of the automatic test equipment (ATE). These digital resources can either be provided by the test equipment or can be integrated on-chip. The main idea is to perform a binary capture of the analog signal under test using the simple comparator available in a standard digital tester channel. Signal characteristics are then computed from this binary vector using specifically developed algorithms, in particular the phase noise level of the signal under test.

The first chapter describes the constraints associated with industrial testing in high volume and introduces several options to reduce the cost of phase noise measurement in the context of production testing with a state of the art of proposed solutions. The approach taken in this thesis is then justified.

The second chapter presents a mathematical model of phase noise. This model is implemented in simulation and is compared to hardware measurements. A hardware solution to synthesize a signal with a controlled phase noise level is also presented.

A first solution for phase noise evaluation based on 1-bit capture is presented in chapter three. This solution relies on instantaneous frequency estimation and analysis of the deviation of instantaneous frequency estimates. The technique is validated in simulation and an on-chip implementation is proposed. Hardware experiments are also presented, which demonstrate the validity of the proposed approach but also reveal a limitation in case of very low phase noise level due to the sensitivity of the technique to amplitude noise. An additional filtering step is then designed in order

to discriminate deviations induced by amplitude noise from deviations induced by phase noise. This permits to extend the valid measurement range, but imposes specific constraints on the sampling process.

A second algorithm for phase noise evaluation based on 1-bit capture is proposed in chapter four. It relies on instantaneous phase reconstruction rather than instantaneous frequency estimation. This solution is much more robust to amplitude noise and does not suffer from the specific constraints on the sampling process encountered in the previous chapter. Moreover this new technique not only permits to measure the phase noise level at a specific frequency offset, but it also gives access to the frequency-domain characteristics of phase noise. Simulation results as well as a stochastic model and hardware measurements are presented.

The fifth chapter presents an on-chip implementation of this last solution. This built-in self-test solution permits to lower even more the test cost because the digital channel of the automatic test equipment does not have to be as performant as when performing the acquisition. It also allows to measure phase noise during the life cycle of the chip, opening up new possibilities for chip designers.

Finally in the conclusion, the main contributions of this thesis are summarized and perspectives for future work are presented.

Chapter 1. Context and state of the art

1.1 Industrial testing

1.1.1 Generalities

Test of microelectronic circuits is an essential step in the fabrication process and marketing of new technology products. It aims in guaranteeing the good operation of the system. The test affects two main phases of the fabrication cycle of the integrated circuit: the characterization test of the device, in order to produce its datasheet, and the production test that is performed on every single product in order to guaranty its performance. The two objectives of these tests are quite different.

The characterization test is made on the first samples of the product, coming out of foundry. It validates the conception of the circuit for its operations and specifications. A whole battery of tests is applied to the circuit in different conditions to identify the limits of the product and compare them to the specifications. This characterization test can necessitate high performance equipment and the time of testing is not a primary constraint.

Once the characterization is done, the production of the circuit at high volume begins. Each circuit out of the production line has to be tested to guarantee it is correctly operational. Because of the very high number of fabricated devices, typically several millions a year, and the competitive environment, the time and cost of test is a primary constraint. Effective production test flow has to be cheap, fast and reliable to detect faulty circuits

Test is actually involved in different stages all along the production line of a system. The circuit is first tested at the wafer-level then it is tested once packaged. Once the circuit shipped to the client the system is tested with the circuits mounted on a PCB board. It is important to limit the impact of losses on the global cost. Table 1.1 is an illustration of the replacement cost at each stage of the system integration [1].

Detection level	Replacement cost
On wafer	0.1
Packaged circuit	1
PCB board	10
System in production line	100
Shipped system	1000

Table 1.1 Replacement cost with detection level in the integration process

In the end, the cost of testing represents a large part, up to 50%, of the production cost. It is important to develop solutions to limit the testing cost. The solutions for cost reduction are generally focused over 3 targets:

- Test time reduction
- Cheap instrumentation
- Parallelization of testing

Over years, efficient solutions have been developed for digital circuits thanks to structural test approaches, design for test (DFT) and built in self-testability (BIST). This is not the case for analog and RF circuits.

1.1.2 Case of analog/RF circuits

Analog/RF circuits are usually tested with a specification-oriented approach, which consists in measuring the circuit performances and comparing the measured values with specification tolerance limits. This approach offers good test quality but suffers from very high test cost. Several factors contribute to this cost.

First testing time is a critical issue. Indeed, there are usually a high number of specifications to verify which often require different specific test configurations. Measurements are therefore performed sequentially, resulting in long test times. As an example in [2], the test time for analog/RF blocks represents 57% of the total test time, while the test time for memory and digital blocks only represent 2% and 11% respectively.

Second the measurement of analog/RF performances necessitates the use of an Automatic Test Equipment (ATE) equipped with specific dedicated resources. Ranked from the most expensive to the cheapest ones, we find:

- RF resources
- Mixed-signal resources
- Analog resources
- DC measurements
- Digital resources

RF/MS options constitute around 55% of tester cost. A digital channel costs 50 times less than a RF channel. It is clear that the use of specific dedicated resources is a major contributor of the overall test cost. As an example in [3], the test of RF functions represents more than 40% of the testing cost of the SOC, mainly because of the required dedicated resources.

Finally it should be highlighted that due to their high cost, analog/RF resources are usually available in small quantity, limiting the possibility to reduce the testing cost through multi-site testing.

In this context, there is therefore a strong demand in developing test solutions for analog and RF circuits applicable with low-cost test equipment and compatible with multi-site test implementation.

1.2 Phase noise testing

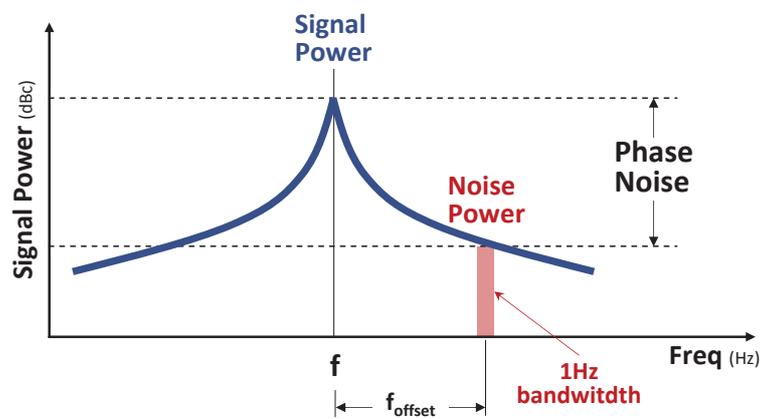
1.2.1 Phase noise definition

Phase noise is a key element in many RF and radio-communication systems as it can significantly affect the performance of systems. For radio receivers, phase noise on the local oscillators within the system can affect specifications such as reciprocal mixing and the noise floor. For transmitters, it can affect the wideband noise levels that are transmitted. Additionally it can affect the bit error rate on systems using phase modulation. Phase noise is also important for many other systems including RF

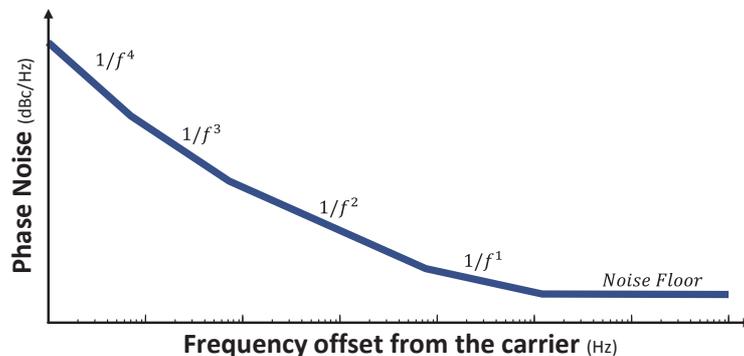
signal generators, where very clean signals are required to enable the generator to be used as a reference source.

Phase noise refers to random short term fluctuations that affect the nominal frequency of a periodic signal. Note that phase noise and timing jitter are both ways of describing the same parameter. Phase noise describes the performance in the frequency domain, whereas jitter describes the performance in the time domain.

In the frequency domain, phase fluctuations manifest themselves as sidebands that appear on either side of the fundamental tone in the signal power spectrum, as illustrated in figure 1.1a. The phase noise may be specified in a number of ways, but the most common one is to look at the noise level at a given point. Phase noise is typically measured in dBc/Hz and corresponds to the noise power contained in a 1 Hz bandwidth located at a given offset from the carrier, relative to the carrier power.



a) Signal power spectrum



b) Phase noise spectrum

Figure 1.1 Frequency-domain representations of phase noise

For a complete phase noise specification, several points will be measured to give an indication of the frequency domain characteristics. This leads to the phase noise spectrum illustrated in figure 1.1.b, in which the phase noise level is plotted as a function of the frequency offset from the carrier. Different phase noise sources can be identified on this spectrum, looking at the asymptotic behavior.

Indeed there are various origins for phase noise, and therefore various phase noises with corresponding power laws. Each phase noise has its response in the frequency domain. Each one can be identified by the slope S_φ of its phase spectrum or by the slope S_y of its frequency spectrum.

In the total noise power, the frequency spectrum corresponds to the sum of the different noises contributions, weighted with their power coefficient:

$$S_y = \sum_{\alpha=-2}^{+2} h_\alpha f^\alpha \quad (1)$$

$S_y(f)$	$S_\varphi(f)$	Noise type	Origin
$h_{-2}f^{-2}$	$b_{-4}f^{-4}$	Random walk FM	Environment
$h_{-1}f^{-1}$	$b_{-3}f^{-3}$	Flicker FM	Resonator
h_0	$b_{-2}f^{-2}$	White FM	Thermal noise
h_1f	$b_{-1}f^{-1}$	Flicker Phase modulation	Electronic noise
h_2f^2	b_0	White PM	External white noise

Table 1.2 Phase noise power law response

Table 1.2 summarizes the origins of the main noises sources classically encountered in electronic systems [4].

1.2.2 Prior work on phase noise testing

Traditionally, phase noise measurement relies on analog methods in order to evaluate the frequency-domain characteristics of phase noise, using spectrum analyzers or dedicated analog measurement systems [5-7]. Several methods are available for measuring phase noise, including the direct spectrum method, phase lock loop (PLL) method, delay-line discriminator method, and cross-correlation method. These methods differ from one to another in measurement principle, performance and measurement frequency range. Note that the usual goal for measuring phase noise in an R&D environment is to achieve the lowest measurement noise floor possible. However in a production environment, the objective is fast throughput for product phase noise performance testing. Traditional methods are therefore not necessarily appropriate for this specific context.

With the advances in technology and the availability of high-performance data acquisition systems, digital signal processing approaches have been proposed in order to derive the frequency domain characteristics of phase noise from digitized time-domain waveforms.

In [8-11], digital signal processing methods are developed based on oversampled capture of the input signal and optimized quadrature demodulation scheme. As illustrated in figure 1.2, the signal is directly digitized with a fast analog to digital converter. A pre-processing consisting in low-pass filtering and decimation is applied to have a better resolution of the signal. A quadrature demodulation scheme is then used to recover the phase. The phase can then be used to compute phase noise with the help of a Fourier transform or Allan deviation. This method has been validated for signals with frequencies around 1MHz.

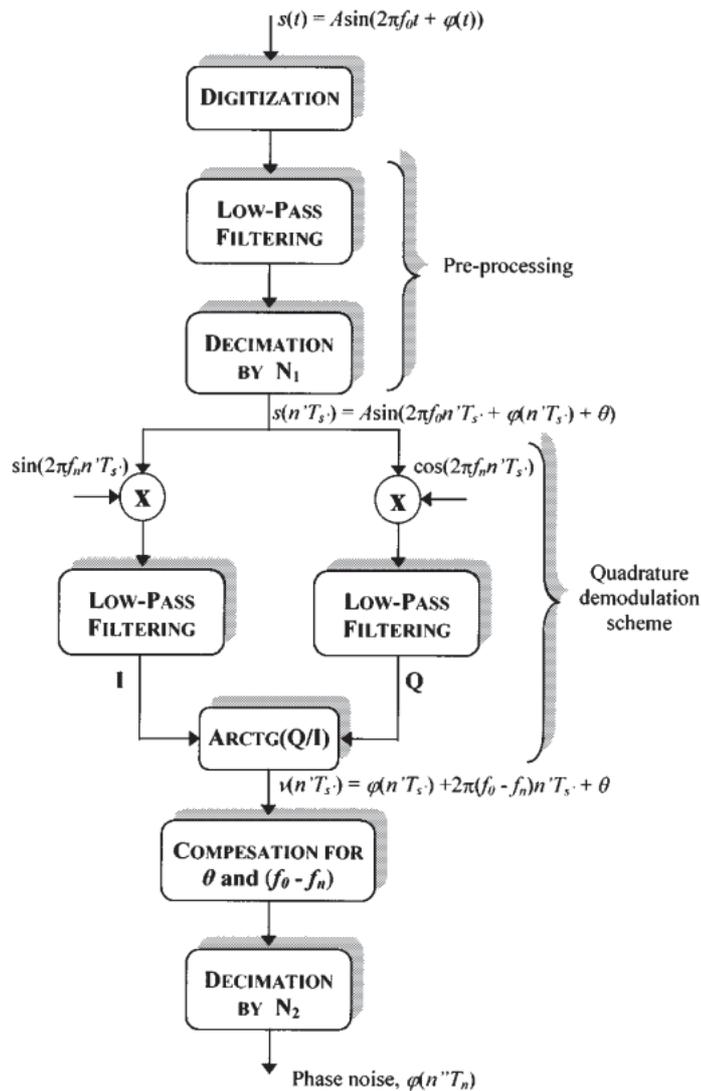
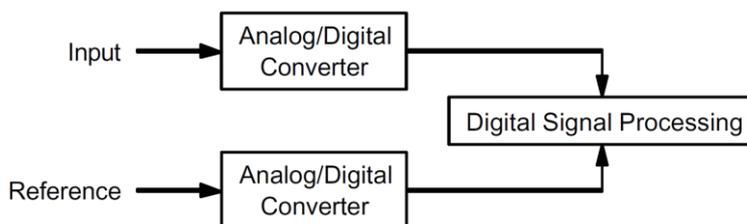
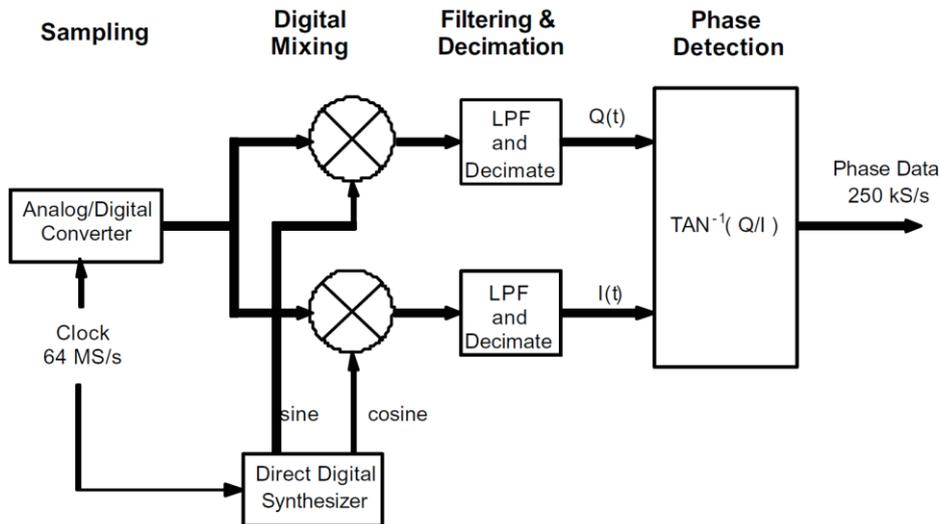


Figure 1.2 Block diagram of the method proposed in [8]

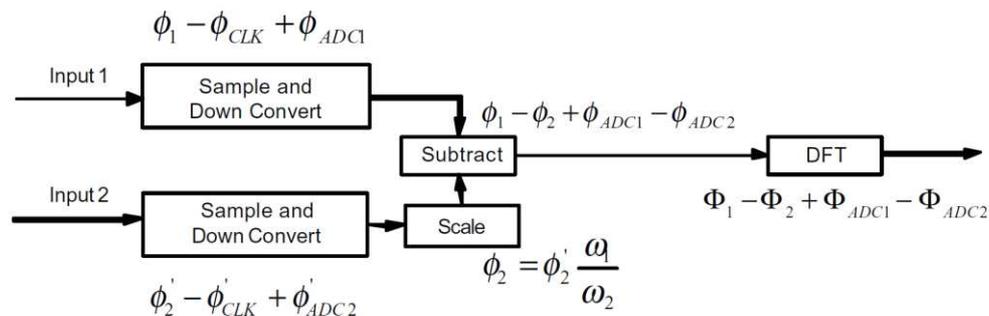
Similarly in [12], the input RF signal and a reference signal are digitized by fast analog-to-digital converters in order to perform direct-digital phase noise measurements. Down-conversion and phase detection functions are then implemented by digital signal processing. The principle of the method is illustrated in figure 1.3.



- a) The RF signals are immediately converted to digital samples in order to perform direct - digital phase-noise measurements



- b) A local oscillator synthesized from the internal clock down-converts the input to base-band where the samples are used to compute the phase difference between the LO and the input



- c) After down-conversion, the phase differences are scaled, subtracted, and Fourier analyzed to determine the frequency content

Figure 1.3 Direct-digital phase noise measurement method proposed in [12]

Current phase noise analysis techniques also include all-digital methods for acquiring time-domain data such as waveform crossing points. In particular, Time Interval Analyzers (TIAs) are used in [13,14] to record the zero-crossing times of the analyzed signal. Figure 1.4 show a simplified block diagram of the system including the zero-crossing measurement machine, frequency ω_0 estimator, phase calculation and numerical signal processing. The zero-crossing times are processed to compute phase deviation, with the reference frequency specified as a numerical value or derived from the times themselves. Phase digitizing can be applied even in the presence of modulation, as the underlying clock can be reconstructed in software to fit the data. Measurements derived from this phase data such as phase noise, jitter analysis, Allan variance (AVAR), maximum time interval error (MTIE), and time deviation (TDEV) can be applied.

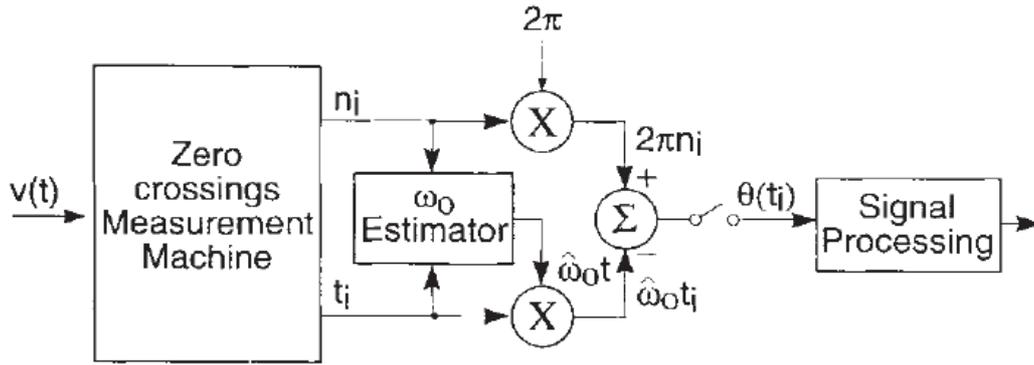


Figure 1.4 Block diagram of the proposed method in [13]

The same approach is adopted in [15] in order to characterize phase noise in a spin torque oscillator from time-domain measurements.

Another all-digital method is presented in [16], which relies on the use of a time-to-digital converter (TDC) to obtain digitized waveform zero-crossing. This technique is directed towards the detection of sinusoidal phase noise components located at particular frequencies. As illustrated in figure 1.5, the classical analog sampling is replaced by a time-to-digital converter, and an infinite-impulse response (IIR) filter is used as a replacement for conventional FFT. This eliminates the need for storing the entire waveform before calculations can begin, and the need for making calculations to evaluate frequencies which are not relevant.

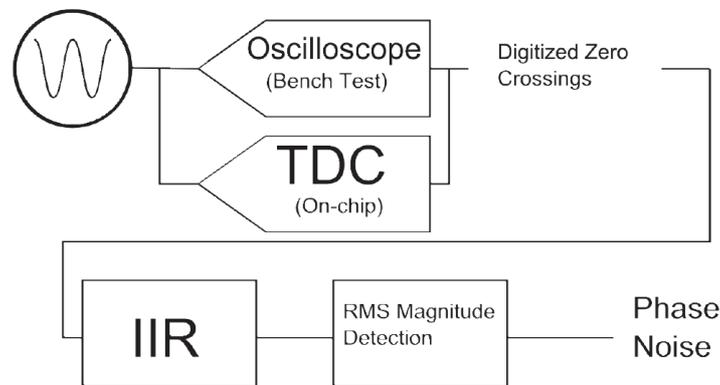


Figure 1.5 IIR extraction technique proposed in [16] for the detection of sinusoidal phase noise at a particular frequency

Finally in order to reduce the testing costs, another approach is to develop Built-In Self-Test (BIST) solutions, where the required test resources are integrated within the circuit itself. While many works can be found in the literature regarding BIST solutions for jitter measurement [17-35], only few deal with BIST solutions for phase noise measurement.

In [36], an on-chip spectrum analyzer using switched-capacitor techniques is proposed. The main characteristic of the system is the inherent synchronization. Both, frequency of the sinewave generator and filter center frequency follow the main clock frequency, making the system self-synchronized and allowing a simple way to perform a frequency characterization. The resolution of this on-chip spectrum analyzer is limited to 8 bits.

In [37], authors propose an analog BIST circuitry for Go/No-Go testing of synthesizer phase noise. The paper exploits the fact that there is a relation between the output noise of the PLL and the amplitude noise on the control voltage of the VCO. They therefore design a BIST circuit that measures the band-limited, low frequency noise power at the input of the voltage controlled oscillator (VCO). The block diagram of the noise measurement system is illustrated in figure 1.6, which comprises a linear preamplifier, a squarer that takes the time-domain square of the noise signal, and an integrator that accumulates the squared noise signal, effectively calculating its energy over a given time window.

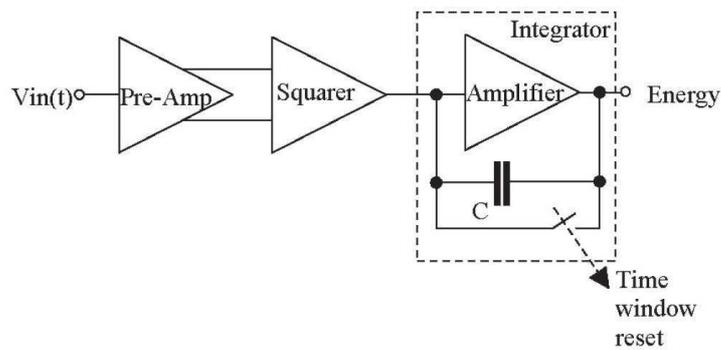


Figure 1.6 Block diagram of the noise power measuring system proposed in [37] for go/no-go testing of synthesizer phase noise

In [38], an on-chip phase noise measurement circuit is proposed, based on a delay-line discriminator architecture. As illustrates in figure 1.7, the proposed circuit uses a low-noise voltage-controlled delay-line (VCDL) and mixer-based frequency discriminator to extract the phase-noise fluctuations at baseband. A self-calibration circuit is used to operate the measurement circuit at its highest sensitivity point. This circuit is intended to measure phase noise of clock signals.

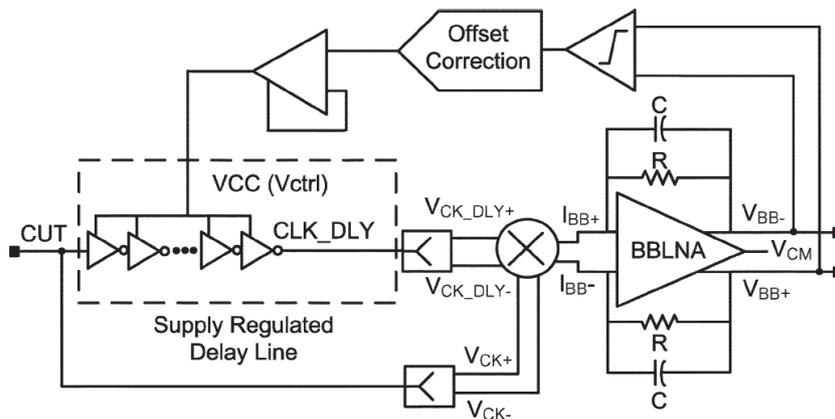


Figure 1.7 Circuit architecture of the phase-noise measurement module proposed in [38]

Finally in [39], authors use an all-digital on-chip $\Sigma\Delta$ -frequency discriminator to compute the instantaneous frequency which they can compare to a reference frequency. The result of the comparison produces a noise shaped digital bit stream whose average value is proportional to the frequency error between the DUT signal and the reference signal. Thanks to phase domain model of the $\Sigma\Delta$ -frequency discriminator, this technique can trace the phase noise asymptotic behavior. This technique is intended to be integrated as part of a BIST scheme for PLL-based clock synthesizers.

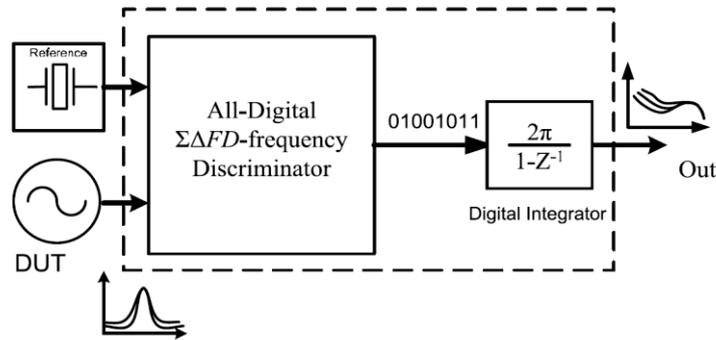
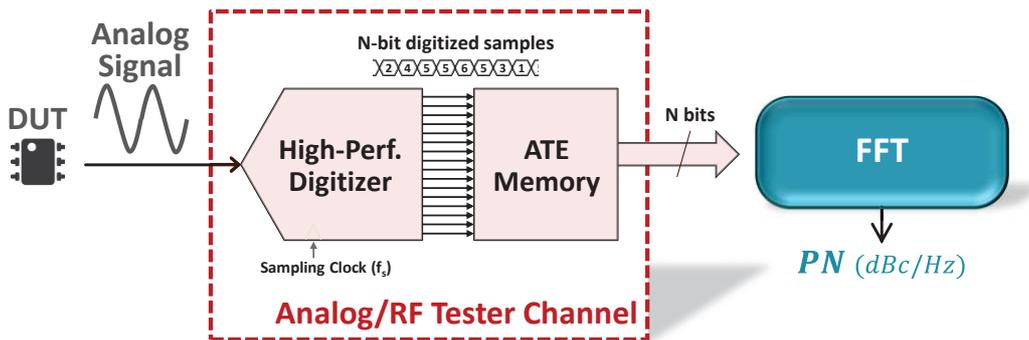


Figure 1.8 Phase noise characterization scheme proposed in [39]

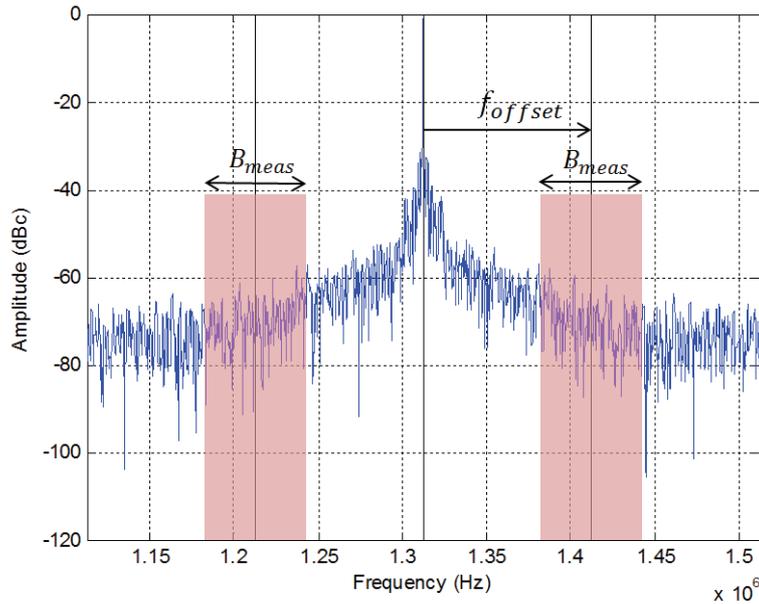
1.2.3 Current practice for industrial phase noise testing

In a standard production test, phase noise is extracted from a spectral analysis of the signal. As illustrated in figure 1.9, the signal is first digitized by means of ATE analog/RF capture resources (a N-bit high-performance Analog-to-Digital-Converter) and stored in the ATE memory. FFT is then performed on the recorded digital data to compute the resulting signal power spectrum. As introduced in section 1.2.1, the widening of the spectral content around the fundamental tone can be associated with phase noise.

Practically, the phase noise value expressed in dBc/Hz is computed by subtracting the signal level to the noise level, where the noise level is measured by integrating noise over a specific bandwidth B_{meas} on both sides of the signal tone and applying a correcting factor for 1Hz bandwidth normalization. Note that spectral averaging is often implemented using data captured in several time window frames in order to reduce measurement variability. A Blackman window has also to be applied to the signal if the capture is not synchronous.



a) Industrial setup



b) Signal power spectrum with phase noise

Figure 1.9 Current industrial practice for industrial phase noise testing

1.3 Thesis objective and positioning

The main objective of this thesis is to reduce the phase noise testing cost for complex RF devices. Indeed in the market of analog and RF integrated circuits, complex RF devices are gaining more and more importance in modern communication and multimedia applications. An example of such devices is silicon tuners, which have found use in set-top boxes for satellite and cable for nearly more than a decade. Now, thanks to their high performance, broad frequency coverage, compactness, and universality, they are making inroads into phones, service gateways, automobiles, personal computers, and even televisions. In this very competitive market, cost is a crucial factor, and in particular the testing costs which become the dominant part of the total manufacturing cost for this type of devices.

Phase noise is an essential characteristic of RF complex devices. In such devices, phase noise is usually measured on the analog output signal delivered at Intermediate Frequency (IF) and corresponds to the accumulated phase noise contribution of the various elements present along the signal path. The conventional practice for phase noise production testing relies on an analysis in the frequency domain, by taking time-domain data on the IF analog output and computing the Fast Fourier Transform (FFT). This approach requires the use of industrial testers equipped with expensive analog/RF resources, in particular high-performance digitizers to perform the signal capture, as well as computing power for the FFT. Moreover because such expensive resources are available in small quantity, the possibility to reduce the testing cost through multi-site testing is limited.

In order to reduce the testing cost, the strategy we have chosen to explore in this thesis relies on the use of digital test resources to perform phase noise measurement. Note that this work is the continuation of a previous thesis defended at LIRMM in 2011 by Nicolas Pous [40], which laid the foundations of the analysis of analog/RF signals using solely the test resources available in a standard digital tester channel. In particular, the analysis of FM-, AM- and QAM-modulated signals has been

investigated [41-43], as well as SNR estimation of analog sine-wave signals [44]. The same strategy of using a digital ATE for the analysis of analog signals has also been investigated in [45], in case of triangular waveform signals.

The general idea of this strategy is to perform oversampled 1-bit signal acquisition of the analog/RF signal with a standard digital channel, and to analyze the resulting bit stream with a dedicated digital processing algorithm. Indeed as illustrated in figure 1.10, the comparator included in the digital tester channel realizes a level-crossing operation which converts the amplitude and/or frequency information contained in the analog signal into a timing information in the resulting bit stream. In other words, the sequence of 0 and 1 stored in the ATE memory is somehow representative of some characteristics of the analog signal. It is then the role of the post-processing algorithm to retrieve this information and extract the desired signal characteristics.

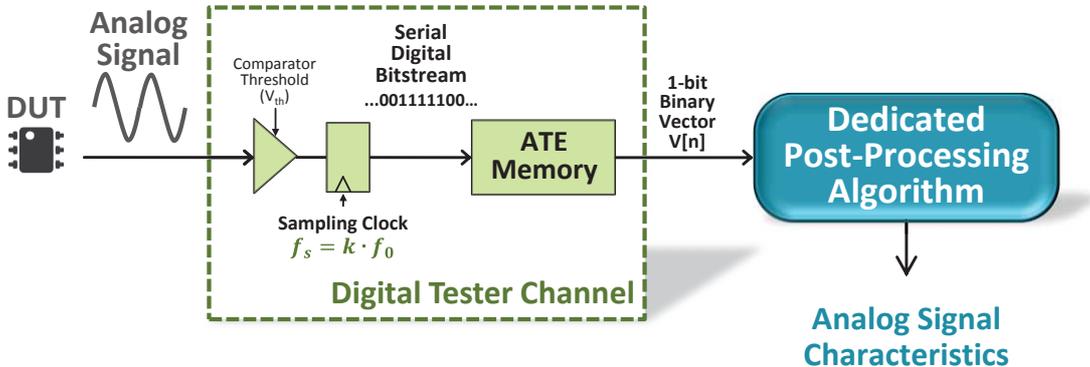


Figure 1.10 General idea of analog/RF signal analysis using digital test resources

For a general point of view, this strategy relies on the assumption that it should be possible to retrieve the characteristics of an analog signal using the data resulting from level-crossing operation. The literature actually provides a number of elements that support this assumption. Indeed the concept of level-crossing has been used for many years for signal reconstruction in various fields such as image processing, speech recognition... [46-48] In the microelectronics domain, this concept is at the origin of digital FM demodulator architectures [49-61] proposed from beginning of 90's, and more recently of asynchronous analog-to-digital converter architectures [62-67]. The principle of this new class of converters is to perform sampling in the voltage domain followed by voltage-to-time conversion. The idea is actually to exploit the current trend of technology scaling, which permits to design circuits operating at lower supply voltages but with ever higher frequencies. These circuits therefore undergo a degraded voltage resolution whereas they benefit from an improved time resolution [68]. The same trend is exploited in [13-16] for phase noise testing based on timing measurements. However, proposed methods rely on the use of precise timing measurement instrumentation such as TIAs or TDCs. Our objective in this thesis is to develop a solution that does not require this precise timing measurement instrumentation, but rely only on the standard resources available on a digital ATE channel, so that we can have a very low-cost production test solution.

Chapter 2. A phase noise model

Because new ideas are best first explored using mathematical and computer simulations, the first step in the study of phase noise measurement was to setup a correct simulation environment. In particular, we need to define a phase noise model that permits to perform realistic phase noise injection in the time-domain in an analog signal.

In this thesis, we target phase noise evaluation for complex RF devices such as silicon tuners for instance. For those kind of products, the total phase noise evaluated on the IF analog output involves the contribution of the various elements present along the signal path. The total phase noise measured on the IF output typically corresponds to white FM phase noise, with $1/f^2$ characteristics. Our objective is therefore to define a model that permits a time-domain generation of such $1/f^2$ noise.

Further away in the development of the thesis we will specify how other phase noise types can be measured, but for this simulation model sole white FM is considered first. Later on, other noises such as amplitude noise and jitter, which corresponds to white PM in table 1.2, will also be considered.

2.1 Phase noise as a Brownian motion

2.1.1 Phase fluctuation model

Let us consider an analog sinusoidal signal $s(t) = A \cdot \sin(\Omega(t))$ affected with phase noise and sampled at frequency rate f_s . The resulting signal $s(nT_s)$ is expressed by:

$$s(nT_s) = A \cdot \sin(2\pi f n T_s + \varphi_0 + \varphi(nT_s)) \quad (2)$$

where A and f are nominal values of the amplitude and signal frequency respectively, T_s ($T_s = 1/f_s$) is the sampling period, φ_0 is the initial signal phase, and $\varphi(nT_s)$ is the time-varying phase instability corresponding to phase noise.

In order to control phase noise, we need to define the phase fluctuation $\varphi(nT_s)$. A simple way to generate phase noise with $1/f^2$ characteristics is to consider a one-dimension Brownian motion [69]. The phase noise model considered here is therefore represented by:

$$\varphi(nT_s) = 2\pi T_s \cdot \sum_{j=0}^n b \cdot r_j \quad (3)$$

where r_j are random variables associated with a centered normal distribution $\mathcal{N}(0, \sigma_f^2)$ and b is a standardizing factor explained in the next section. This model allows to control the phase noise level injected in the time-domain signal by adjusting the value of σ_f , which is a standard deviation, in Hertz unit.

Figure 2.1 gives an example of a time-domain phase fluctuation generated using this model.

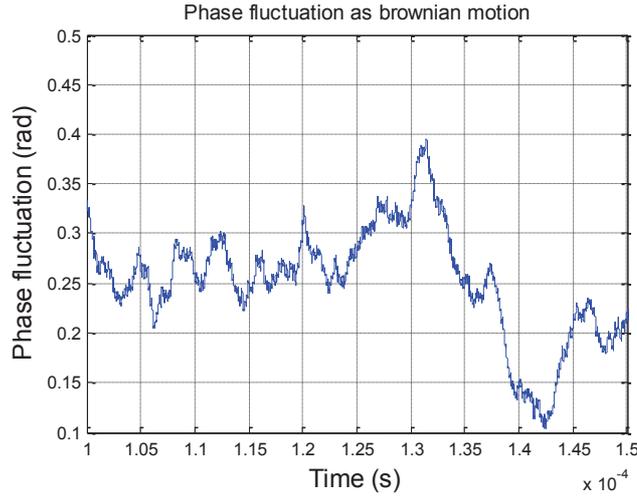


Figure 2.1 Example of time-domain phase fluctuation generated using a one-dimension Brownian motion

2.1.2 Standardizing factor

$b = \sqrt{f_s/f} = \sqrt{n}$ is a standardizing factor that ensures independence of the noise level with respect to the sampling frequency. Indeed, we want to have the same deviation over 1 signal period, whatever the sampling frequency. There are $n = \frac{f_s}{f}$ samples in one period. Since the variation of a Brownian motion is:

$$Var\left(2\pi T_s \cdot \sum_{j=0}^n b \cdot r_j\right) = \frac{4\pi^2}{f_s} b^2 n \sigma_f^2 = \frac{4\pi^2}{f^2} \sigma_f^2 \quad (4)$$

So the standard deviation is:

$$std\left(2\pi T_s \cdot \sum_{j=0}^n b \cdot r_j\right) = \frac{2\pi}{f} \sigma_f \quad (5)$$

2.1.3 Hertz: a unit amongst others

The input of the model is therefore σ_f in hertz. Note that it could be any of the four following variables since it is equivalent to consider any of the standard deviations σ , σ_f , σ_T or $\sigma_{2\pi}$.

$$\sigma = \frac{\sigma_f}{f} = \frac{\sigma_{2\pi}}{2\pi} = \frac{\sigma_T}{T} \quad (6)$$

The choice to use σ_f with hertz as unit is consequently arbitrary.

Taking the previous equation:

$$std\left(2\pi T_s \cdot \sum_{j=0}^n b \cdot r_j\right) = \frac{2\pi}{f} \sigma_f = \sigma_{2\pi} \quad (7)$$

We thus always have the same standard deviation over 1 unit of time: the signal period.

2.1.4 Validation of model in Matlab®

To validate this model, Matlab® simulations were performed. In figure 2.2, the spectrum of a simulated noisy 1 MHz signal is calculated for various sampling frequencies. The obtained spectrums

perfectly overlap as expected. The simulated signal thus has the same behavior whatever the sampling frequency thanks to the b coefficient. On this simulation $N_{periods} = 1000$ and $\sigma_f = 2000\text{Hz}$.

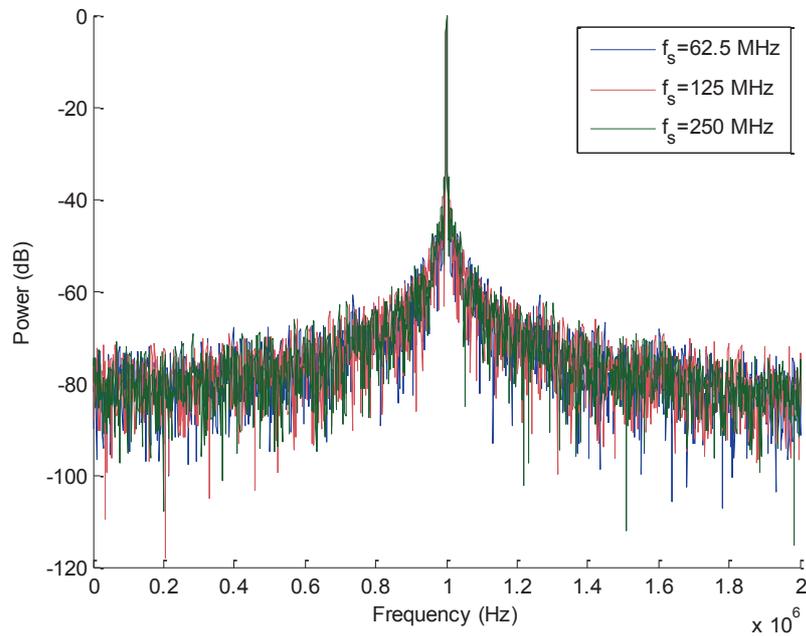


Figure 2.2 Spectrum comparison between simulated signal with injected phase noise for different sampling frequencies

Figure 2.3 reports the phase noise value measured on the spectrum at a frequency offset of 10kHz, for different values of the sampling frequency between 10MHz and 1GHz. Here we compute the spectrum using 10 000 periods to reduce the dispersion; both f and σ_f were kept constant, as previously. As desired, the phase noise value measured on the spectrum does not depend on the sampling frequency.

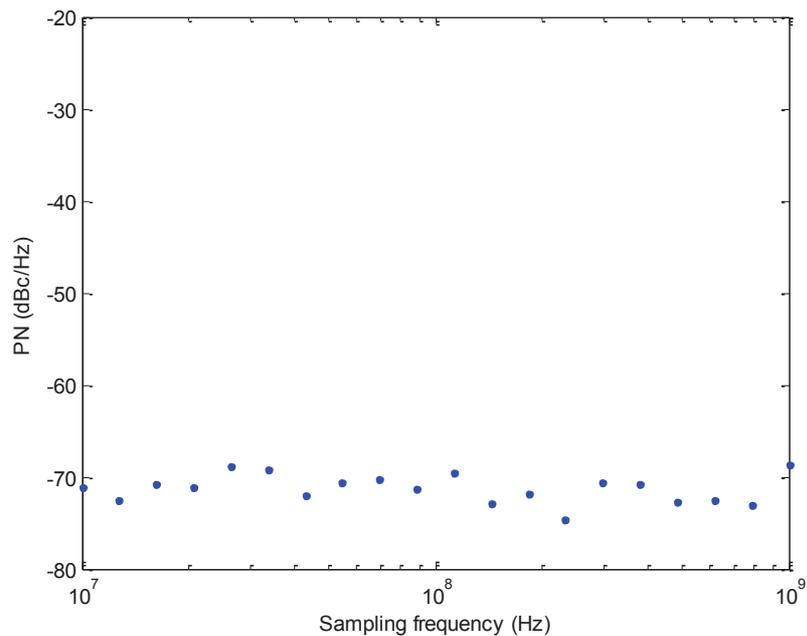


Figure 2.3 Phase noise measurement on simulation model for different sampling frequencies

Figure 2.4 demonstrates the ability to control the phase noise level present in the simulated signal by adjusting the value of σ_f . The exact relationship between the two phase noise representations will be addressed later on.

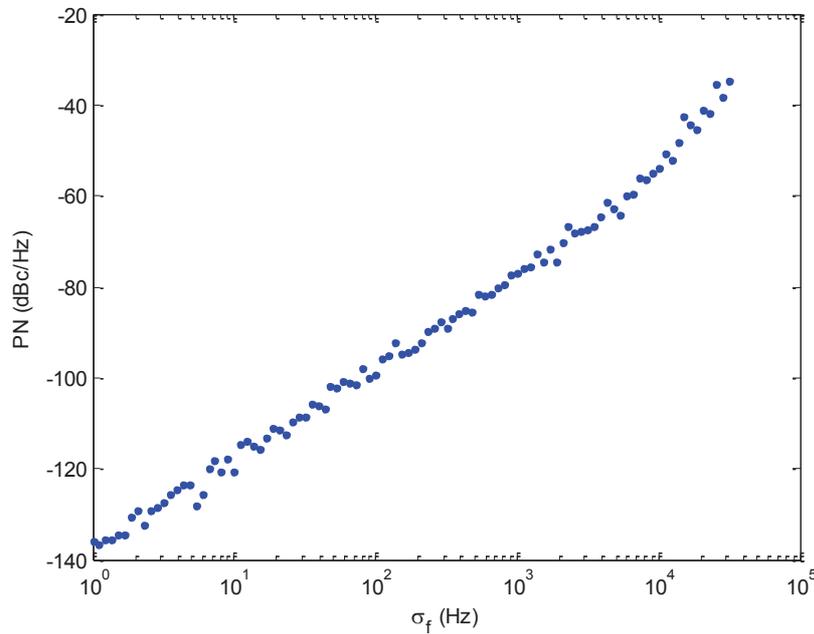


Figure 2.4 Phase noise measurement for different values of injected phase noise level σ_f

2.2 Phase noise on hardware signals

2.2.1 Experimental setup

In order to perform measurement on physical signals, we also setup an experimental environment. Test signals are obtained through frequency modulation of a “pure” RF sinusoidal signal with a white noise signal. More precisely, an Agilent 33120A Arbitrary Waveform Generator (AWG) is used to generate a white noise signal. The output of this generator is connected to the FM input of an Agilent E4425B RF signal generator. The output of this RF generator is therefore a sinusoidal signal affected by phase fluctuations, which corresponds to the Signal Under Test (SUT). The phase noise level present in the synthesized signal can be adjusted by means of the modulation depth f_Δ on the RF generator.

In order to evaluate the phase noise level, acquisition of the synthesized signal capture is realized using a Yokogawa DLM2054 Mixed-Signal Oscilloscope and the signal spectrum is obtained through FFT computation. Classical phase noise measurement is then performed on this spectrum. The complete setup is illustrated in figure 2.5.

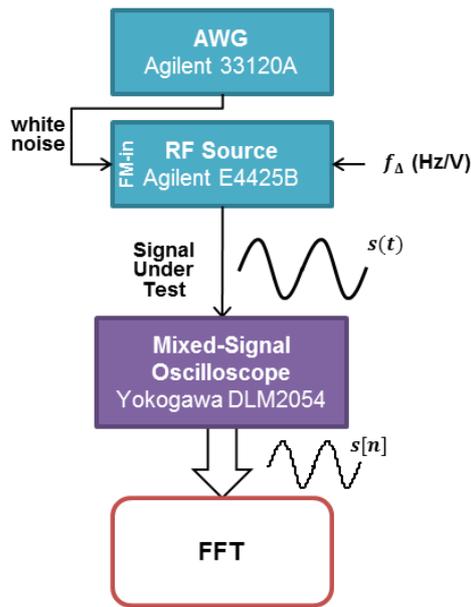


Figure 2.5 Experimental setup for phase noise measurement on a synthesized signal affected with phase noise

2.2.1.1 White noise analysis

The generation of white noise is accomplished by the arbitrary waveform generator. We arbitrarily set $V_{pp} = 2V$, which doesn't mean much when generating Gaussian voltage noise. Figure 2.6 shows a short sequence of the output of the AWG captured at 62.5MHz .

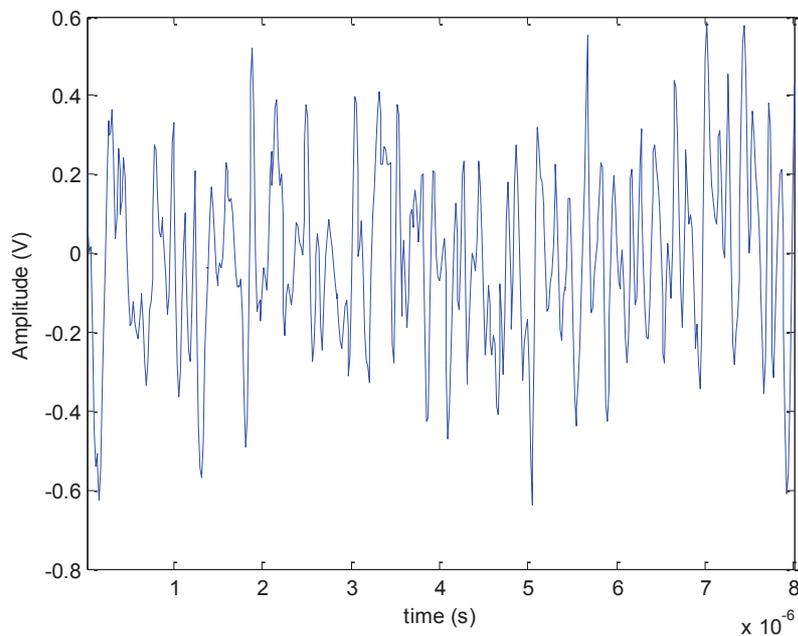


Figure 2.6 White noise from arbitrary waveform generator

We measured the generated signal to determine the white noise level. Figure 2.7 provides a histogram of the captured samples. As expected, the sample distribution is a Gaussian distribution with zero mean; its standard deviation is:

$$\sigma(V_{AWG}) = 0,223V \tag{8}$$

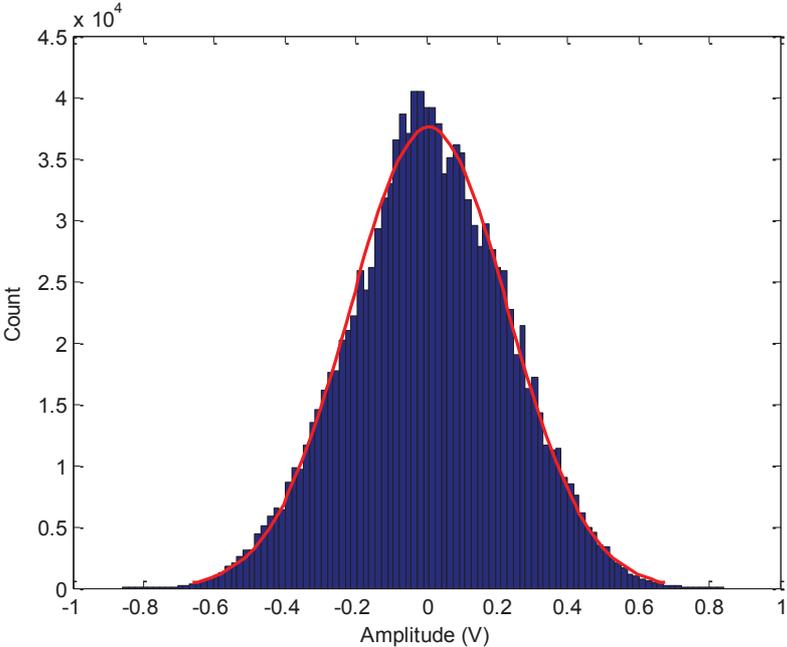


Figure 2.7 Histogram of white noise signal captured on AWG output

Figure 2.8 represents the Fourier transform of the so-obtained white noise. All the frequencies up to approximately 10MHz have the same power. Having a constant power over all the frequency range is expected to suit the model. This spectrum is equivalent to S_y in the table 1.2.

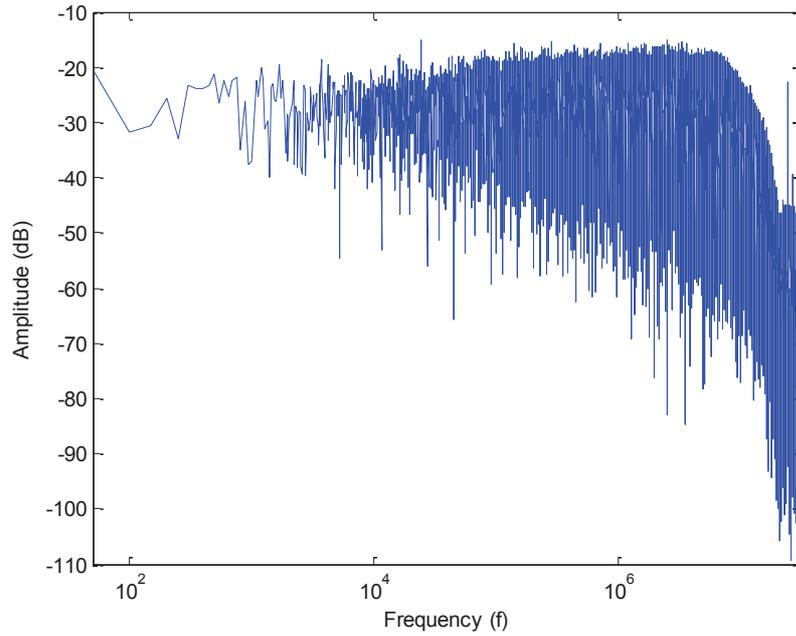


Figure 2.8 Spectrum of white noise signal captured on AWG output

2.2.1.2 Synthesized signal analysis

The synthesized signal is to be an analog/RF sinusoidal signal affected with phase noise. For this, we use the RF source to generate a sinusoidal signal, which is modulated in frequency by the white noise signal generated by the AWG. The resulting signal is therefore given by:

$$y(t) = A \cdot \cos\left(2\pi \int_0^t f(\tau) d\tau\right) \quad (9)$$

With $f(\tau) = f_c + f_\Delta x_m(t)$ where f_c is the frequency of the sinusoidal signal generated by the RF source, f_Δ is the modulation depth and $x_m(t)$ is the white noise signal generated by the AWG, we obtain:

$$y(t) = A \cdot \cos\left(2\pi f_c t + 2\pi f_\Delta \int_0^t x_m(\tau) d\tau\right) \quad (10)$$

So the phase fluctuation $\varphi(t)$ is:

$$\varphi(t) = 2\pi f_\Delta \int_0^t x_m(\tau) d\tau \quad (11)$$

$\varphi(t)$ corresponds to a Brownian motion if x_m is Gaussian.

As an illustration, figure 2.9 shows the spectrum a synthesized signal generated with $f_c = 1 \text{ MHz}$ and $f_\Delta = 11200 \text{ Hz}$. The spectrum of a simulated 1 MHz signal with $\sigma_f = 200 \text{ Hz}$ is also reported for comparison (the noise level of the simulated signal has been manually tuned to fit the spectrum of the captured synthesized signal). Both spectrums exhibit the same shape, which demonstrates that both methods generate a signal affected by similar phase noise. Still, one can notice that the physical signal exhibits slightly higher noise level at both ends of the plot (i.e. far from the fundamental tone). This is due to amplitude noise, which exists in the real world but which is not represented in the model so far.

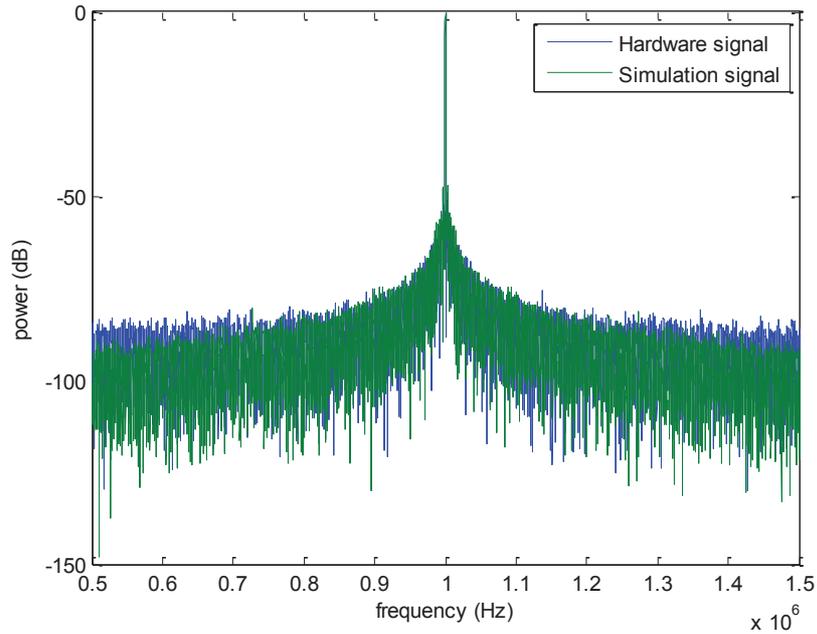


Figure 2.9 Spectrum of simulated and synthesized 1MHz signals affected by phase noise

Figure 2.10 reports the phase noise level measured on the spectrum of the synthesized signal at a frequency offset $f_{Offset} = 25kHz$ according to the value of the modulation depth f_{Δ} programmed on the RF signal generator. This figure clearly demonstrates that the phase noise level present in the synthesized signal can be adjusted on a large range by changing the modulation depth. However in case of very low phase noise level, i.e. low values of the modulation depth f_{Δ} , some other noise sources (amplitude, jitter) take over and we observe a saturation of the phase noise level measured on the spectrum.

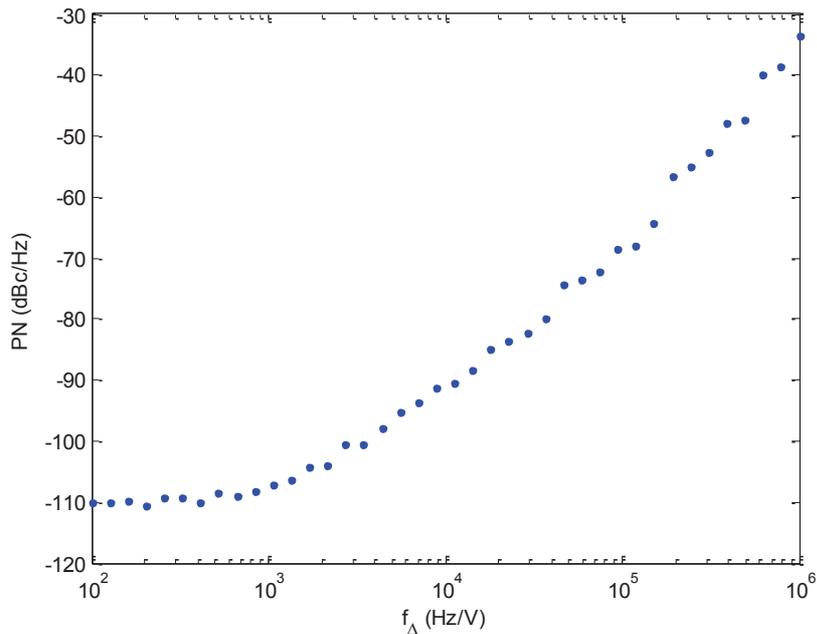


Figure 2.10 Phase noise measurement of a 1MHz synthesized signal for different values of the modulation depth

2.2.2 Validation on practical devices

It is important to verify that both simulated and synthesized signals exhibit the same characteristics (regarding phase noise) as signals obtained from real industrial devices to be tested. NXP provided us with one golden device and 10 faulty devices of their TDA18260 silicon tuners. It goes without saying that the faulty devices solely failed for phase noise reason.

NXP usually do a phase noise test on the IF output of the device. This is what we did here. In order to setup test conditions, the device has to be provided with a 1004MHz input signal and have particular configuration of its gain control chain. Figure 2.11 illustrates the experimental setup.

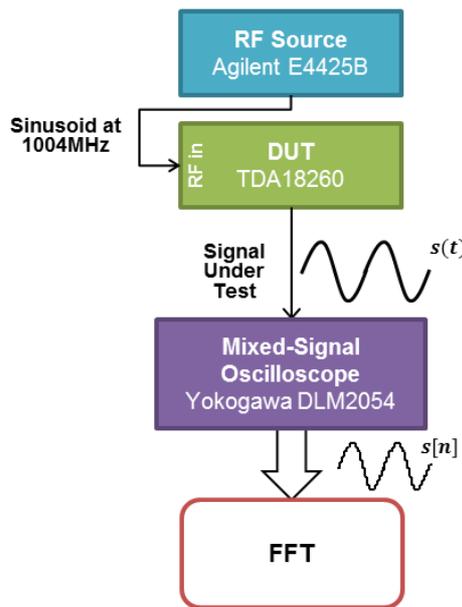


Figure 2.11 Experimental setup for phase noise measurement on industrial chip

The spectrum of the captured signal is compared to the spectrum of a simulated signal in figure 2.12. Both spectrums exhibit a similar shape, demonstrating the validity of the proposed model to perform realistic time-domain phase noise injection for complex RF devices.

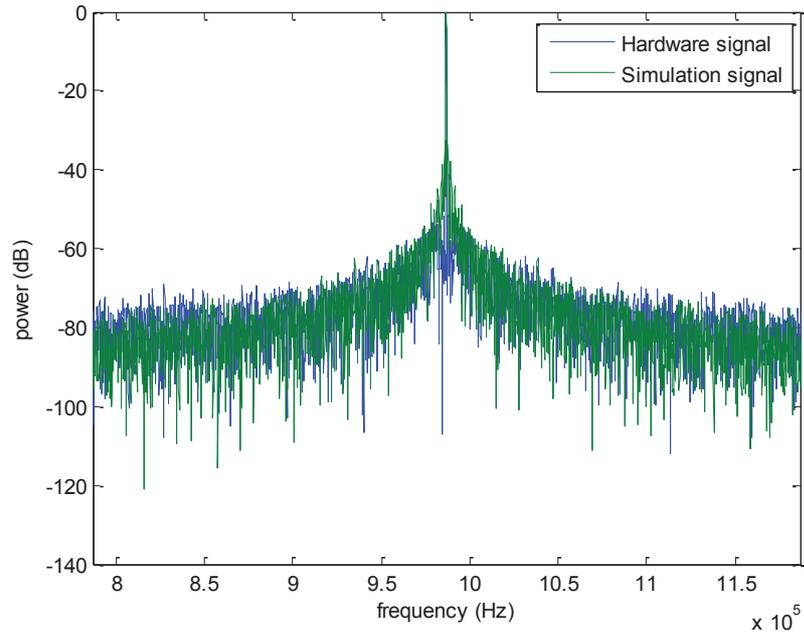


Figure 2.12 Comparison of spectrum of simulated signal with signal from industrial chip

2.3 Relationship between injected PN in time domain and measured PN in frequency domain

Using the previously proposed model, we need to establish the relationship between the phase noise level injected in the time domain and the phase noise level measured in the frequency domain, which corresponds to the reference phase noise. To do that, we performed several simulations varying the noise level injected in the time domain, under different signal and measurement conditions.

Illustrative results are given in figures 2.13 and 2.14 which report the phase noise measured on the spectrum (PN) as a function of the injected phase noise level (σ_f) for different signal frequencies (f), and for two values of the measurement frequency offset (f_{offset}). These simulations reveal two distinct behaviors depending on the level of noise. For low values of σ_f we observe a linear relationship with a slope of about $+20dB/decade$, while for high values of σ_f we still have a linear relationship but with a slope of about $+40dB/decade$. Note that in both cases, the slope neither depends on the signal frequency nor on the frequency offset. Only the vertical location varies with these parameters.

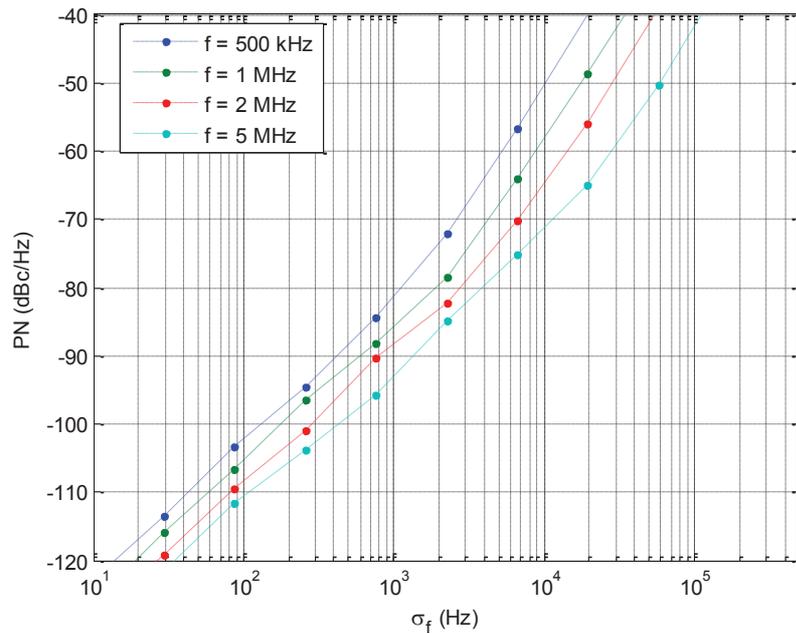


Figure 2.13 PN measured in frequency domain vs. PN injected in time domain, for different values of the signal frequency with $f_{offset} = 25kHz$

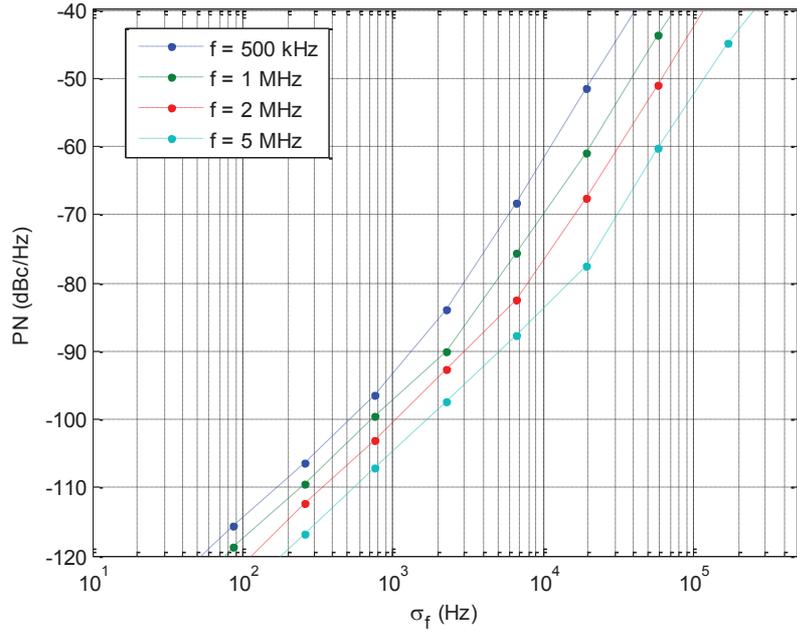


Figure 2.14 PN measured in frequency domain vs. PN injected in time domain, for different values of the signal frequency with $f_{offset} = 100kHz$

These relationships can be expressed with:

$$PN_1 = 20 \log_{10}(\sigma_f) + \alpha_1 \quad (12)$$

$$PN_2 = 20 \log_{10}(\sigma_f^2) + \alpha_2 \quad (13)$$

where Eq.(13) corresponds to the formulation for low level of injected time-domain phase noise, and Eq.(14) to the formulation for high level of injected noise.

The following step is to model, for both cases, the relation that links the intercept coefficient α to the signal frequency f_0 and the measurement frequency offset f_{offset} . For this we considered a fixed level of noise for both cases ($\sigma_f = 1kHz$ for low noise level and $\sigma_f = 50kHz$ for high noise level) and we performed simulations varying the signal frequency (f) for different values of the frequency offset (f_{offset}). Results are shown in figure 2.15 and 2.16 for the two noise levels.

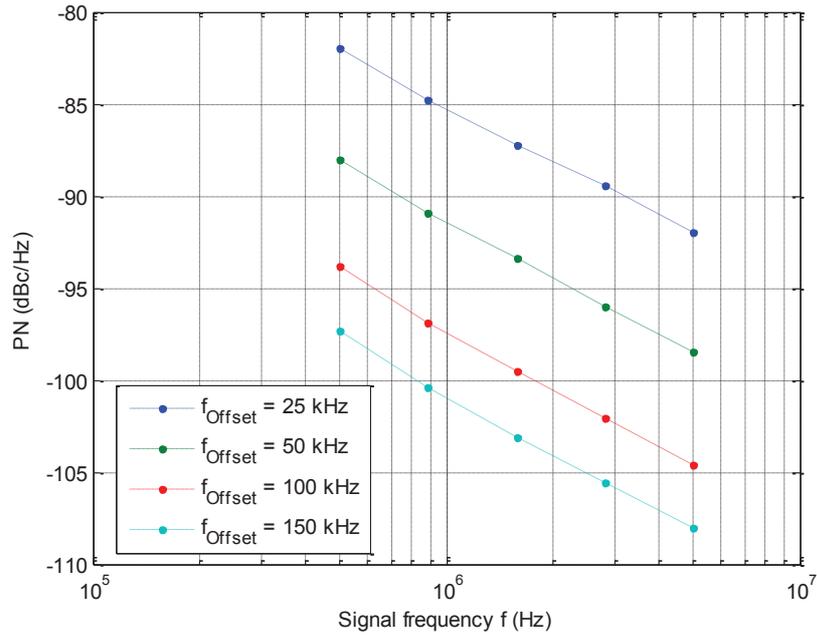


Figure 2.15 PN measured in frequency domain vs. signal frequency, for different values of the measurement frequency offset with $\sigma_f = 1\text{kHz}$ (low phase noise level)

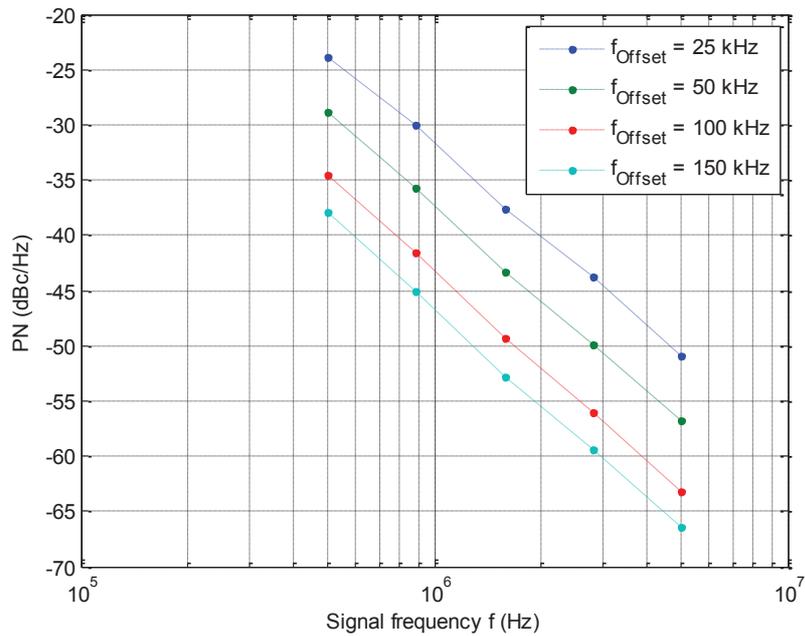


Figure 2.16 PN measured in frequency domain vs. signal frequency, for different values of the measurement frequency offset with $\sigma_f = 50\text{kHz}$ (high phase noise level)

We observe a linear relation on log-lin scale with a slope of about -10dB/decade for low noise level, which corresponds to α_1 varying with a $-10 \log_{10}(f)$ function. A slope of about -30dB/decade is observed high noises, which corresponds to α_2 varying with a $-30 \log_{10}(f)$ function. In both cases, the slope does not depend on the frequency offset, only the vertical intercept term. Consequently, we can express:

$$PN_1 = 20 \log_{10} \left(\frac{\sigma_f}{f^{1/2}} \right) + \beta_1 \quad (14)$$

$$PN_2 = 20 \log_{10} \left(\frac{\sigma_f^2}{f^{3/2}} \right) + \beta_2 \quad (15)$$

Figures 2.17 and 2.18 report the evolution of PN with respect to the frequency offset for different values of the signal frequency, in case of low and high noise levels respectively. In both cases, we observe a straight line with a slope of about -20dB/decade , leading to:

$$PN_1 = 20 \log_{10} \left(\frac{\sigma_f}{a_1 f^{1/2} f_{Offset}} \right) \quad (16)$$

$$PN_2 = 20 \log_{10} \left(\frac{\sigma_f^2}{a_2 f^{3/2} f_{Offset}} \right) \quad (17)$$

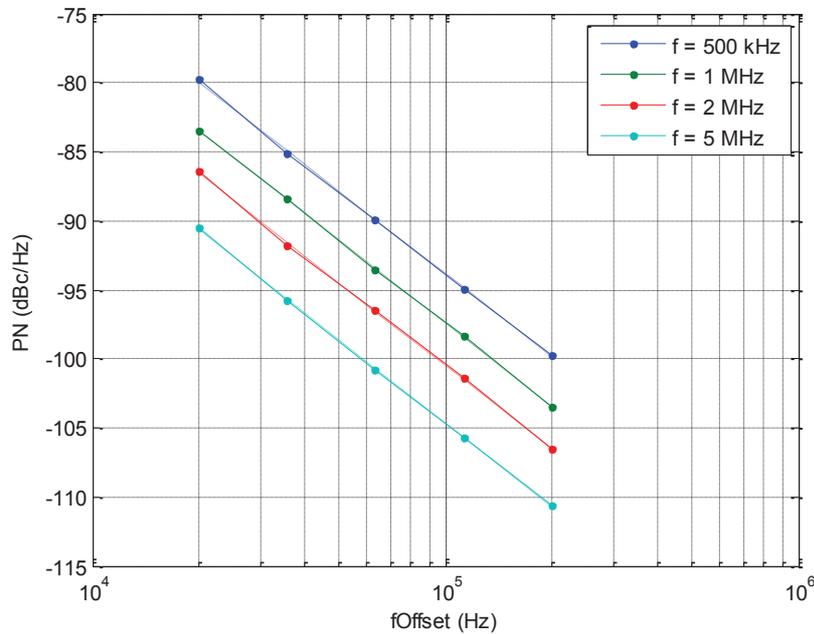


Figure 2.17 PN measured in frequency domain vs. measurement frequency offset for different values of the signal frequency with $\sigma_f = 1\text{kHz}$ (low phase noise level)

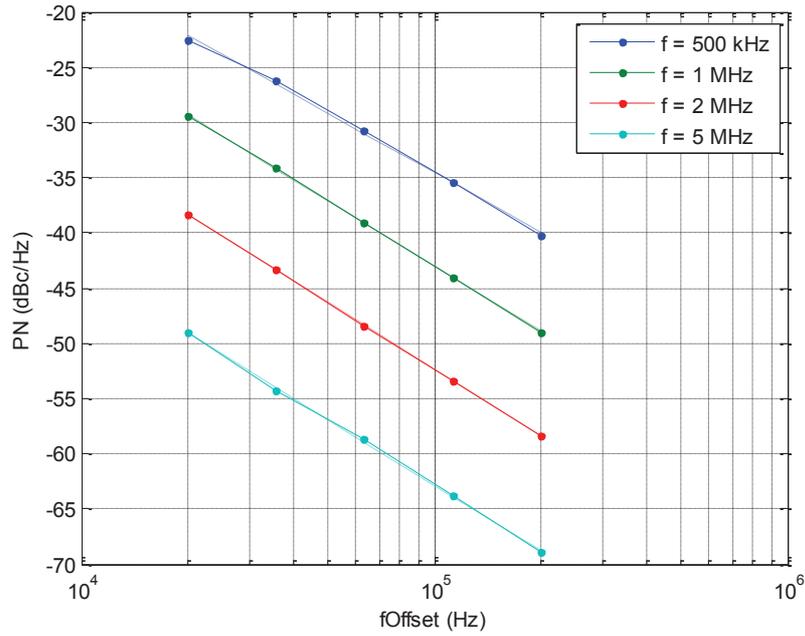


Figure 2.18 PN measured in frequency domain vs. measurement frequency offset for different values of the signal frequency with $\sigma_f = 50\text{kHz}$ (high phase noise level)

The terms a_1 and a_2 are then computed from best fit curve regression for the different values of f_{offset} . Analytical expressions corresponding to low and high levels of injected phase noise can then be obtained:

$$PN_1 = 20 \log_{10} \left(\frac{\sigma_f}{0.81 \cdot f^{1/2} \cdot f_{offset}} \right) \quad (18)$$

$$PN_2 = 20 \log_{10} \left(\frac{\sigma_f^2}{4.28 \cdot 10^{-3} \cdot f^{3/2} \cdot f_{offset}} \right) \quad (19)$$

Finally, the analytical relation between the noise injected in the time domain and the noise measured in the frequency domain is simply given by the maximum of the two expressions above:

$$PN = \max(PN_1, PN_2) \quad (20)$$

Figures 2.19 and 2.20 compare the phase noise value computed using this analytical expression to the phase noise value measured on the spectrum, for different values of the signal frequency (f), and for two values of the measurement frequency offset (f_{offset}). In all cases, a good agreement is observed.

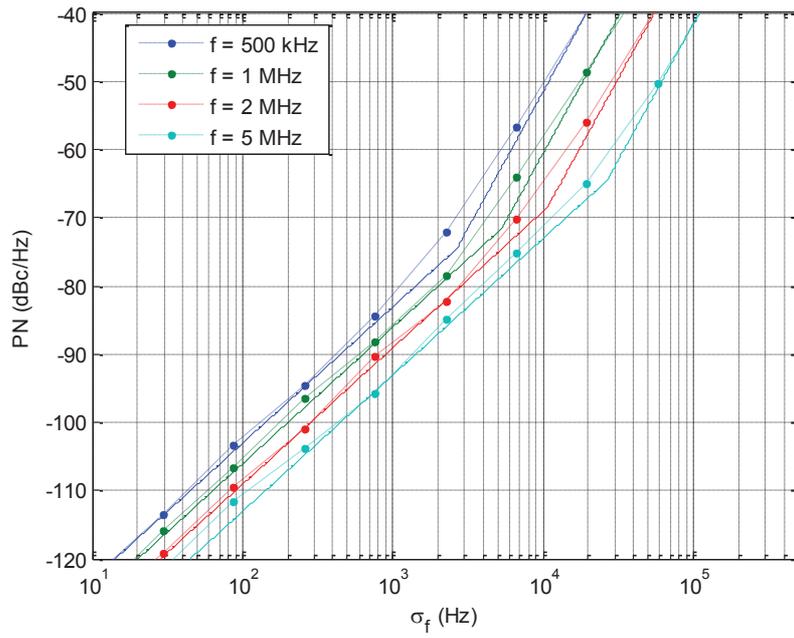


Figure 2.19 PN computed with analytical formulation vs. PN measured through FFT computation, $f_{offset} = 25kHz$

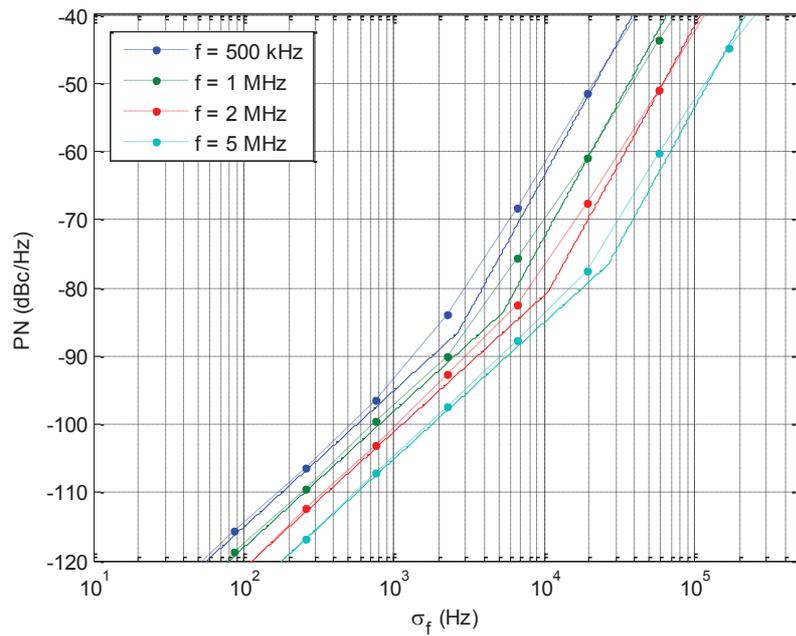


Figure 2.20 PN computed with analytical formulation vs. PN measured through FFT computation, $f_{offset} = 100kHz$

2.4 Conclusion

In this chapter, we have introduced a phase noise model based on a one-dimensional Brownian motion that permits phase noise injection in a time-domain signal. This model corresponds to phase noise with $1/f^2$ characteristics, which is typical of complex RF products that are the target of this thesis. The level of injected phase noise can be controlled by means of the standard deviation σ_f of the random variables involved in the Brownian motion. The validity of this model has been checked against a real device fabricated by NXP and an experimental setup allowing to synthesize a physical signal affected by phase noise has been developed. Finally, an analytical relationship between the phase noise level injected in the time domain and the phase noise level measured in the frequency domain has been established through a comprehensive simulation study.

Chapter 3. First method: phase noise estimation based on instantaneous frequency estimation

This chapter presents the first approach developed in this thesis in order to measure phase noise by means of a simple 1-bit capture. In a first part, background on frequency estimation from 1-bit capture is recalled. The principle of phase noise estimation based on an analysis of instantaneous frequency deviation is introduced in the second part. Then, integration of a measurement module is proposed as a solution toward Built-In Self-Test (BIST). Primary experiments are presented in the fourth part and limitations of the proposed technique are highlighted. An improved version of the technique that includes an additional filtering step is then developed and validated through experimental measurements on hardware signals.

3.1 Background on frequency estimation

3.1.1 Frequency estimation from 1-bit capture

The object of previous work [40] was to analyze RF signals with solely digital resources. One of the main contributions was the ability to recover frequency and amplitude of complex modulated signals using a simple 1 bit capture. Basics of frequency estimation are recalled here.

The analog signal $s(t)$ to be analyzed is first oversampled (f_s , sampling frequency) and converted to a digital signal by means of the comparator and the latch comprised in the digital tester channel. The resulting signal is therefore a binary vector $V[n]$ expressed with:

$$V(n) = \begin{cases} 1 & \text{if } s(nT_s) \geq 0 \\ 0 & \text{else} \end{cases} \quad n = 1 \dots N \quad (21)$$

where $T_s = 1/f_s$ is the sampling clock period, and N is the total number of captured samples. This binary vector stored in the ATE memory is then processed in order to retrieve the analog signal characteristics.

One can exploit the transitions of the captured binary vector $V[n]$ that correspond to the zero-crossings of the analog signal $s(t)$ in order to compute an estimation of the instantaneous period $\hat{T}(m)$. This estimation can be simply obtained by counting the number of samples between successive transitions of the binary vector, which corresponds to the signal half-period:

$$\begin{aligned} \text{count}(m) &= \left[\begin{array}{l} Nb_samples \\ \text{transition}(m+1) \\ \text{transition}(m) \end{array} \right] \\ \hat{T}(m) &= 2 T_s \text{count}(m) \quad m = 1 \dots 2(Nper - 1) \end{aligned} \quad (22)$$

where $Nper$ is the number of captured signal periods.

This is illustrated in figure 3.1 where the analog signal is represented in blue and $V[n]$ is represented by red dots. The zero-crossing times of the analog signal can be inferred from the knowledge of rise and fall transition times of the binary vector.

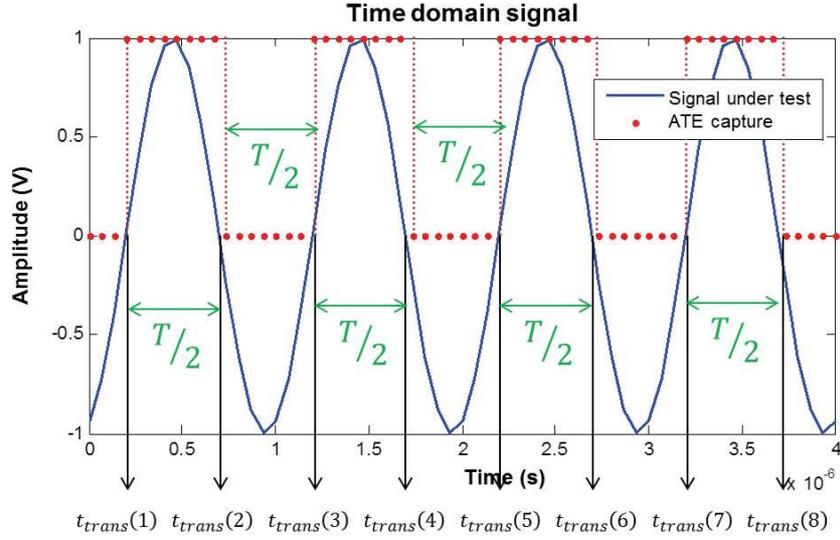


Figure 3.1 1-bit acquisition of the signal under test. The transitions of the binary vector are used to estimate the instantaneous period.

The instantaneous frequency is then simply given by:

$$\hat{f}(m) = \frac{1}{2 T_s \text{count}(m)} \quad (23)$$

With this process, we obtain $2(N_{per} - 1)$ values of the instantaneous frequency. Note that an estimation of the signal frequency can be computed as the arithmetic mean of instantaneous frequency estimations:

$$\bar{f} = \frac{1}{M} \sum_{m=1}^M \frac{1}{2 T_s \text{count}(m)} \quad (24)$$

3.1.2 Discretization quantum

The sampling process of the analog signal $s(t)$ introduces a quantization of the estimated values for the instantaneous period or frequency. Indeed as expressed by eq.(22), estimated values of the instantaneous period can take only even integer multiple values of the sampling period. The quantization step of the instantaneous period $\hat{T}(m)$ is therefore:

$$Q_T = 2 T_s \quad (25)$$

The quantization step of the instantaneous frequency $\hat{f}(m)$ can be calculated with a differentiation of $f = 1/T$:

$$df = -\frac{dT}{T^2} \quad (26)$$

So the quantization step of $\hat{f}(m)$ is:

$$Q_f = \left| -\frac{Q_T}{T^2} \right| = \frac{2 T_s}{T^2} = \frac{2 f^2}{f_s} \quad (27)$$

3.1.3 Running average filtering

Practically, the signal is noisy in amplitude. In case of high amplitude noise level, the binary vector could be affected by multiple transitions in the zero crossing neighborhoods as illustrated in figure 3.2. In order to get rid of these potential perturbation areas, basic filtering is implemented [40].

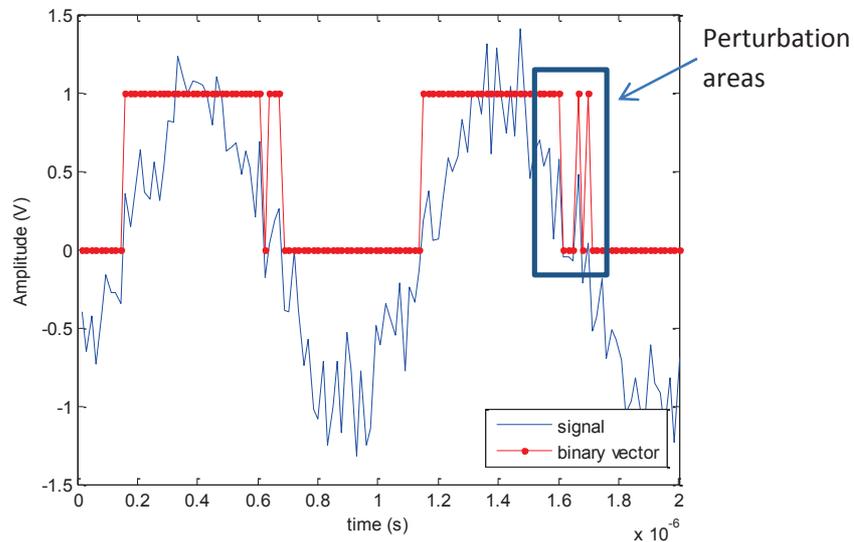


Figure 3.2 Amplitude noise may cause perturbation in the binary vector near transitions

A simple running average over the binary vector does the job. The running average is performed over a window of N_{avg} samples:

$$\tilde{V}[n] = \frac{1}{N_{avg}} \sum_{i=n-N_{avg}}^n V[i] \quad (28)$$

The result of this averaging is compared to 0.5 in order to locate zero-crossing events and obtain another binary vector $V_1[n]$ that presents a single transition associated to each zero-crossing event.

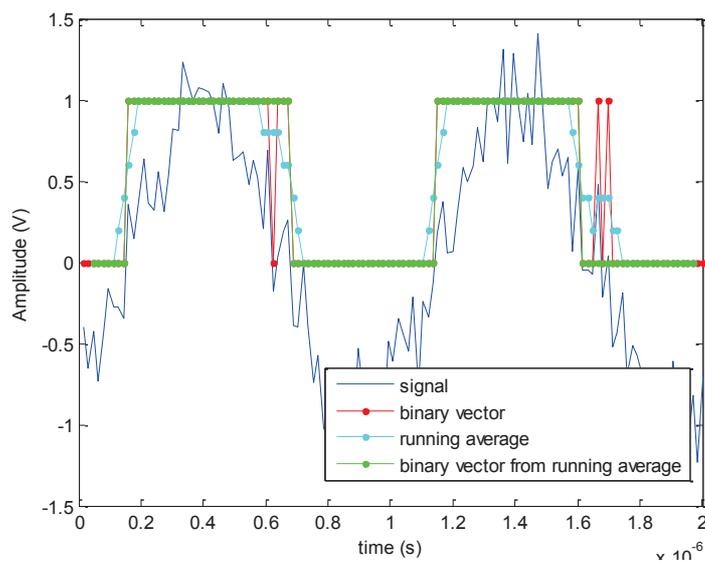


Figure 3.3 Perturbation in the binary vector can be filtered with a running average

Figure 3.3 illustrates how the binary vector computed from the running average isolates one single transition from the perturbation area where there was multiple transitions. This algorithm is robust and proven to be well functioning for N_{avg} around $\frac{f_s}{4f}$ as a rule of thumb.

3.1.4 Error on threshold level

In practice we want to be robust to any error on the threshold voltage of the comparator that performs 1-bit quantization. A simple solution is to consider time of extremums rather than transition time stamps:

$$\widehat{t_{extr}}(i) = \frac{\widehat{t_{trans}}(i) + \widehat{t_{trans}}(i-1)}{2} \quad (29)$$

This way, any offset error on the comparator threshold will be cancelled by the sum between two successive t_{trans} . Then we have:

$$count(m) = \begin{bmatrix} Nb_samples \\ \end{bmatrix} \begin{matrix} t_{extr}(m+1) \\ t_{extr}(m) \end{matrix} \quad (30)$$

for the instantaneous frequency estimations in eq.(22).

Note that this is equivalent to distinguish rising and falling transitions and separately count the number of samples between successive rising events $count_R(m)$ and successive falling events $count_F(m)$, which directly corresponds to the signal period. In this case, the instantaneous frequency is given by:

$$\hat{f}(m) = \frac{1}{T_s count_R(m)}, \quad \hat{f}(m) = \frac{1}{T_s count_F(m)} \quad (31)$$

and quantization step of the instantaneous frequency Q_f changes since the frequency is estimated over one whole period to:

$$Q_f = \frac{f^2}{f_s} \quad (32)$$

3.2 Statistical dispersion on instantaneous frequency estimations

From the oversampled 1-bit acquisition of an analog signal, we have established in the previous section how to compute the instantaneous frequency, with two estimations per signal period. In case of an ideal sinusoidal signal with no phase noise, the instantaneous frequency should exhibit the same value at each period of the signal, while in case of a sinusoidal signal affected with phase fluctuation, we expect a dispersion on the computed values. It is therefore our objective to analyze the distribution of the instantaneous frequency in order to derive an estimation of the phase noise affecting the analog signal.

3.2.1 Standard deviation

The first natural idea to analyze the distribution of the estimated instantaneous frequency is to compute its standard deviation. Standard deviation is indeed a very commonly-used measure of statistical dispersion. This standard deviation is defined by:

$$\sigma(\hat{f}) = \sqrt{\frac{1}{2N_{per} - 3} \sum_{m=1}^{2(N_{per}-1)} (\hat{f}(m) - \bar{f})^2} \quad (33)$$

where \bar{f} is the mean of the instantaneous frequency estimations.

Figure 3.4 gives the evolution of the instantaneous frequency standard deviation $\sigma(\hat{f})$ according to the injected phase noise level σ_f for different values of the oversampling ratio f_s/f , in case of a 1 MHz signal. Analyzing these results reveals that there is a direct correlation between $\sigma(\hat{f})$ and σ_f for high level of injected phase noise and high oversampling ratio. For instance with an oversampling ratio of 400, there is a perfect match with an ideal straight line for noise level higher than 2 000Hz. In this region, we can directly estimate the phase noise level injected in the time-domain signal with:

$$\hat{\sigma}_f = \sigma(\hat{f}) \quad (34)$$

However validity of this simple correlation is lost when decreasing the oversampling ratio. For instance with an oversampling ratio of 50, Eq.(34) is valid only for noise level higher than 10 000Hz.

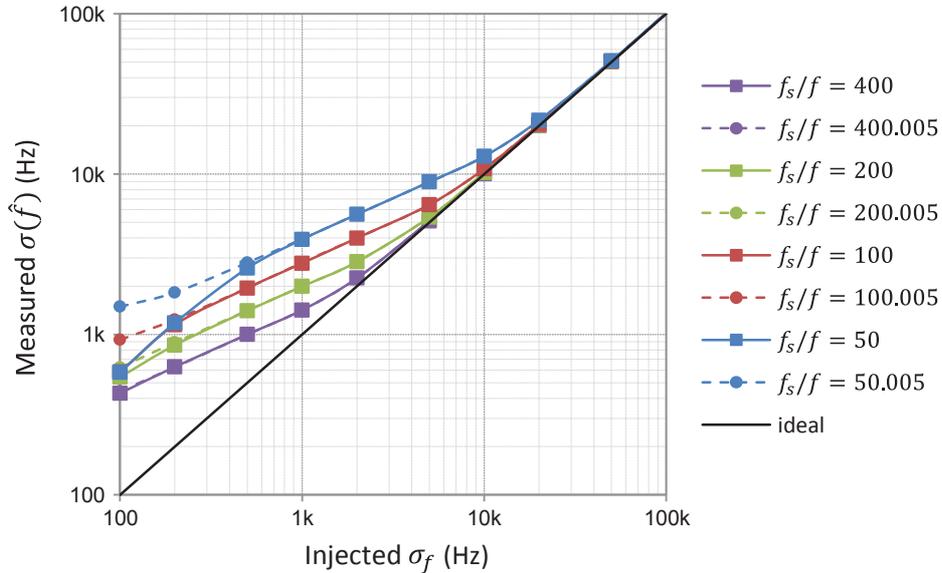


Figure 3.4 Standard deviation of estimated instantaneous frequency vs. injected phase noise level, for different values of the oversampling ratio

This phenomenon comes from the quantization of the estimated instantaneous frequency introduced by the sampling process, with a quantization step equal to $Q_f = f^2/f_s$. As a result, the distribution of the estimated instantaneous frequency is a discrete distribution. Figure 3.5 illustrates this distribution for 3 different values of the oversampling ratio, with a 1% phase noise level injected in a 1MHz signal ($\sigma_f = 10kHz$). In case of a low oversampling ratio of 25, the quantization step is large ($Q_f = 40kHz$) and the discrete distribution is evaluated with a low resolution. In contrast with a larger oversampling ratio of 400, the quantization step is small ($Q_f = 2.5kHz$) and the discrete distribution is obtained with a fine resolution. Direct evaluation of the phase noise level can be achieved using the standard deviation only if the resolution is sufficient, i.e. if the quantization step is smaller than the phase noise level to be estimated. Practical use of standard deviation for phase noise evaluation is therefore limited. For instance, the correct evaluation of -90dBc/Hz phase noise at

25kHz frequency offset on a 1MHz signal ($\sigma_f = 640\text{Hz}$ from Eq.(18)) would require a sampling frequency higher than 1.56GHz, which exceeds the capabilities of a standard digital tester channel.

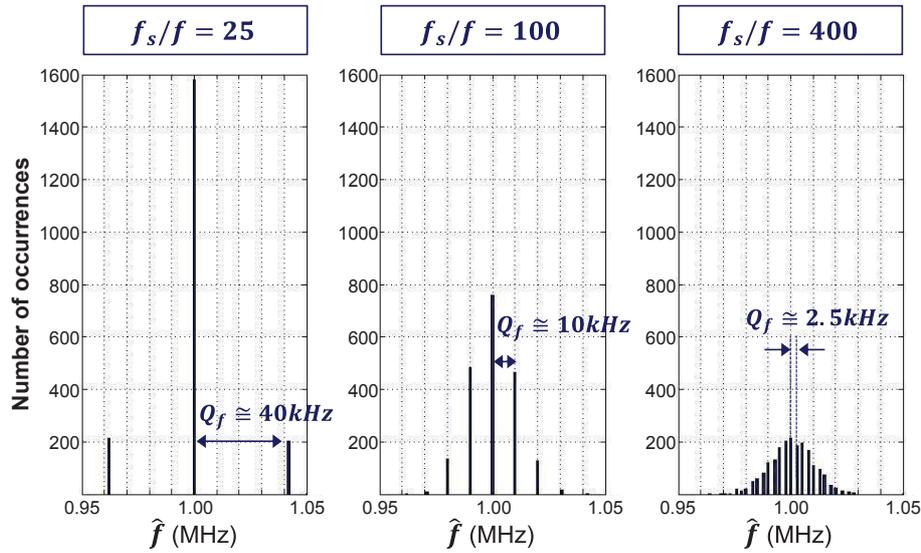


Figure 3.5 Effect of frequency quantization on distribution of estimated instantaneous frequency

3.2.2 Mean Absolute Deviation (MAD)

In this context, we have looked for another estimator that can apply to our specific case of discrete distribution evaluated with low resolution. Other than the standard deviation, many different measures of statistical dispersion exist, such as the interquartile range, the mean difference, the median absolute deviation, the mean absolute deviation, the distance standard deviation... Among these classical measures, we have identified that the Mean Absolute Deviation (MAD) is a measure that can be easily computed and that permits robust estimation in our specific context.

The Mean Absolute Deviation (MAD) is defined as the mean of the absolute deviations of a set of data measured apart from the data mean value. In our case, we have:

$$MAD(\hat{f}) = \frac{1}{2(N_{per} - 1)} \sum_{m=1}^{2(N_{per}-1)} |\hat{f}(m) - \bar{\hat{f}}| \quad (35)$$

In case of a normal distribution, the MAD is related to the standard deviation with:

$$MAD(\hat{f}) = \sqrt{\frac{2}{\pi}} \cdot \sigma(\hat{f}) \quad (36)$$

Consequently, we propose to estimate the phase noise injected in the time-domain signal with:

$$\hat{\sigma}_f = \sqrt{\frac{\pi}{2}} \cdot MAD(\hat{f}) \quad (37)$$

Replacing \hat{f} by its expression and rearranging the equation, we obtain:

$$\hat{\sigma}_f = \sqrt{\frac{\pi}{2}} \cdot \frac{Q_f}{2(N_{per} - 1)} \sum_{m=1}^{2(N_{per}-1)} |count(m) - n| \quad (38)$$

where $Q_f = f^2/f_s$ is the frequency quantization step, N_{per} is the number of captured signal periods, $n = f_s/f$ is the oversampling ratio, and $count(m)$ is the number of samples between successive extremums.

Note that this analytical relationship has been derived based on an intuitive use of the MAD function. The same expression can be established based on a stochastic model of the measurement process, as detailed in Annex A.

Figure 3.6 reports the estimated phase noise level $\hat{\sigma}_f$ vs. the injected phase noise level σ_f , for different values of the oversampling ratio f_s/f . It can be observed that all curves obtained with an integer oversampling ratio perfectly match the ideal straight line, whatever the integer value. In case the oversampling ratio is not exactly an integer, curves deviate from the ideal straight line for low level of injected phase noise. These results clearly indicate that the proposed estimator has the potential to achieve correct phase noise even for low level of noise and low oversampling ratio, provided that the sampling frequency is a perfect multiple of the signal frequency.

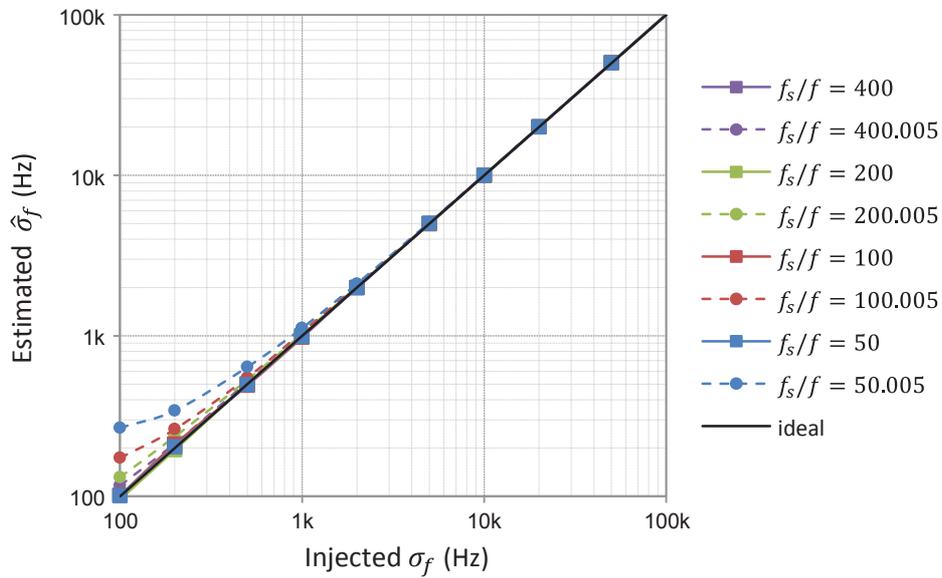


Figure 3.6 Estimation of PN level based on Mean Absolute Deviation of estimated instantaneous frequency

3.2.3 Considerations for sampling ratio setting

The proposed technique relies on the use on an integer oversampling ratio $= \frac{f_s}{f}$. This ratio should preferably be chosen odd. The reason for this is illustrated in figure 3.7.

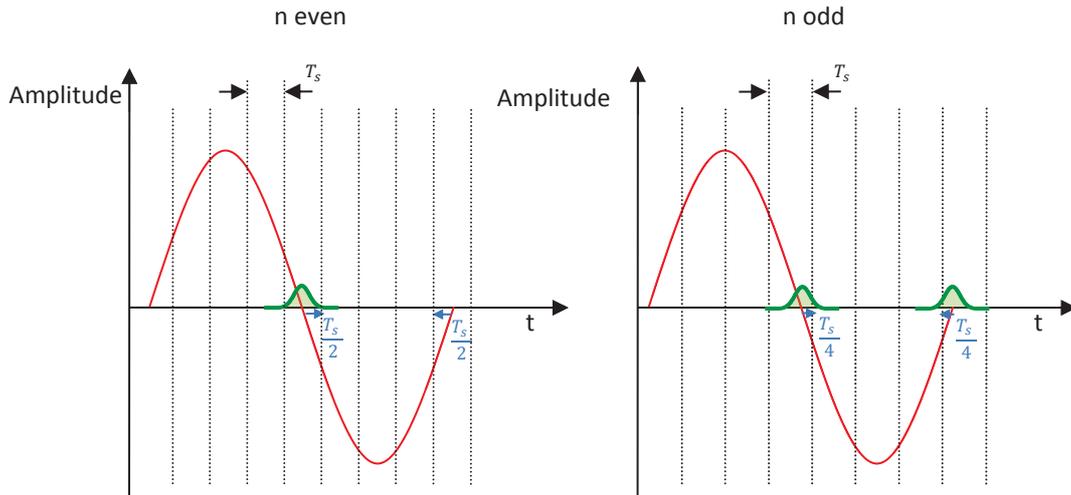


Figure 3.7 The maximum distance between a zero-crossing event and the closest sampling point is reduced by choosing an odd oversampling ratio n

Considering an ideal sinewave, there are two zero-crossing events per period, one corresponding to the falling slope of the other to the rising slope of the analog signal. The maximum distance between the actual location of these crossing events and the closest sampling points actually depends on the choice of the sampling ratio as odd or even. In case of an even number, this maximum distance is equal to $\frac{T_s}{2}$ while in case of an odd number, it cannot exceed $\frac{T_s}{4}$. This means that with the same phase noise, represented as a Gaussian distribution in green, the signal is much more likely to be detected when n is odd.

To illustrate this point, we have simulated a 1MHz signal affected with phase noise ($\sigma_f = 1kHz$) and sampled using either an even ($n = 200$) or odd ($n = 199$) ratio. In both cases, the phase noise level $\hat{\sigma}_f$ present in the signal has been estimated using eq.(38). The experiment has been repeated 500 times and we have computed the standard error on the estimated phase noise values $SE(\hat{\sigma}_f)$. Results are summarized in figure 3.8 which reports the distribution of estimated values. It clearly appears that correct estimation is achieved in both cases with a distribution well-centered on the expected value of 1kHz. However the dispersion is lower in case the oversampling ratio is odd rather than even, which corresponds to a reduced measurement uncertainty.

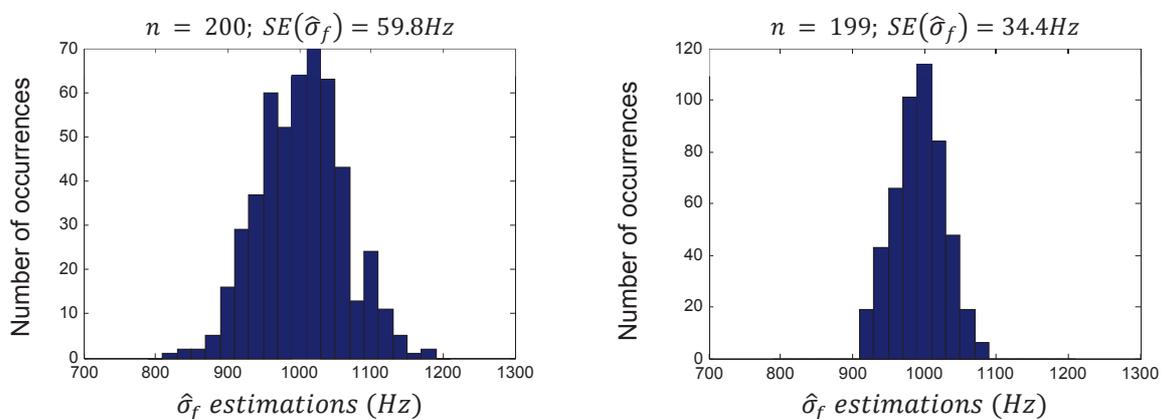


Figure 3.8 Distributions of $\hat{\sigma}_f$ estimations obtained in simulation with $n = 200$ and $n = 199$

Finally, we have evaluated the impact of inaccuracy on the setting of the oversampling ratio. The number of instantaneous frequency deviations due to an inaccurate oversampling ratio $n' = \frac{f_s + \Delta f_s}{f}$ should obviously be much smaller than the number of instantaneous frequency deviations generated by phase noise in order to not disturb the estimation. We have established that the sampling frequency error Δf_s should respect the following constraint:

$$\Delta f_s \ll \frac{n}{\sqrt{2\pi}} \sigma_f|_{target} \quad (39)$$

where n is the ideal integer oversampling ratio and $\sigma_f|_{target}$ is the typical phase noise level to be measured on the signal under test.

3.2.4 Summary of proposed digital technique

The proposed technique relies on the 1-bit acquisition of the signal to be analyzed with a sampling frequency that is an integer multiple of the signal frequency, associated with simple digital processing to compute the mean absolute deviation of the estimated instantaneous frequency. The technique therefore requires that the signal frequency is known with a good accuracy.

If necessary, the practical implementation may include a preliminary phase dedicated to the determination of the signal frequency before the phase noise test phase, as illustrated in figure 3.9.

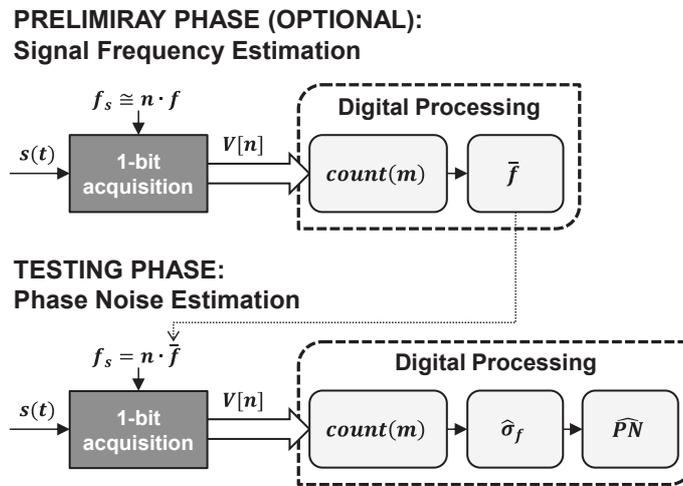


Figure 3.9 Block diagram of the proposed digital technique based on instantaneous frequency estimation

In the preliminary phase, 1-bit acquisition of the signal is performed with a sampling frequency that is not necessarily an integer multiple of the signal frequency. From this acquisition, the signal frequency is computed as the arithmetic mean of the instantaneous frequency estimations using eq.(28).

In the testing phase, a novel 1-bit acquisition of the signal is performed with the sampling frequency adjusted at an integer multiple of the computed signal frequency $f_s = n \cdot \bar{f}$. From this acquisition, we compute the estimated value of the phase noise level $\hat{\sigma}_f$ using eq.(38). The final estimation of the phase noise value \widehat{PN} expressed in dBc/Hz is then obtained from eq.(20).

3.3 Integrated module

3.3.1 Integrated module overview and principle

We investigated ways to integrate this technique on chip, in order to perform phase noise measurement directly within the Device Under Test (DUT). The objective is to develop an Embedded Test Instrument (ETI) that delivers a digital signature directly related to the phase noise level present in the analog output signal of the DUT, as shown in Fig.3.10.a. However the direct integration of the method described in previous section is not feasible since it requires excessive area with respect to the area occupied by the DUT itself. Indeed, although the 1-bit digitizer can be implemented with small area using a comparator for instance, the computation of the phase noise level necessitates the integration of both important memory resources and DSP to perform the required arithmetic operations. Our purpose is therefore to redefine the post-processing algorithm in order to implement the computation of the digital signature using a limited number of memory resources and only simple binary operators, as illustrated in Fig.3.10.b.

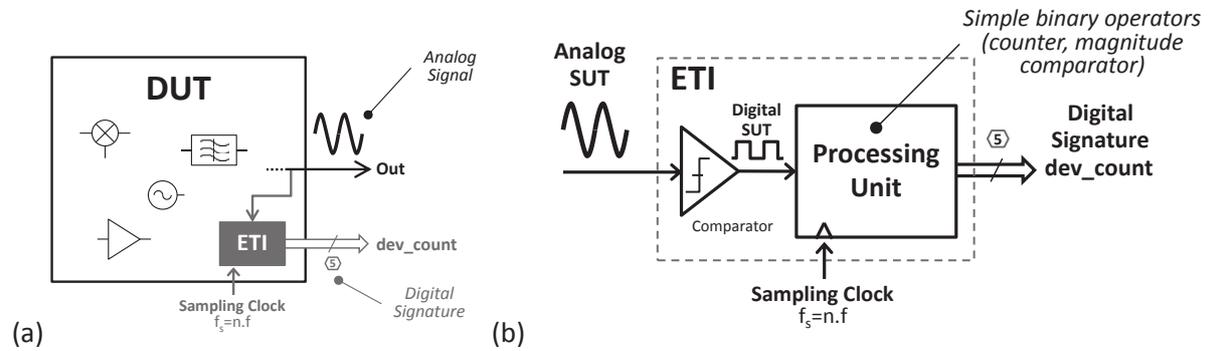


Figure 3.10 Embedded Test Instrument for phase noise measurement

In order to identify an appropriate digital signature, let us first define $dev(m) = count(m) - n$ as the deviation from the ideal count at each estimated value of the instantaneous frequency, where $count(m)$ corresponds to the number of samples between successive transitions of same type (either rising or falling transitions). The phase noise estimator defined in eq.(38) can then be expressed with:

$$\hat{\sigma}_f = \sqrt{\frac{\pi}{2}} \cdot \frac{Q_f}{2(N_{per} - 1)} \cdot \sum_{m=1}^{2(N_{per}-1)} |dev(m)| \quad (40)$$

This equation clearly indicates that the basic operation to be performed for on-chip implementation is a sum of absolute deviations, which can be considered as the digital signature of phase noise level present in the analog signal:

$$dev_count = \sum_{m=1}^{2(N_{per}-1)} |dev(m)| \quad (41)$$

In order to perform the computation of this digital signature with minimal hardware resources, the key idea is to implement on-the-fly processing of the 1-bit signal delivered at the output of the comparator. In particular, the idea is to generate a pulse train signal whose pulse durations correspond to differences between the actual period duration and a reference based on the ideal

period length. The digital signature related to the phase noise level can then be computed with a simple accumulation of these deviations over a number of signal periods.

The main principle for generating such pulse train is based on the XOR-comparison between dedicated signals triggered by specific SUT transitions. As depicted in figure.3.11, it involves:

- The generation of a signal *signal_R*, that toggles at every rising zero-crossing of the SUT,
- The generation of :
 - a first reference signal *R_trig_ref_signal_R* whose rising transition is triggered by the rising transition of *signal_R* and duration at high level is equal to the ideal signal period,
 - a second reference signal *F_trig_ref_signal_R* whose falling transition is triggered by the falling transition of *signal_R* and duration at low level is equal to the ideal signal period,
- The XOR-comparison between
 - *signal_R* and *R_trig_ref_signal_R*
 - *signal_R* and *F_trig_ref_signal_R*,
- The logic OR of the two XOR-comparison results.

The result of the logic OR is then a pulse signal *dev_signal_R* whose pulse durations correspond to deviations between SUT rise-to-rise transitions and the ideal period.

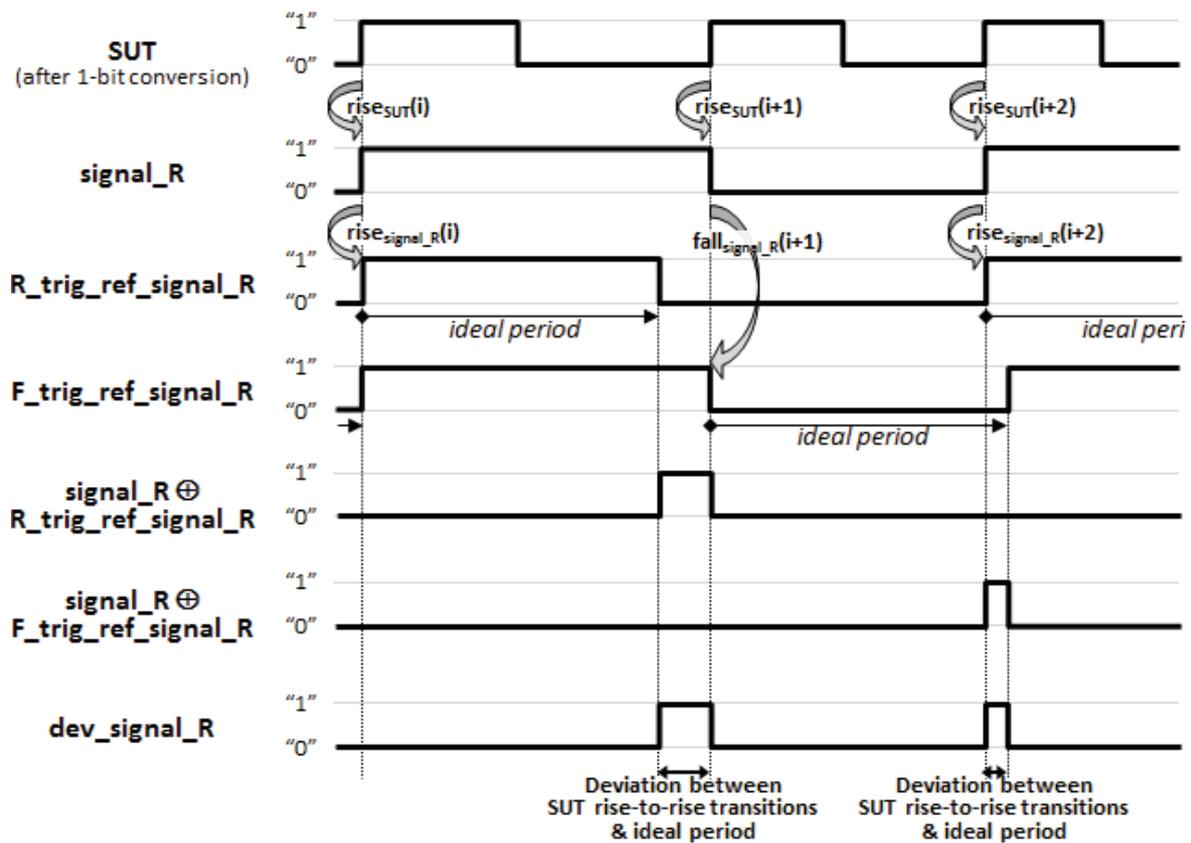


Figure 3.11 Principle of deviation signal generation (associated with SUT rise-to-rise transitions)

In the same way we can define a pulse signal dev_signal_F whose pulse durations correspond to deviations between SUT fall-to-fall transitions and the ideal period, based on the generation of a signal $signal_F$ whose toggling is triggered by successive falling transitions of the SUT.

The deviation signal dev_signal that includes all types of period deviations is then simply the result of a logic OR between the two pulse signals dev_signal_R and dev_signal_F . The digital signature dev_count can be computed with a simple accumulation of dev_signal high states over a number of signal periods.

The general architecture of such an Embedded Test Instrument is given in figure 3.12. It comprises two “Divide-by-2” blocks, one directly connected to the digital SUT and the other to the inverted digital SUT, which deliver $signal_R$ and $signal_F$ respectively. Each one of these two signals is fed to two reference signal generators, one triggered on the positive edge and the other on the negative edge. The deviation signal dev_signal is then generated through simple combinational gates, and accumulated by the “Cum_Sum” block to deliver the digital signature dev_count .

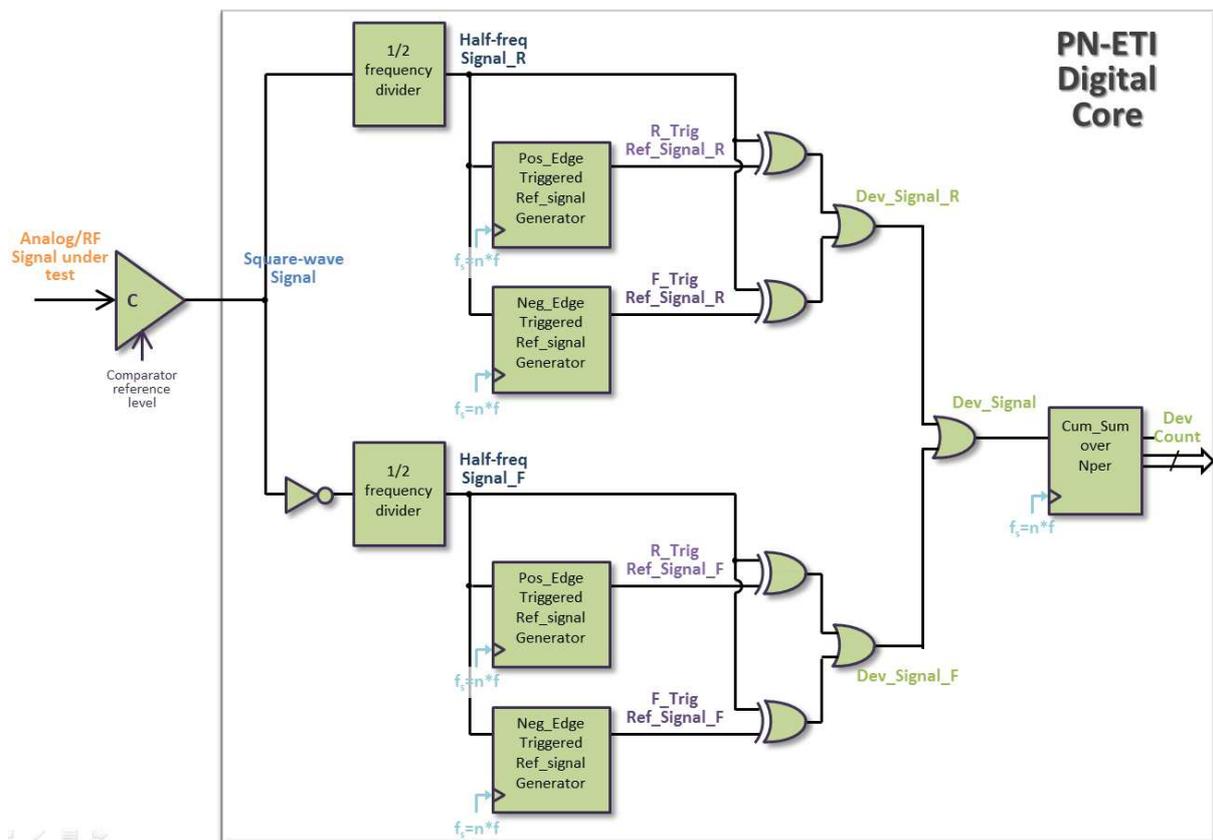


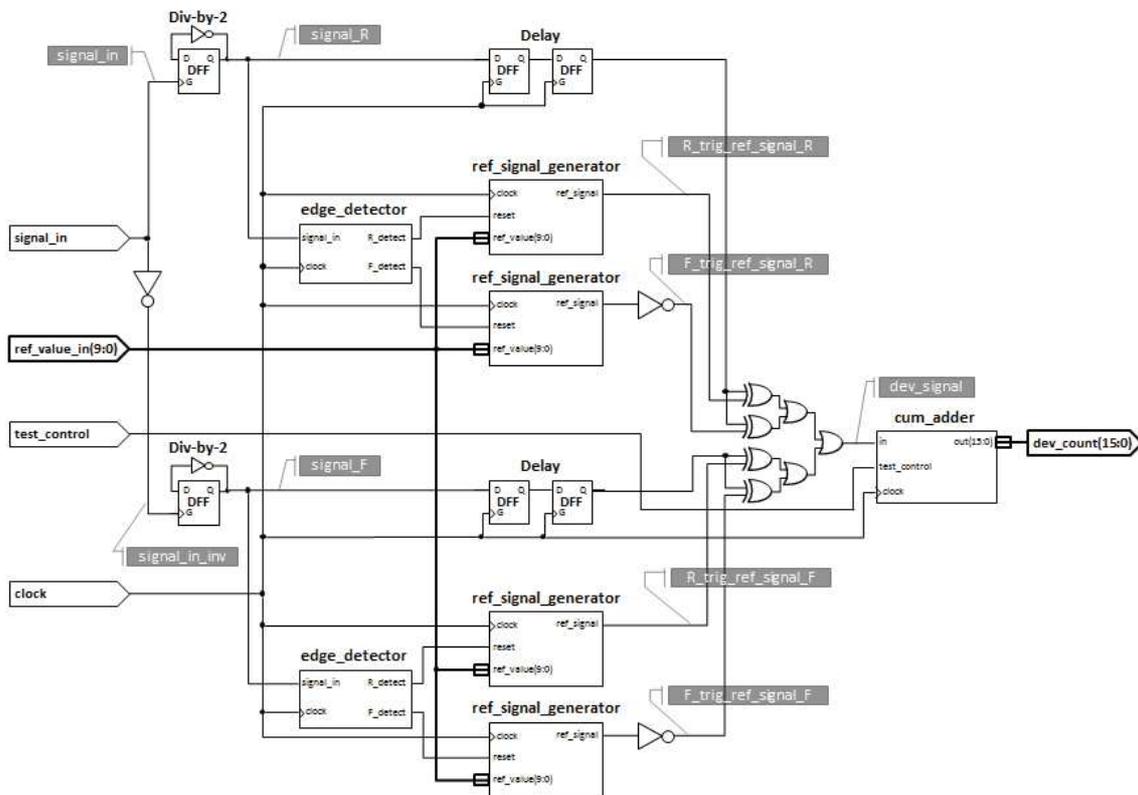
Figure 3.12 General architecture of the proposed ETI

3.3.2 Implementation

Several ways of implementing the general architecture of the proposed ETI are obviously possible. This section describes a simple implementation that leads to small silicon area.

Figure 3.13 gives a detailed block diagram of the proposed implementation of the processing unit together with the functional timing sequence. The module inputs and output are:

- *signal_in* is the digitized input signal, taken at the output of the comparator, corresponding to the 1-bit conversion of the analog signal to be measured.
- *clock* is a reference clock with a frequency f_s that is an integer multiple n of the signal frequency f .
- *ref_value_in* is a configuration value to be provided to the module by mean of a pre-loaded register for instance. It should be equal to the oversampling ratio $n = f_s/f$ and controls the duration of the reference signals.
- *test_control* is a test input to enable the measure for a specific period of time.
- *dev_count* is the digital signature correlated to the phase noise to be measured; it can be read from a register.



Functional Timing Sequence

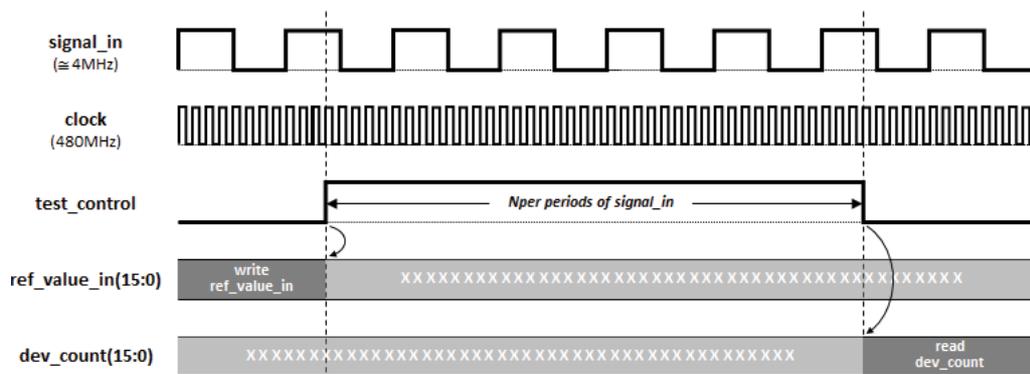


Figure 3.13 Block diagram of the processing unit

Note that the “Divide-by-2” block is simply implemented using a D-FlipFlop (DFF) with a feedback loop between the inverted output \bar{Q} and the data input D . Also note that delay elements are inserted to ensure appropriate synchronization between signals delivered by the reference generators and signals delivered by the “Divide-by-2” blocks, with respect to the subsequent XOR-comparison.

Figure 3.14 shows the implementation of the “edge detector” block and its functional timing diagram. It simply consists of 2 DFFs, 2 inverters and 2 AND gates. Two instances of this block are used to generate triggering signals (R_detect and F_detect) for the reference generators.

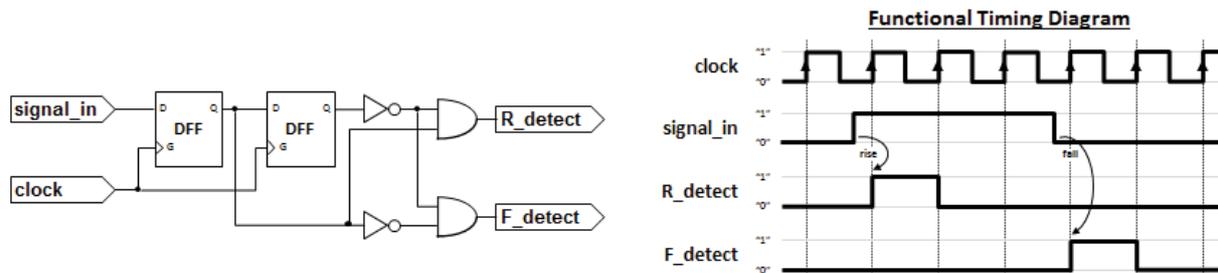


Figure 3.14 “egde_detector” block

Figure 3.15 shows the implementation of “ref_signal_generator” block and its functional timing diagram. It is based on a free-running counter (count enable input CE tied to “1”) and a data comparator ($LT = "1"$ if $A < B$). Signal generation is triggered by the reset input R of the counter (connected to R/F_detect coming from the “edge detector” block) and the duration at high level is determined by the configuration value ref_value_in ($ref_signal = "1"$ while $count_value < ref_value_in$). Four instances of this block are used to generate $R_trig_ref_signal_R$, $F_trig_ref_signal_R$, $R_trig_ref_signal_F$ and $F_trig_ref_signal_F$. In case of reference signals triggered by falling transitions and with duration at low level equal to the ideal signal period, the output of the “ref_signal_generator” block is inverted.

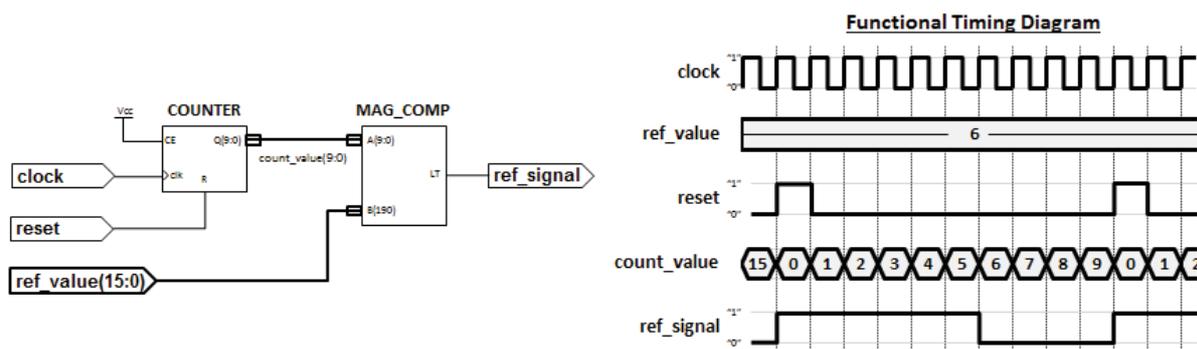


Figure 3.15 “ref_signal_generator” block

Figure 3.16 shows the implementation of the “cum_adder” block and its functional timing diagram. Its role is to count the number of high levels contained in the deviation signal dev_signal over a given number of signal periods. It is based on a counter with control of the count enable input CE , i.e. the counter is incremented if both $test_control$ and in are at high level, otherwise it maintains the current value. Therefore, the duration of the cumulative sum computation is determined by $test_control$. In addition, a reset of the counter is performed at the beginning of the computation with the rising edge of $test_control$, and the counter value is maintained at the end of the

computation with *test_control* at low level. This value directly corresponds to the digital signature *dev_count*. One instance of this block is used, with its input connected to *dev_signal* and its output to *dev_count*.

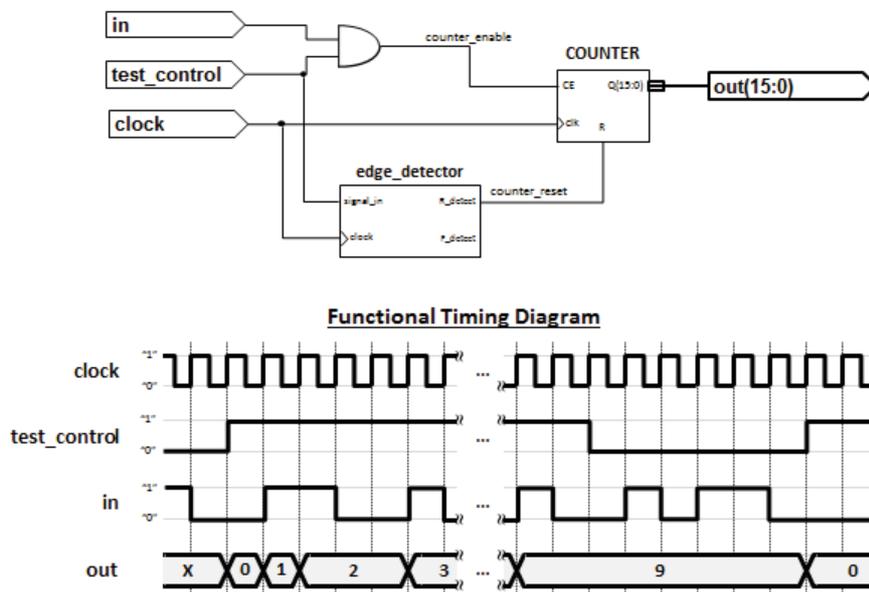


Figure 3.16 “cum_adder” block

Table 3.17 summarizes the required hardware resources in terms of sequential and combinational elements, assuming a 10-bit counter for the “ref_signal_generator” blocks (oversampling ratio *n* from 1 up to 512), and a 16-bit counter for the “cum_adder” block (digital signature *dev_count* from 0 up 65,536). Overall, the processing unit comprises about 70 FFs and about 400 combinational gates, which corresponds to a very small silicon area.

Block	Number of instances	Number of FFs	of	Number of gates
Edge_detector	2	2		4
Ref_signal_generator	4	10		84
Cum_adder	1	20		52
Other elements	1	6		12
Total		70		408

Table 3.1 Hardware resources involved in Processing Unit

Structural VHDL simulations have been run to verify the correct behavior of the module. An example is given in figure.3.17. The proposed ETI has then been implemented within a test chip using a CMOS 140nm technology. The module occupies only 7,885 μm^2 , which a represents negligible area of the test chip as illustrated in figure 3.18.

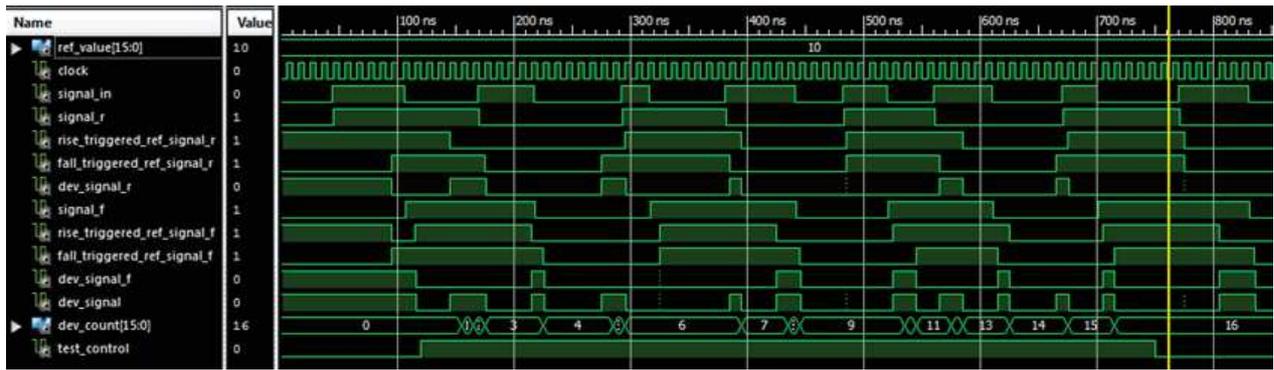


Figure 3.17 VHDH simulation example

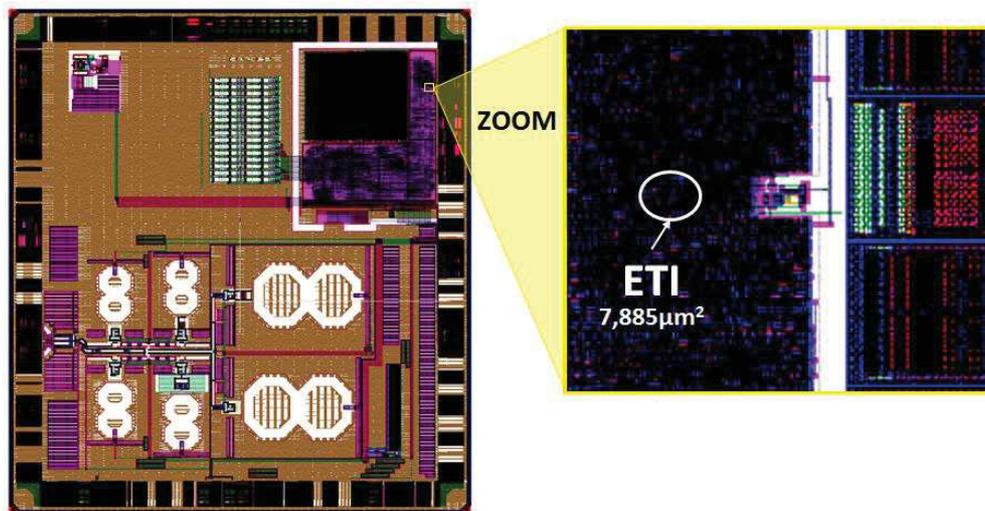


Figure 3.18 Layout of the test chip with embedded test instrument

This prototype has been taped out and we have tried to perform measurements. Unfortunately, the device is not functional, possibly due to a design error on the bus that addresses the configuration register storing *ref_value*.

3.4 Preliminary experimental results

The measurement technique has been further explored by performing measurements on a physically. The signal is generated and captured as previously explained. Both the conventional method (i.e. FFT based) and the proposed digital technique have been applied on this signal for different noise levels.

3.4.1 Experimental setup

In order to compare simulation results with actual measurements and explore the valid measurement range of the technique, the test bench depicted in figure 3.19 has been developed. As already described in section 2.2.1, the generation of synthesized signal with tunable phase noise is accomplished using an RF source, FM-modulated with white noise. Phase noise level can be adjusted by means of the modulation depth parameter.

The capture of the SUT is realized using a Yokogawa DLM2054 Mixed-Signal Oscilloscope (MSO). In the context of an industrial production test, capture would be done with an analog/RF tester channel

for the conventional method, and with a standard digital tester channel for the digital technique. Here, the MSO replaces the digitizer of the analog/RF tester channel, and 1-bit acquisition normally performed by the digital ATE is done through software 1-bit quantization on the digitized samples provided by the MSO.

Two distinct post-processing procedures are then implemented in Matlab. The first one corresponds to the conventional method, i.e. FFT is applied on the digitized samples and the phase noise value PN is measured on the resulting spectrum. The second one corresponds to the digital technique, i.e. the dedicated post-processing algorithm is applied on the binary vector to compute the estimated phase noise value \widehat{PN} .

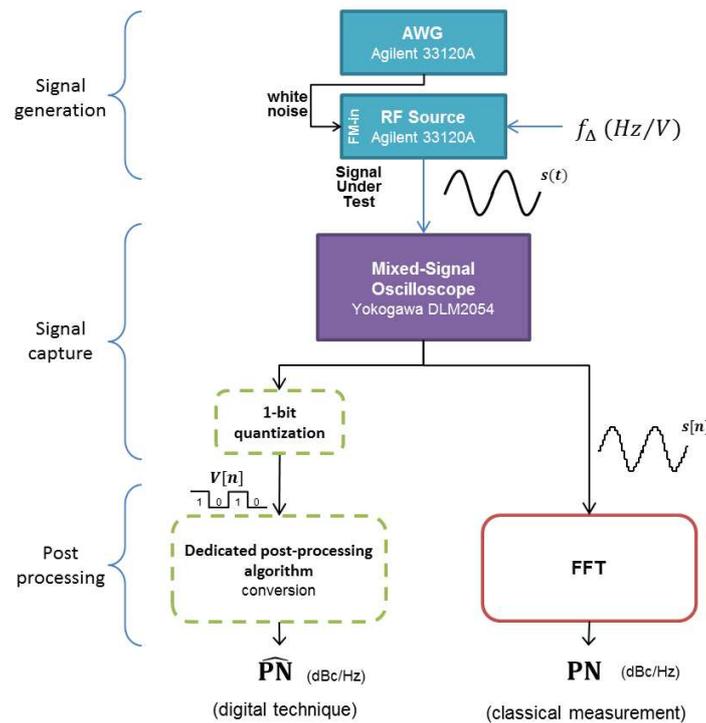


Figure 3.19 Experimental setup for hardware measurements

3.4.2 Results

Results are summarized in figure 3.20 that reports the phase noise level estimated by the digital method with respect to the phase noise level measured by the conventional method. The case study corresponds to a 3.0487MHz signal sampled at 125MHz (oversampling ratio $n = 41$), with analysis of 30,487 signal periods and phase noise evaluation at a frequency offset $f_{offset} = 75\text{kHz}$. Different levels of phase noise were injected varying the modulation depth f_{Δ} of the RF source from 1kHz up to 1MHz . A good correlation between the digital technique and the conventional method can be observed over a large range of phase noise values, from -40dBc/Hz down to -80dBc/Hz , which validates the theoretical developments and demonstrates the viability of the proposed technique. However for phase noise values below -80dBc/Hz the digital technique diverges from the conventional method. Note that this phenomenon is not observed in simulation since results correctly fit the ideal correlation line on the complete range of injected phase noise.

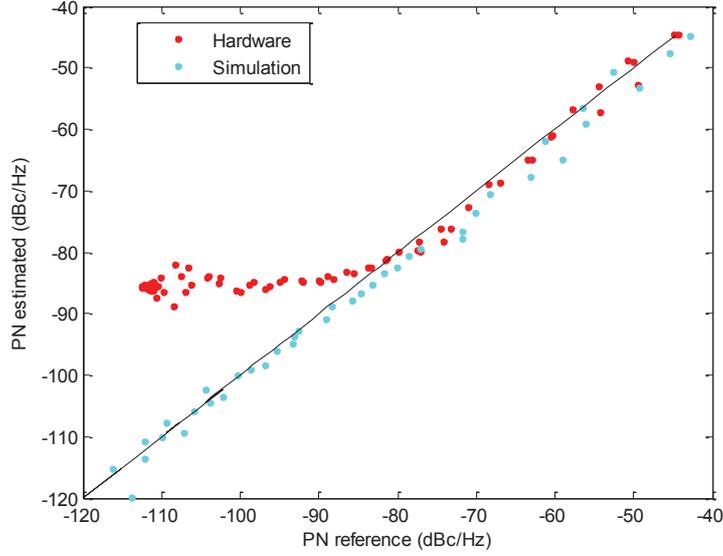


Figure 3.20 Comparison of PN measured with proposed technique and conventional method

3.5 Additional Filtering

3.5.1 Origin of Digital Technique Limitation

Additional experiments have been performed in order to understand phenomenon leading to the saturation observed in case of low phase noise level. In particular, we have applied the digital technique on a sinusoidal signal without phase noise, i.e. a signal directly delivered by an RF generator without frequency modulation. Ideally the number of samples between successive rising/falling transitions should be constant and exactly equal to the oversampling ratio, which means that the deviation count should be equal to zero whatever the analyzed period of the signal:

$$\forall m \in \{1, \dots, 2(N_{per} - 1)\}: dev(m) = 0 \quad (42)$$

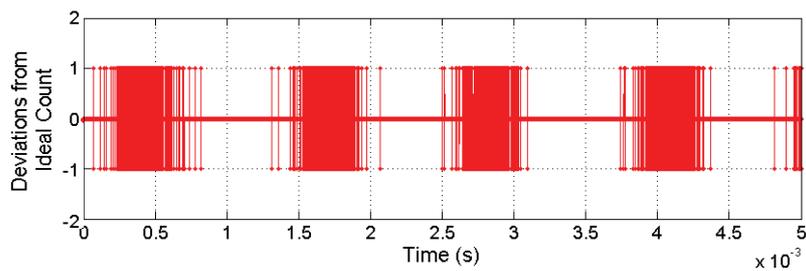


Figure 3.21 Deviation of the estimated instantaneous frequency in case of a sinusoidal signal without phase noise. Regions with multiple unwanted deviations are observed at regular time intervals.

In practice, we observe a different situation, as illustrated in figure 3.21 obtained with a 3.0487MHz signal sampled at 125MHz . There are indeed some moments where deviations remain null, but there are also moments where deviations toggle between $\{-1, 0, +1\}$. Obviously, these unwanted deviations are considered to be phase noise related, and therefore produce overestimation. These deviations are responsible of the saturation phenomenon observed in case of low phase noise level. These unwanted deviations may come from non-idealities of either the input signal or the acquisition process. Further investigations have been carried out in simulation.

First we have simulated the effect of non-idealities in the acquisition process, such as imperfect setting of the oversampling ratio due to RF generator inaccuracy (as described in section 3.2.3) and jitter in the sampling clock.

Figure 3.22 reports simulation results in case of an imperfect setting of the oversampling ratio $n = f_s/(f - \Delta f)$ with $\Delta f = 10\text{Hz}$. It can be observed that unwanted deviations are generated at regular time intervals. In this case however, only single deviations are recorded instead of a burst of deviations as observed in the hardware experiment. Note that the time interval between unwanted deviations is directly related to frequency inaccuracy with:

$$\Delta T = \frac{1}{2n \cdot \Delta f} \quad (43)$$

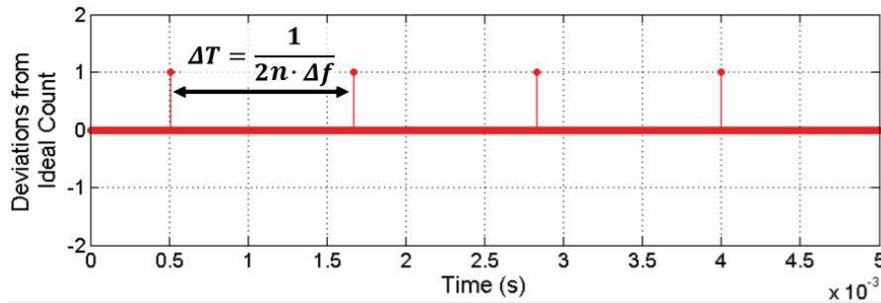


Figure 3.22 Deviation of the estimated instantaneous frequency in case of a simulated sinusoidal signal without phase noise and imperfect setting of the oversampling ratio ($\Delta f = 10\text{Hz}$). Single unwanted deviations are observed at regular time intervals

Figure 3.23 reports simulation results in case of an imperfect setting of the oversampling ratio with $\Delta f = 10\text{Hz}$ and Random Jitter (RJ) in the sampling clock with $\sigma_{RJ} = 350\text{ps}$. The behavior in this case is very similar to the one observed in the hardware experiment, i.e. regions with multiple unwanted deviations that occur at regular time intervals.

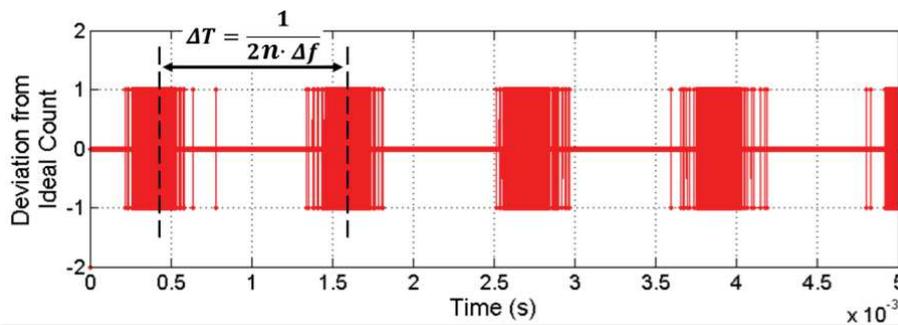


Figure 3.23 Deviation of the estimated instantaneous frequency in case of a simulated sinusoidal signal without phase noise, imperfect setting of the oversampling ratio ($\Delta f = 10\text{Hz}$) and jitter in the sampling clock ($\sigma_{RJ} = 400\text{ps}$). Regions with multiple unwanted deviations are observed at regular time intervals.

Then we have simulated the effect of non-idealities in the signal to be analyzed and in particular we have investigated the influence of voltage noise, in case of an imperfect setting of the oversampling ratio. Here again, the behavior is very similar to the one observed in the hardware experiment, as illustrated in figure 3.24 for a sine-wave signal with 0.7% amplitude noise ($\sigma_{Anoise} = 0.007 \times A$).

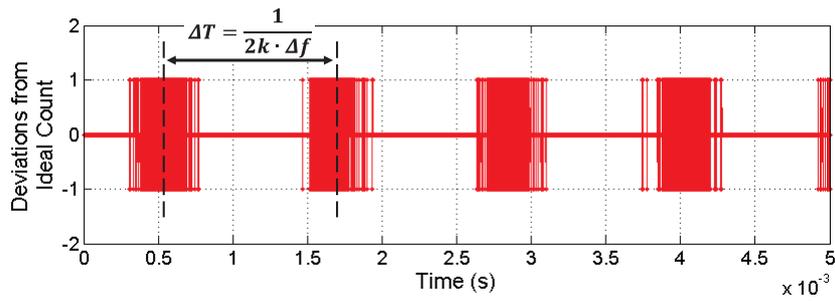


Figure 3.24 Deviation of the estimated instantaneous frequency in case of a simulated sinusoidal signal without phase noise, with voltage noise ($\sigma_{Anoise} = 0.007 \times A$) and imperfect setting of the oversampling ratio ($\Delta f = 10\text{Hz}$). Regions with multiple unwanted deviations are observed at regular time intervals

From these experiments, we can guess the origin of the limitation of the digital technique at low phase noise level. Unwanted deviations result from the combination of imperfect setting of the oversampling ratio and either jitter in the sampling clock or voltage noise in the analog signal (or both, as likely it is in practice). These parasitic deviations occur in distinct regions located at regular time intervals, and these moments actually correspond to time where crossing events fall at the close vicinity of sampling points.

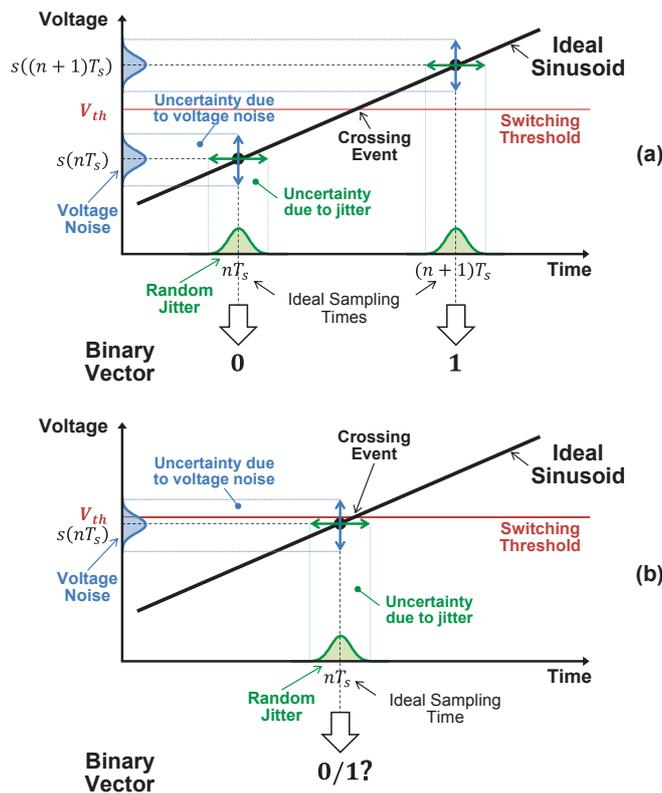


Figure 3.25 A didactic example to illustrate the influence of jitter and voltage noise on the conversion of the analog signal: (a) crossing event in the middle between sampling points and, (b) crossing event close to the sampling point

Indeed as illustrated on the didactic example of figure 3.25, when the crossing event falls in the middle between two consecutive sampling, the signal voltage at the sampling points is far enough from the switching threshold so that the signal is correctly converted into a logic "0" or logic "1". In contrast when the crossing event falls close to a sampling point, the signal voltage at the sampling point is close to the switching threshold and uncertainty in the sampling time or in the signal amplitude may result in an erroneous conversion of the signal.

3.5.2 Jitter or voltage noise?

Further investigations have been driven in order to discriminate contributions of jitter and voltage noise. In particular we have evaluated the sensitivity of the phase noise estimation to jitter and voltage noise for various signal frequencies. Basically the proposed technique relies on voltage-to-time conversion using level-crossings, i.e. the voltage information contained in the analog signal is converted into timing information in the binary signal. Voltage noise affecting the analog signal therefore produces timing noise in the binary signal. A weak voltage noise $\delta v \ll A$ will cause a timing noise δt approximately given by:

$$\delta t \approx \frac{\delta v}{2\pi A f} = \frac{\delta v}{2\pi A f_s} n \text{ with } n = \frac{f_s}{f} \quad (44)$$

This equation indicates that the timing noise in the binary signal depends not only on the voltage noise in the analog signal but also on the signal frequency. In other words it means for the same value of voltage noise δv , a signal with low frequency produce a timing noise larger than a signal with high frequency, and therefore is more susceptible to generate unwanted deviations.

On the other hand, it is clear that the timing noise induced by jitter does not depend on the signal characteristics but just on the value of the sampling clock jitter.

We can therefore distinguish between the two types of noise by studying effects of the signal frequency. In case of jitter, we expect that the number of unwanted deviations remain constant whatever the signal frequency while in case of amplitude noise, the number of unwanted deviations should increase for signal with a lower frequency.

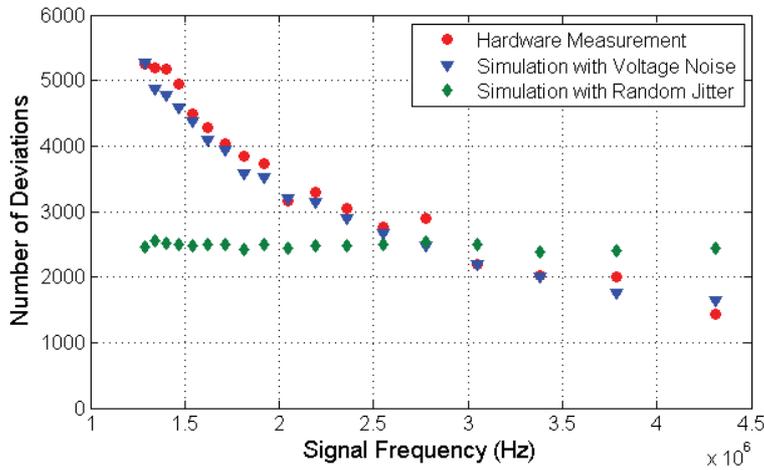


Figure 3.26 Number of deviations w.r.t. signal frequency for 1-bit acquisition at $f_s = 125MHz$ of a sinusoidal signal without phase noise.

Measures have been done, considering a sinusoidal signal without phase noise, both in simulation and experimentally. More specifically we have performed 1-bit signal acquisition with a fixed sampling rate $f_s = 125MHz$ and we have computed the total number of deviations observed on 20,000 periods, for different values of the signal frequency from 1.29MHz up to 4.31MHz. Results are summarized in figure 3.26. As expected, the number of deviations is roughly constant for the simulation of the sinusoidal signal with clock jitter, and decreases with the signal frequency in the presence of amplitude noise. Hardware measurement fits the case of voltage noise, revealing the predominance of this type of noise in the experimental setup.

3.5.3 Improved Digital Technique

In the previous section, we have established that the limitation of the proposed digital technique in case of low phase noise level comes from imperfect setting of the oversampling ratio and unwanted deviations of the instantaneous frequency generated by voltage noise in the analog signal. One approach to alleviate this limitation is to modify the post-processing algorithm in order to discriminate deviations induced by amplitude noise from deviations induced by phase noise.

The objective is therefore to develop a filter mechanism that removes unwanted deviations while preserving the information related to phase noise. For this, we have analyzed the evolution over time of deviations generated by either voltage noise or phase noise. Simulations have been performed considering the phase noise model described in chapter 2, which relies on a one-dimension Brownian motion.

As an illustration, figure 3.27.a reports an example of deviations observed in case of a simulated sinusoidal signal affected by 0.7% voltage noise and figure 3.27.b reports an example of deviations observed in case of a simulated sinusoidal signal affected by 1.4% phase noise ($\sigma/f = 0.014$). Note that results are displayed with a zoom on a small portion of the sequence to allow detailed analysis.

A number of distinctive features can be highlighted for deviations generated by amplitude noise:

- deviations only take +1, 0 or -1 values,
- a non-zero deviation is systematically followed by a zero deviation,
- consecutive non-zero deviations systematically exhibit opposite signs.

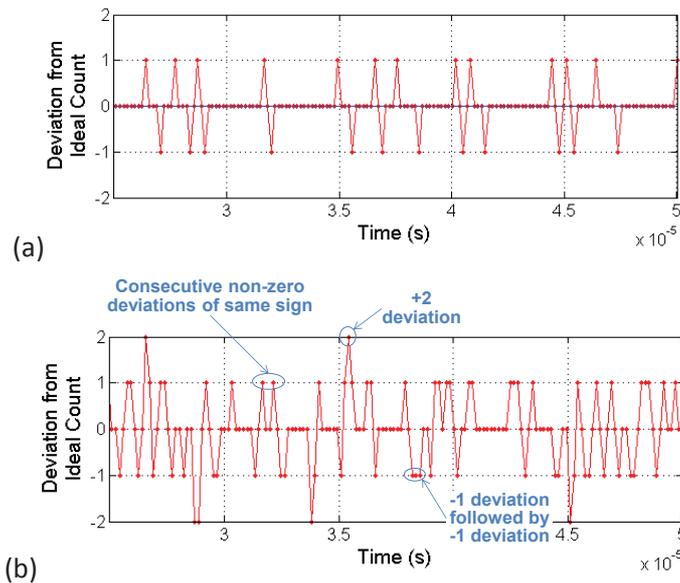


Figure 3.27 Example of deviations observed in case of (a) a simulated sinusoidal signal affected by 0.7% voltage noise, and (b) a simulated sinusoidal signal affected by 1.4% phase noise

In contrast, deviations generated by phase noise do not exhibit any of these features. The idea is therefore to develop a filter that exploits this distinctive behavior. More specifically, we have implemented an algorithmic filter that removes consecutive non-zero deviations having different signs and keep only non-zero deviations having the same sign or deviations with amplitude higher than 1. We denote $\widetilde{dev}(m)$ the deviations obtained after application of the filter.

Results are illustrated in figure 3.28 for a sequence of 10,000 periods. The efficiency of the filter to remove unwanted deviations generated by voltage noise is clearly demonstrated, while a large number of deviations induced by phase noise are still present after the filtering operation. We therefore expect that the information related to phase noise is preserved.

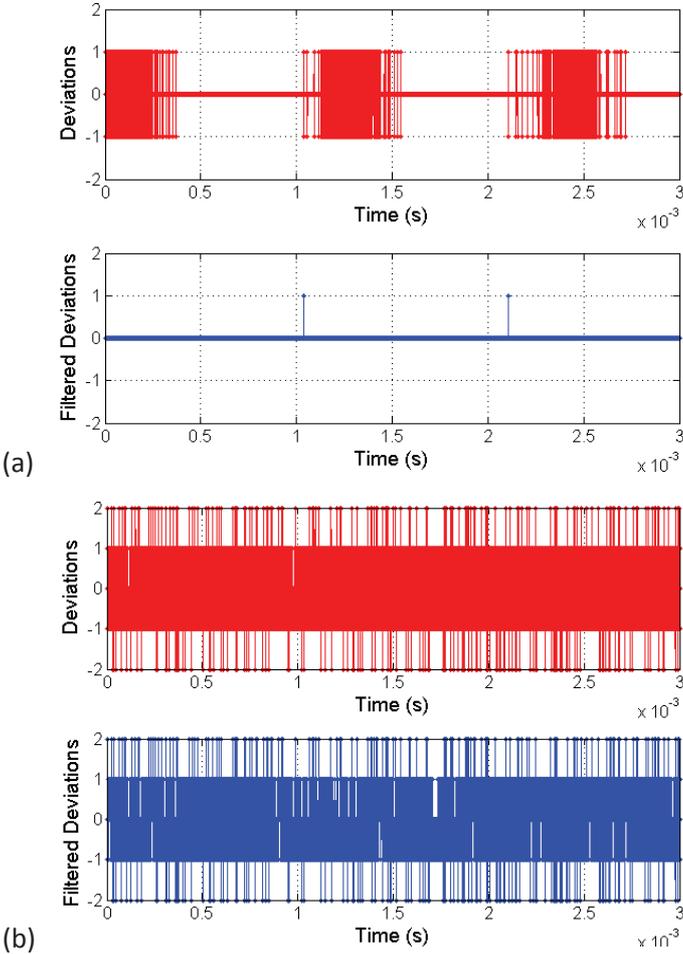


Figure 3.28 Filter operation in case of (a) a simulated sinusoidal signal affected by 0.7% voltage noise, and (b) a simulated sinusoidal signal affected by 1.4% phase noise

This assumption can actually be verified by looking at the cumulative sum of deviations, which gives an image of the Brownian motion corresponding to the input signal phase fluctuations. Figure 3.29 gives the evolution over time of this cumulative sum (zoom on a small portion of the sequence), in case of the simulated sinusoidal signal affected by phase noise, before and after application of the filter. We observe that both curves follow a similar trend, indicating that the proposed filter preserves the essential characteristics of phase noise. However it is clear that some of the deviations induced by phase noise are removed by the filter; this has to be taken into account in the post-processing algorithm.

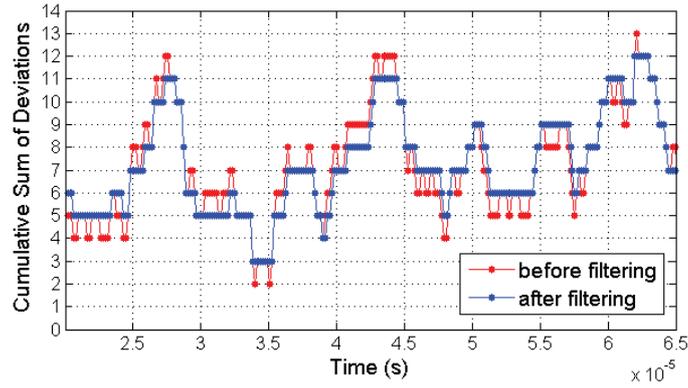


Figure 3.29 Evolution over time of the cumulative sum of deviations

Additional simulations have been carried out to evaluate filter impact. Results are summarized in figure 3.30 which reports the sum of absolute deviations after application of the filter $\sum|\widetilde{dev}(m)|$ with respect to the phase noise level σ_f/f injected in the analog signal (without any voltage noise).

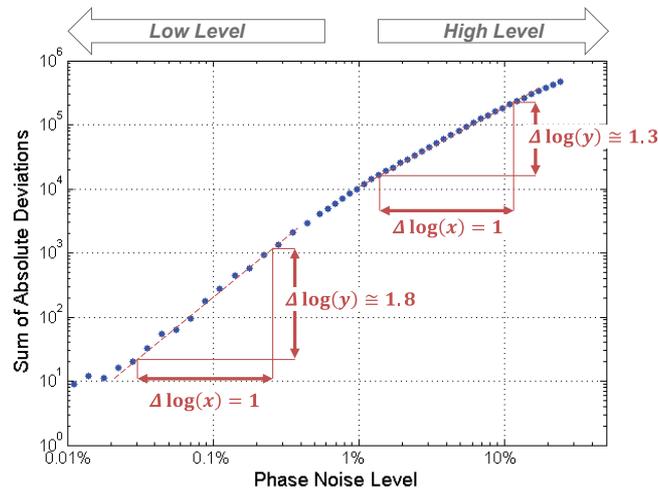


Figure 3.30 Sum of absolute deviations after application of the filter w.r.t. phase noise level injected in the analog signal (without voltage noise).

On this log-log graph, we can observe that the behavior slightly differs depending on the injected phase noise level:

- in case of low phase noise level, the relationship between the sum of absolute deviations and the phase noise level can be approximated by a straight line with a slope of about 1.8,
- in case of high phase noise level, the relationship between the sum of absolute deviations and the phase noise level can be approximated by a straight line with a slope of about 1.3.

Consequently, the idea is to propose two fitting models corresponding to low and high phase noise levels, using monomial equations of the form $\sum|\widetilde{dev}(m)| = \alpha (\sigma_f/f)^\beta$.

Note that the curve presented in figure 3.30 corresponds to a specific set of acquisition parameters $\{n, N_{per}\}$. In case of different values of these parameters, the shape of the curve remains the same but its vertical location varies. A number of simulations have been performed varying these parameters in order to analyze their influence.

From these simulations, it comes out that the sum of absolute deviations after application of the filter is obviously proportional to the number of periods. The influence of the oversampling ratio is more complex because it has a different impact in case of low and high phase noise levels: the sum of absolute deviations is directly proportional to the oversampling ratio in case of high phase noise level and proportional to the square of the oversampling ratio in case of low phase noise level.

Taking into account all these observations, we propose to use two fitting models corresponding to low and high phase noise levels:

$$\frac{\sum |\widetilde{dev}(m)|}{k^2 \cdot Nper} = \alpha_{LPN} \left(\frac{\sigma_f}{f}\right)^{\beta_{LPN}} \quad (45)$$

$$\frac{\sum |\widetilde{dev}(m)|}{k \cdot Nper} = \alpha_{HPN} \left(\frac{\sigma_f}{f}\right)^{\beta_{HPN}} \quad (46)$$

Using least-squares regression, we obtain:

$$\alpha_{LPN} = 1.07, \quad \beta_{LPN} = 1.80$$

$$\alpha_{HPN} = 3.04, \quad \beta_{HPN} = 1.28$$

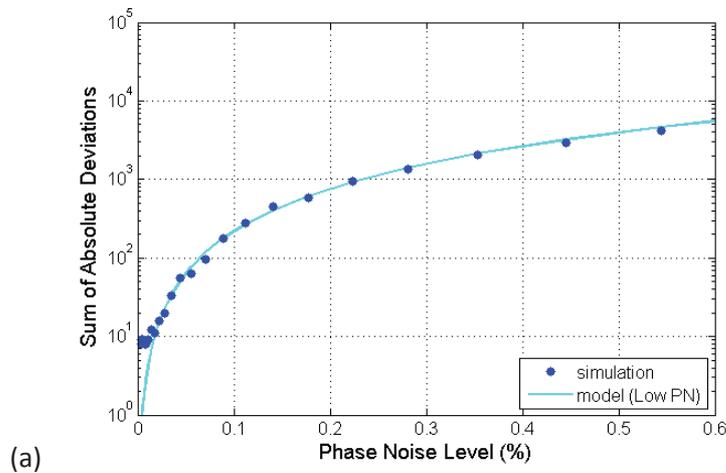
Figure 3.31 compares the sum of absolute deviations computed using the proposed models with the sum of absolute deviations observed in simulation. The case study is a 3.0487MHz signal sampled at 125MHz with analysis of 30,487 signal periods. A very good agreement is observed for both models in their validity region.

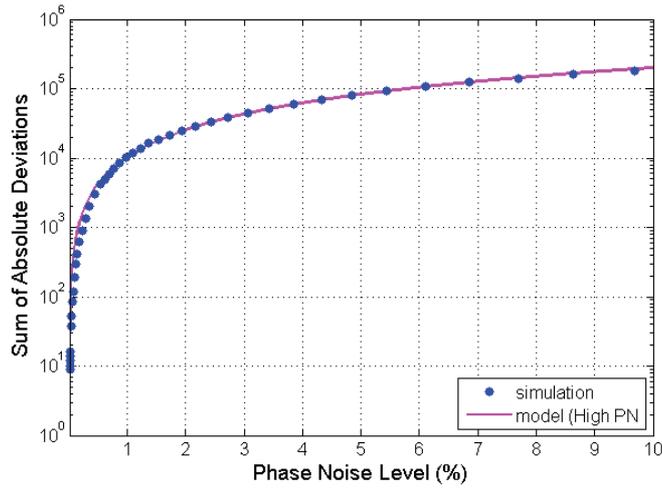
We can now use these models to address the influence of the filter on the number recorded deviations. In particular solving Eq.(12) and (13), we have:

$$\hat{\sigma}_f = f_0 \cdot \left(\frac{\sum |\widetilde{dev}(m)|}{\alpha_{LPN} \cdot k^2 \cdot Nper}\right)^{1/\beta_{LPN}} \quad \text{for } \frac{\sum |\widetilde{dev}(m)|}{k \cdot Nper} < 0.004 \quad (47)$$

$$\hat{\sigma}_f = f_0 \cdot \left(\frac{\sum |\widetilde{dev}(m)|}{\alpha_{HPN} \cdot k \cdot Nper}\right)^{1/\beta_{HPN}} \quad \text{for } \frac{\sum |\widetilde{dev}(m)|}{k \cdot Nper} > 0.004 \quad (48)$$

Note that we switched between models at $\sum |\widetilde{dev}(m)| / k \cdot Nper = 0.004$. It actually corresponds to a phase noise level σ_f/f of about 0.6%.





(b)

Figure 3.31 Validation of the fitting models that relate the sum of absolute deviations observed after application of the filter to the phase noise level present in the analog signal: (a) low phase noise level and (b) high phase noise

This novel estimator has been validated in simulation considering the case study of figure 3.20, i.e. a 3.0487MHz signal sampled at 125MHz with analysis of 30,487 signal periods. Note that a voltage noise of 0.7% has been included in the simulation. Results are summarized in figure 3.32 which reports the phase noise value estimated by the digital method (both the original technique and the new one), with respect to the phase noise value measured by the conventional method. The superior performance of the new post-processing algorithm is clearly highlighted since it does not suffer from the saturation effect observed with the original algorithm and permits correct estimation of the phase noise level on a large range of injected noise from -40dBc/Hz down to -110dBc/Hz .

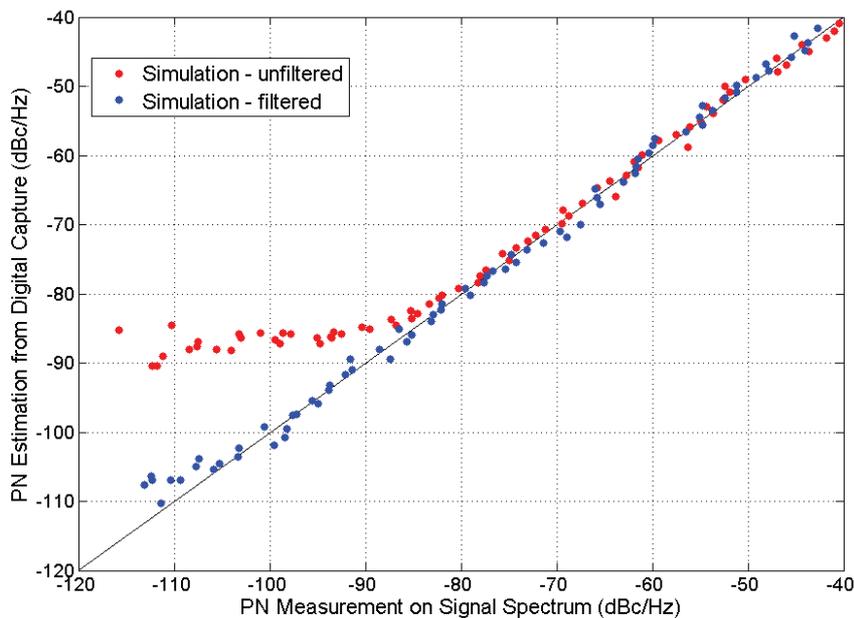


Figure 3.32 PN estimation with digital technique vs. PN measurement with conventional method, for a simulated 3.0487MHz signal sampled at 125MHz with 0.7% voltage noise

For practical implementation, the improved digital technique relies on the same setup than the one used with the original technique, i.e. 1-bit signal acquisition with a sampling frequency that is an odd multiple of the signal frequency using a standard digital tester channel. Only the post-processing

algorithm is modified. As illustrated in figure 3.33, an additional filtering step is inserted in order to remove unwanted deviations generated by voltage noise and a new estimator is used to address the influence of the filter on the deviations induced by phase noise; all other steps remains the same.

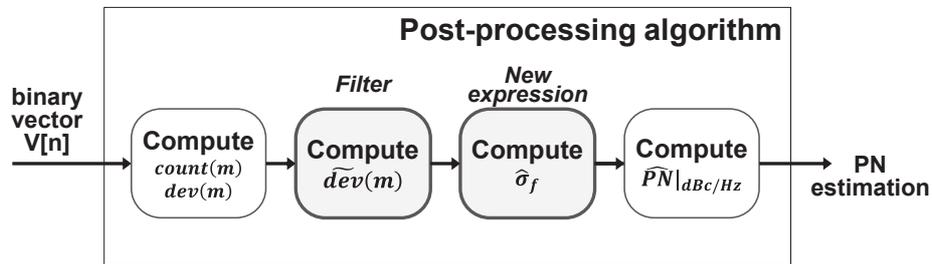


Figure 3.33 Novel post-processing algorithm for phase noise estimation from digital capture

3.6 Experimental results

In order to validate simulation results, experiments have been performed using the same setup as the one introduced in section 3.4.1.

3.6.1 Filter validation

The purpose of the first experiment was to verify the correct operation of the filter in the case of a sinusoidal signal without phase noise. To do this, we have considered the same case study than in figure 3.21, i.e. a 3.0487MHz signal delivered by the RF generator without frequency modulation and sampled at 125MHz by the mixed-signal oscilloscope.

Results are illustrated in figure 3.34 which gives the deviations observed before and after application of the filter. These results clearly demonstrate the efficiency of the filter which permits to remove the unwanted deviations generated by voltage noise.

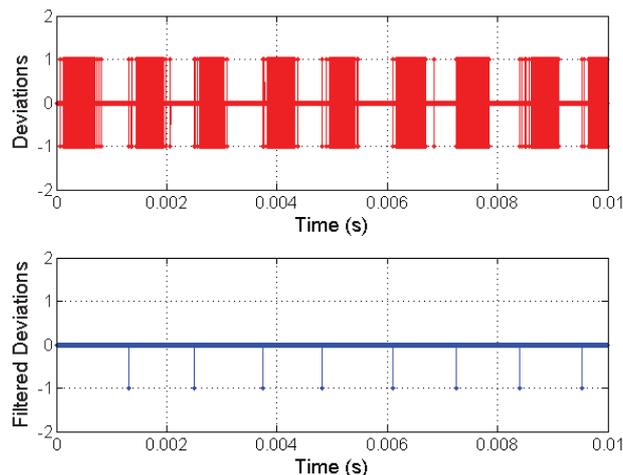


Figure 3.34 Filter operation for a 3.0487MHz signal without phase noise delivered by the RF generator without frequency modulation, and acquisition with an oversampling ratio $n = 41$.

Further experiments have been performed to verify the correct operation of the filter for different oversampling ratios. More specifically, we have analyzed signals with different frequencies while maintaining the same sampling frequency $f_s = 125\text{MHz}$ (maximum value for this equipment).

Results are summarized in figure 3.35 which reports the sum of absolute deviations after filtering according to the oversampling ratio.

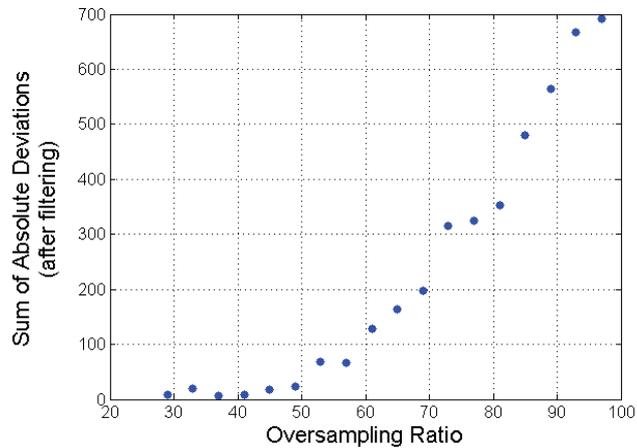


Figure 3.35 Sum of absolute deviation after filtering vs. oversampling ratio.

From the results, it appears that there is actually a limitation on the value of the oversampling ratio ensuring correct operation of the filter. Indeed the sum of absolute deviations remains very low in case of an oversampling ratio below 45, revealing efficient filtering of unwanted deviations. However, it drastically increases with the oversampling ratio for values above 45, indicating that the filter is not functional in such situations.

As an illustration, figure 3.36 shows the operation of the filter in case of a 2.77MHz signal sampled at 125MHz, which corresponds to an oversampling ratio $n = 45$. We observe that regions with unwanted deviations are larger than the ones seen in figure 3.34 and often overlap in time. Obviously, the distinctive features we have identified in case of deviations induced by voltage noise are not preserved in these overlapping regions, resulting in filter malfunction.

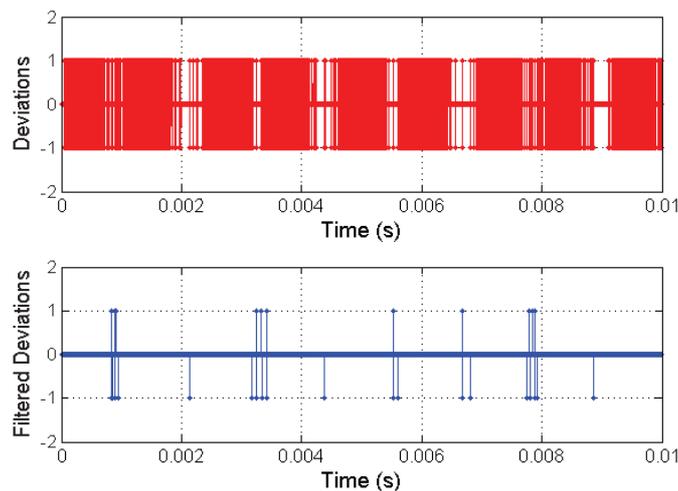


Figure 3.36 Filter operation for a 2.77MHz signal without phase noise delivered by the RF generator without frequency modulation, and acquisition with an oversampling ratio $n = 45$

A necessary condition for the correct operation of the filter is therefore that there is no overlap between regions with unwanted deviations, which imposes a limitation on the value of the oversampling ratio. This limitation does not prevent from the use of the technique, but it has an impact on the measurement variability. Indeed as demonstrated in the annex concerning the

stochastic model, the measurement variability reduces when increasing the oversampling ratio and/or the number of captured periods. The consequence of a limited oversampling ratio can therefore be compensated by increasing the number of captured periods, thus increasing test time.

Note that the maximum value of the oversampling ratio ensuring correct operation of the filter should be evaluated for each case since it depends not only on the signal characteristics (voltage noise amplitude, signal frequency) but also on the performance of the test equipment (frequency inaccuracy).

3.6.2 PN Hardware Measurements

PN measurements were performed on a sinusoidal signal affected with phase fluctuations using a previous case study (figure 3.20) based on a 3.0487MHz signal sampled at 125MHz . Different levels of phase noise were injected varying the modulation depth f_{Δ} of the RF source from 1kHz up to 1MHz . Results are summarized in figure 3.37 which reports the phase noise value estimated by the digital method (both the original technique and the new one), with respect to the phase noise value measured by the conventional method. For the sake of comparison, simulation results are also reported.

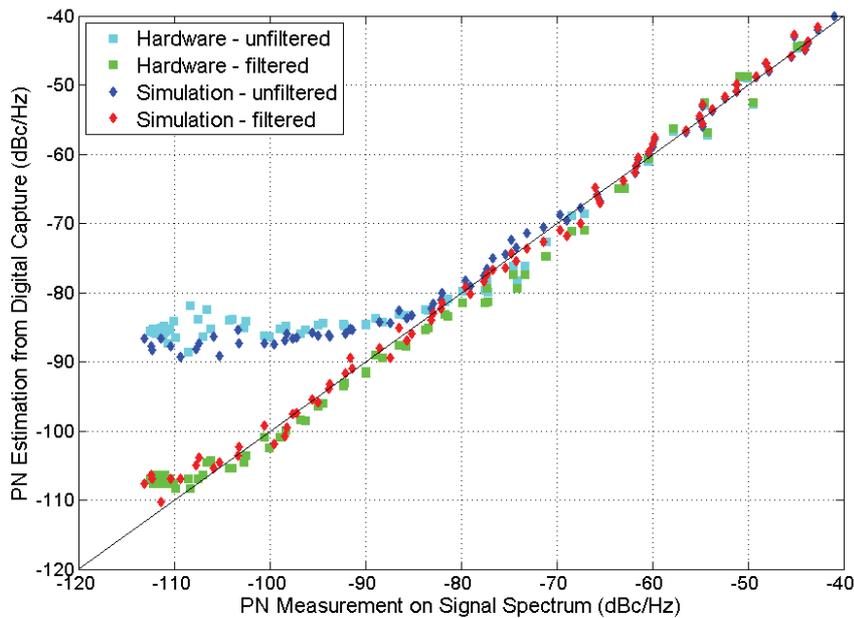


Figure 3.37 Hardware validation: comparison between PN estimation with digital technique and PN measurement with conventional method for a 3.0487MHz signal sampled at 125MHz

Experimental results present a very good agreement with simulation and confirm the superior performance of the filtered digital technique that permits to remove the saturation effect initially observed.

3.7 Conclusion

In this chapter, we have demonstrated that the analysis of the distribution of the instantaneous frequency estimated from an 1-bit acquisition of the signal under test can be used to perform phase noise evaluation, provided that the sampling frequency is an integer multiple of the signal frequency. More specifically, the mean absolute deviation of the estimated instantaneous frequency has been

identified as a robust estimator and we have established an analytical expression that permits compute the phase noise level present in the analog signal from simple operations on the binary vector. An integrated module with very small silicon area has then been defined.

Hardware experiments have demonstrated the validity of the proposed approach, but also revealed that amplitude noise is a limiting factor in case of low phase noise level. An additional filtering step has been proposed which permits to extend the valid measurement range, but imposes to keep the oversampling ratio at relatively low values.

To summarize, the technique proposed in this chapter offers a low-cost solution for measuring phase noise using only digital test resources, but suffers from two main constraints:

- the technique requires the ATE sampling frequency to be an exact integer multiple of the signal frequency, which means that the signal frequency has to be known with a good accuracy prior to the application of the digital phase noise test,
- the technique undergoes sensitivity to amplitude noise, which limits the oversampling ratio that can be used.

In this context, we have looked another way to perform phase noise measurement based on a different paradigm. As a starting point, let us observe in figure 3.38 that the density of deviations, even caused by amplitude noise, still carries information on actual phase shift of the analog signal.

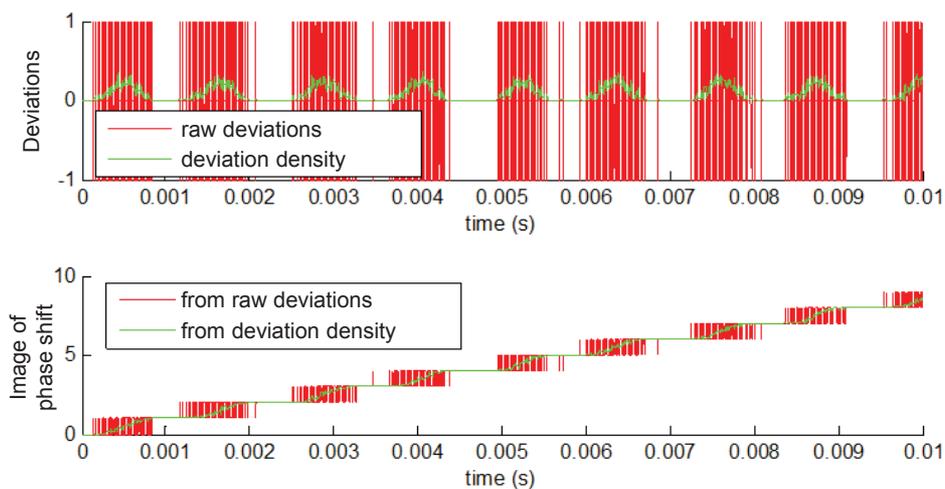


Figure 3.38 Deviations visualized with their density and image of the time-domain phase shift on hardware capture with oversampling ratio $n = 41$

Indeed, high deviation densities are representative of zero crossing times located in the vicinity of a sampling point. On the other hand, low deviation density means that the signal is far from zero crossings at sampling points. A representation of the actual phase shift can be obtained by integrating the deviation density, which basically corresponds to the cumulative sum of raw deviations. The process is illustrated in figure 3.38 and figure 3.39 in case of oversampling ratio $n = 41$ and $n = 45$ respectively. Opportunely, such reconstitution of the phase shift is robust to higher sampling ratios.

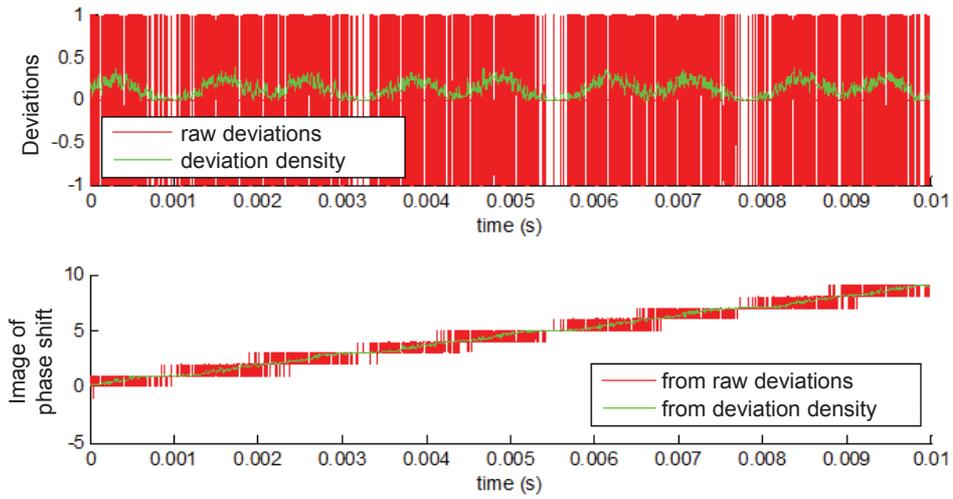


Figure 3.39 Deviations visualized with their density and image of the time-domain phase shift on hardware capture with oversampling ratio $n = 45$

This idea of phase shift reconstruction is the foundation of a second method for measuring phase noise, which is developed in next chapter.

Chapter 4. Second method: phase noise estimation based on phase fluctuation reconstruction

This chapter presents a second approach for measuring phase noise which uses the same digital resources than the previous technique but that is robust to amplitude noise. Indeed, instead of considering instantaneous frequency estimation, we consider reconstruction of time-domain phase fluctuation. The constraint on the sampling frequency is also released.

In the first part of this chapter, we introduce the theoretical background for estimation of phase noise level from the analysis of phase fluctuation, under the assumption of a Brownian motion phase noise model. Based on this principle, different implementations of the post-processing algorithms are explored in the second part. Then a stochastic model of the measurement process is described in the third part, allowing to establish guidelines for the practical setting of measurement parameters. Finally experimental results are presented, obtained both in laboratory and production contexts.

4.1 Theoretical background

Let us consider the phase noise model introduced in chapter 2, where time-domain fluctuation $\varphi(nT_s)$ is defined as a cumulative sum of random variables r_j associated with a centered normal distribution $\mathcal{N}(0, \sigma_f^2)$ (cf. eq.(3)). The standard deviation σ_f of the random variables actually corresponds to the phase noise level present in the analog signal.

This model can be related to a standard one-dimensional Brownian motion $B = \{B(t): t \geq 0\}$ with:

$$\varphi(nT_s) = \frac{2\pi\sigma_f}{\sqrt{f_s f}} \cdot B(t_n) \quad \text{with } t_n = nT_s \quad (49)$$

Such a process is characterized by three properties [69]:

1. $B(0) = 0$
2. The function $t \rightarrow B(t)$ is almost surely everywhere continuous
3. B has independent increments with $B(t + hT) - B(t) \sim \mathcal{N}(0, h)$

The last property means that the Brownian motion increments, i.e. the differences between two values separated by a given time interval, have a normal probability distribution with zero mean and variance equal to the length of the time interval. The idea is to exploit this property to recover the phase noise level σ_f from phase fluctuation $\varphi(nT_s)$.

Indeed let us define $\Delta_{k\varphi}$ as the difference between 2 values of the phase fluctuation separated from k samples:

$$\Delta_{k\varphi}(n) = \varphi((n+1)kT_s) - \varphi(nkT_s) \quad (50)$$

From equation (49) and property 3, an estimation $\hat{\sigma}_f$ of the phase noise level can be computed with:

$$\hat{\sigma}_f = \frac{\sqrt{f}}{2\pi} \cdot \frac{1}{\sqrt{kT_s}} \cdot \sigma(\Delta_{k\varphi}) \quad (51)$$

where $\sigma(\Delta_{k\varphi})$ is the standard deviation defined of phase fluctuation increments by:

$$\sigma(\Delta_{k\varphi}) = \sqrt{\frac{1}{M} \sum_{i=1}^M (\Delta_{k\varphi}(i) - \overline{\Delta_{k\varphi}})^2} \quad (52)$$

As an illustration, figure 4.1 shows an example of time-domain phase fluctuation $\varphi(nT_s)$ observed for a 1.3125MHz signal sampled at $f_s = 200\text{MHz}$ in case of $\sigma_f = 4\text{kHz}$, and the resulting histogram of $\Delta_{k\varphi}$ differences computed from values separated by 20 signal periods ($k = 20 \cdot \frac{f_s}{f}$). It can be observed that the distribution is well-centered on zero and good agreement is obtained between the estimated phase noise level $\hat{\sigma}_f = 4.02\text{kHz}$ computed from Eq. (51) and the injected one.

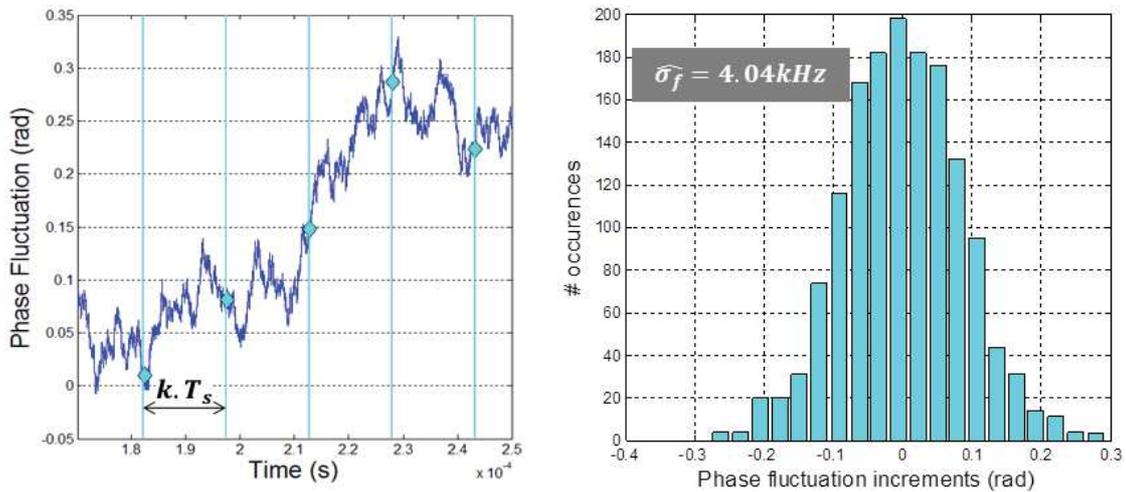


Figure 4.1 Recovering σ_f from $\varphi(nT_s)$ phase fluctuation

4.2 Algorithm

4.2.1 Phase fluctuation reconstruction

As established in the previous section, the phase noise level present in a signal $s(t) = A \cdot \sin(\Omega(t))$ can be estimated through an appropriate analysis of phase fluctuations increments. The first step of the algorithm is therefore to process the binary vector captured by a digital ATE channel in order to perform reconstruction of phase fluctuation.

Practically, we implement a discrete reconstruction exploiting the fact that the signal phase $\Omega(t)$ is incremented by $+\pi$ at each signal extremums as illustrated in figure 4.2. In particular, the transitions of the binary vector $V[n]$ are processed to compute:

$$\begin{cases} t_{extr}(i) = \frac{t_{transition}(i) + t_{transition}(i+1)}{2} \\ \hat{\Omega}(i) = i\pi \end{cases} \quad (53)$$

With this process, we obtain $2(N_{per} - 1)$ values for the extremum times and signal phase estimate, one value at each half-period $T/2$ of the signal.

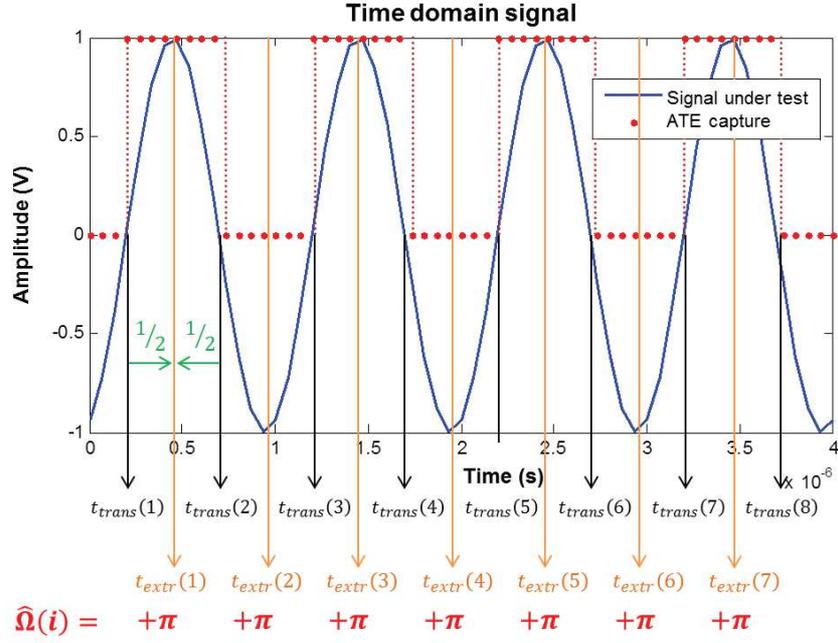


Figure 4.2 1-bit acquisition of the signal under test. The transitions of the binary vector are used to determine extremums and estimate the phase

Reconstructed phase fluctuation can then be computed by subtracting the ideal linear phase at the estimated extremum time to the signal phase estimate:

$$\hat{\phi}\left(i\frac{T}{2}\right) = \hat{\Omega}(i) - 2\pi f t_{extr}(i) \quad (54)$$

Note that we choose to perform phase fluctuation reconstruction from the date of extremums rather than from the date of transitions in order to get rid of any potential offset error in the comparator that performs the zero-crossing.

Figure 4.3 gives an illustration of the phase fluctuation reconstruction process for a $1.3125MHz$ signal sampled at $f_s = 200MHz$ in case of $\sigma_f = 4kHz$. The reconstructed phase fluctuation appears as a good image of the input one. Still, it can be observed that it is a discrete signal, not only in time with only two values per signal period, but also in amplitude. Indeed the sampling process used during 1-bit conversion introduces a quantization of the reconstructed phase fluctuation, with a quantization step directly related to the oversampling ratio $n = f_s/f$: the higher the oversampling ratio, the lower the quantization noise.

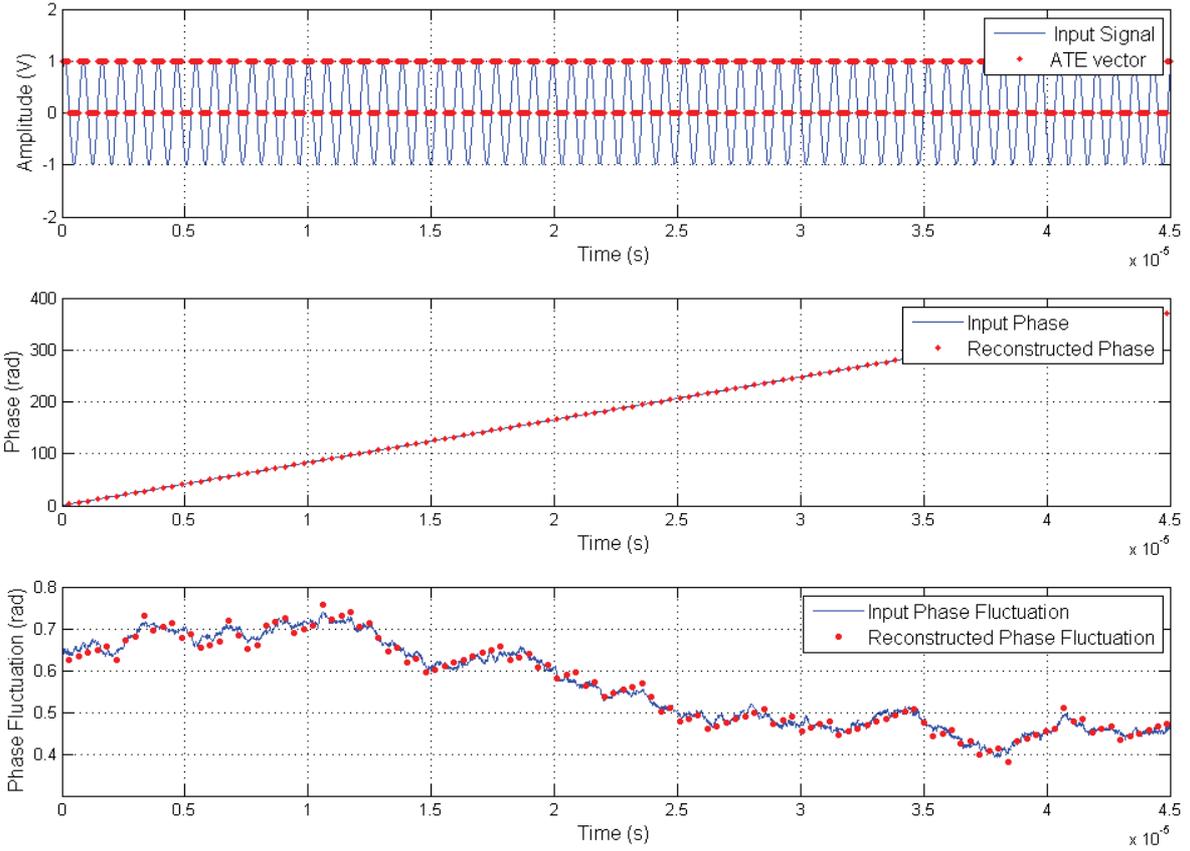


Figure 4.3 Reconstruction of phase fluctuation from 1-bit digital capture

4.2.2 Standard deviation of phase fluctuation increments

The next step of the processing algorithm is then to compute the phase noise level from phase fluctuation increments, as established by eq.(51).

Practically, we propose to perform under-sampling on the reconstructed phase fluctuation $\hat{\varphi}$ in order to collect values separated from K signal half-periods. Increments $\Delta_{K\hat{\varphi}}$ of the reconstructed phase fluctuation are then computed as the difference between successive under-samples. Finally, the estimated phase noise level is determined by:

$$\hat{\sigma}_f = \frac{f}{2\pi} \cdot \frac{1}{\sqrt{K/2}} \cdot \sigma(\Delta_{K\hat{\varphi}}) \quad (55)$$

where $\sigma(\Delta_{K\hat{\varphi}})$ is the standard deviation of the reconstructed phase fluctuation increments.

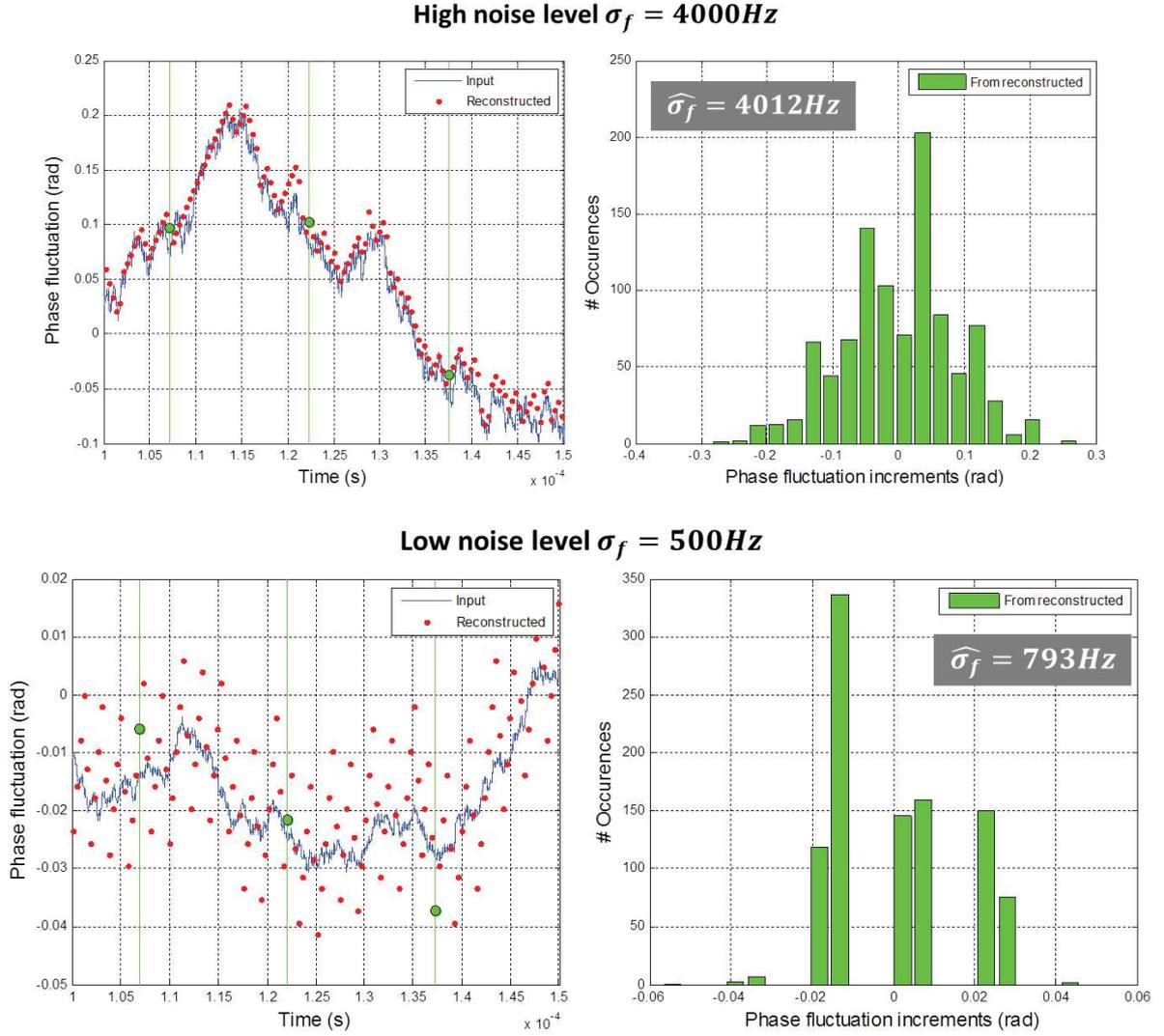


Figure 4.4 Under-sampling of phase fluctuation and distribution of $\Delta_{K\phi}$ increments for two values of injected phase noise

Figure 4.4 illustrates this process for a 1.3125MHz signal sampled at $f_s = 200\text{MHz}$ considering two different level of injected phase noise $\sigma_{fnoise} = 4\text{kHz}$ and $\sigma_{fnoise} = 500\text{Hz}$, and performing under-sampling on $K = 40$ signal half-periods. For the sake of comparison, both the input and reconstructed phase fluctuations are reported. In case of high noise level, correct estimation is achieved by processing the reconstructed phase fluctuation while in case of low noise level, the computed phase noise level is significantly overestimated. This phenomenon is related to the quantization of the reconstructed phase fluctuation. In case of low injected noise, the quantization noise does not permit to evaluate the distribution with a sufficient resolution to yield correct phase noise estimation.

To solve this issue, we propose to reduce the quantization noise by filtering the reconstructed phase fluctuation with a running average:

$$\tilde{\varphi}\left(i\frac{T_0}{2}\right) = \frac{1}{W} \cdot \sum_{j=i}^{i+W-1} \hat{\varphi}\left(j\frac{T_0}{2}\right) \quad (56)$$

where W is the window size for the running average.

This filtered reconstructed phase fluctuation is then under-sampled and increments $\Delta_{K\tilde{\varphi}}$ are computed as the difference between successive under-samples:

$$\Delta_{K\tilde{\varphi}}(i) = \tilde{\varphi}\left((i+K)\frac{T_0}{2}\right) - \tilde{\varphi}\left(i\frac{T_0}{2}\right) \quad (57)$$

Figure 4.5 shows the time-domain phase fluctuation and the histogram of increments obtained with $W = 20$ and $K = 40$, in case of low phase noise level $\sigma_f = 500\text{Hz}$. It can be observed that input and reconstructed filtered phase fluctuations are very close, yielding similar histograms. However it should be noticed that the computed phase noise level is actually under-estimated.

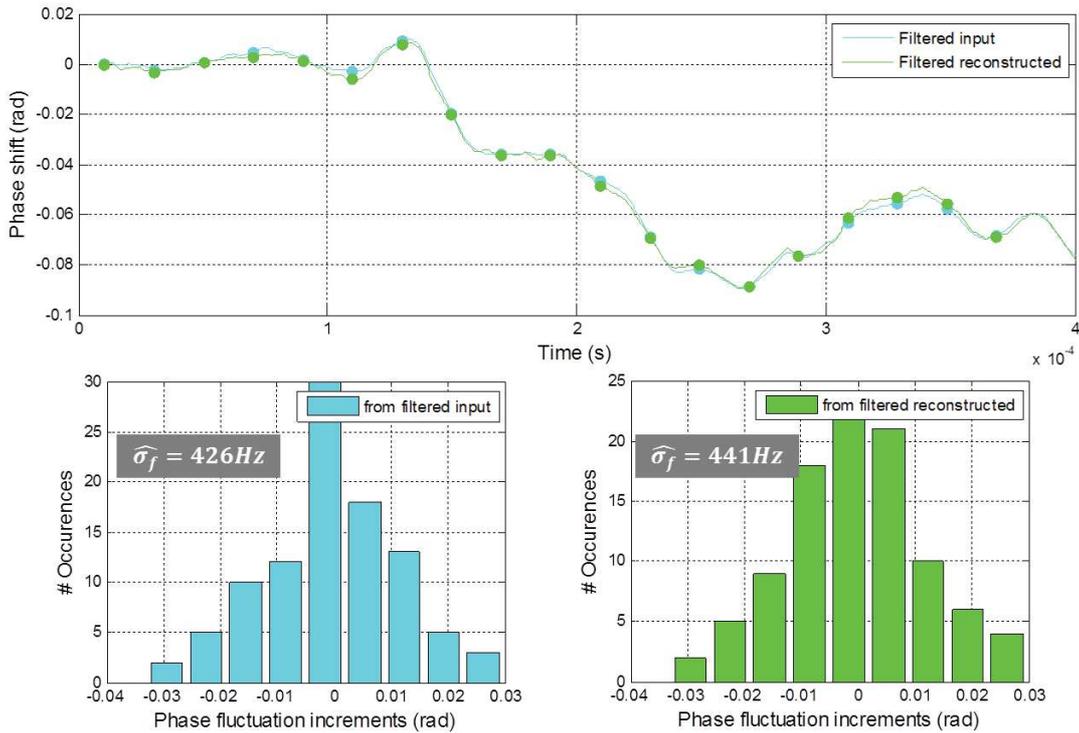


Figure 4.5 Filtering phase fluctuation with running average

This under-estimation comes from running average filtering that affects the increments distribution and introduces a bias in the computed standard deviation $\sigma(\Delta_{K\varphi})$. To illustrate this point, figure 4.6 compare the histogram of increments obtained with and without the running average filtering, considering the input phase fluctuation. Both distributions are well-centered on zero and appear very similar, however the distribution of $\Delta_{K\varphi}$ increments computed on filtered input has a smaller dispersion than the one obtained with the original input phase fluctuation.

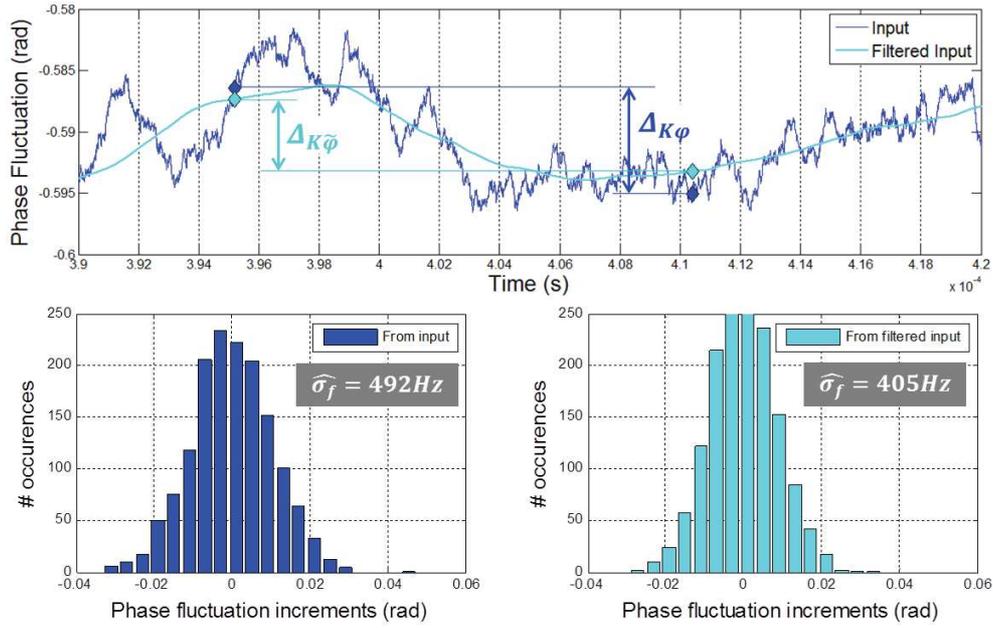


Figure 4.6 Input phase fluctuation and resulting histogram of $\Delta_{K\phi}$ increments with and without running average filtering

The bias introduced by running average filtering can actually be calculated algebraically. Indeed let us consider an academic Brownian motion:

$$x_k = \sum_{i=1}^k y_i \quad \text{with } y \sim \mathcal{N}(\mu, \sigma^2) \quad (58)$$

and the filtered Brownian motion with running average:

$$\tilde{x}_k = \frac{1}{W} \cdot \sum_{i=k-W+1}^k x_i \quad (59)$$

By definition, we know that $\sigma(x_{k+K} - x_k) = \sqrt{K} \cdot \sigma$ and we want to compute $\sigma(\tilde{x}_{k+K} - \tilde{x}_k)$. For this, the difference $\tilde{x}_{k+K} - \tilde{x}_k$ is reorganized as a linear combination of non-overlapping sums:

$$\begin{aligned} \tilde{x}_{k+K} - \tilde{x}_k = \frac{1}{W} \cdot & \left(\sum_{j=k-W+2}^k (W - (k - j + 1)) y_j - \sum_{j=k+1}^{k+K-W+1} W \cdot y_j \right. \\ & \left. + \sum_{j=k+K-W+2}^{k+K} (k + K - j + 1) y_j \right) \end{aligned} \quad (60)$$

Since each term corresponds to an independent variable, the variance $Var(\tilde{x}_{k+K} - \tilde{x}_k)$ can be computed as the sum of the variance of each term and we obtain:

$$Var(\tilde{x}_{k+K} - \tilde{x}_k) = K \cdot \sigma^2 \cdot \left(1 - \frac{1}{3K} \left(W - \frac{1}{W} \right) \right) \quad (61)$$

From this equation, we identify the bias introduced by the running average on the computed standard deviation of the filtered Brownian motion increments as:

$$bias = \sqrt{1 - \frac{1}{3K} \left(W - \frac{1}{W} \right)} \quad (62)$$

Consequently, the estimated phase noise level obtained with the filtered reconstructed phase fluctuation can be corrected using this bias. The final expression of the estimated phase noise level is given by:

$$\hat{\sigma}_f = \frac{1}{bias} \cdot \frac{f}{2\pi} \cdot \frac{1}{\sqrt{K/2}} \cdot \sigma(\Delta_{K\tilde{\varphi}}) \quad (63)$$

where $\sigma(\Delta_{K\tilde{\varphi}})$ is the standard deviation of the filtered reconstructed phase fluctuation increments.

This expression permits to evaluate the phase noise level present in the time domain signal using only information from 1-bit signal acquisition.

4.2.3 Allan deviation of phase fluctuation increments

The previous section focused on the estimation of the phase noise level σ_f by computing the standard deviation of reconstructed phase fluctuation increments $\Delta_{K\tilde{\varphi}}$. This technique can be implemented on a standard digital ATE, however it requires the storage of the entire sequence before calculations can begin. Indeed, it is necessary to compute the mean $\overline{\Delta_{K\tilde{\varphi}}}$ before beginning the computation of the standard deviation. The method therefore requires consequent memory resources and involves additional post-processing time after the acquisition of test data. In order to alleviate these drawbacks, we have investigated another solution inspired from the Allan Variance, which is a classical measure used for frequency stability analysis in clocks, oscillators and amplifiers [4]. The response of the Allan variance to different noises is a well-known topic.

The Allan variance is a measure of the variability of the average frequency of a periodic signal between two adjacent measurement intervals. It is defined by:

$$\sigma_y^2(\tau) = \frac{1}{2} \langle (\bar{y}_{i+1} - \bar{y}_i)^2 \rangle = \frac{1}{2M} \sum_{i=1}^M (\bar{y}_{i+1} - \bar{y}_i)^2 \quad (64)$$

where \bar{y}_i is the i^{th} fractional frequency average over the observation time τ .

The Allan variance is actually a special case of the M-sample variance $\sigma_y^2(M, T, \tau)$, considering 1st differences ($M = 2$) and no dead time between measurements ($T = \tau$). The interest of the Allan variance is that it removes the need of the mean computation on the entire sequence, and so it is cheap in memory usage and well adapted for on the fly computation.

In this context, our proposal is to estimate the phase noise level present in the analog signal from the Allan deviation of reconstructed phase fluctuation increments:

$$\hat{\sigma}_f = \frac{1}{bias'} \cdot \frac{f}{2\pi} \cdot \frac{1}{\sqrt{K}} \sqrt{\frac{1}{M} \sum_{i=1}^M (\Delta_{K\tilde{\varphi}}(2i+1) - \Delta_{K\tilde{\varphi}}(2i))^2} \quad (65)$$

with $M = \lfloor \frac{N_{periods}}{K} \rfloor$ the number of differentiated increments $\Delta\Delta_{K\tilde{\varphi}}(i) = (\Delta_{K\tilde{\varphi}}(2i+1) - \Delta_{K\tilde{\varphi}}(2i))$. Here we took care of not accounting twice the same increments $\Delta_{K\tilde{\varphi}}(i)$ for independency. That is why we only divide by M and not $2M$ as well as have the 2 coefficient on the index for the computation of $\Delta\Delta_{K\tilde{\varphi}}$. The calculation of the bias due to the running average changes for the Allan deviation. Still, the calculation process is similar as the one presented in the previous section for standard deviation.

Considering the Brownian motion x_k defined in eq.(58), we know that the mean $\overline{(x_{k+2K} - x_{k+K}) - (x_{k+K} - x_k)} = 0$ since we deal with a Brownian motion hypothesis. This is the also case for the filtered Brownian motion: $\overline{(\tilde{x}_{k+2K} - \tilde{x}_{k+K}) - (\tilde{x}_{k+K} - \tilde{x}_k)} = 0$.

So:

$$\begin{aligned}
AllanVar(\tilde{x}_k) &= \frac{1}{2M} \sum_{i=1}^M ((\tilde{x}_{k+2K} - \tilde{x}_{k+K}) - (\tilde{x}_{k+K} - \tilde{x}_k))^2 \\
&= \frac{1}{2M} \sum_{i=1}^M \left((\tilde{x}_{k+2K} - \tilde{x}_{k+K}) - (\tilde{x}_{k+K} - \tilde{x}_k) \right. \\
&\quad \left. - \overline{(\tilde{x}_{k+2K} - \tilde{x}_{k+K}) - (\tilde{x}_{k+K} - \tilde{x}_k)} \right)^2 \\
&= \frac{1}{2} Var((\tilde{x}_{k+2K} - \tilde{x}_k) - (\tilde{x}_{k+K} - \tilde{x}_k))
\end{aligned} \tag{66}$$

We can reorganize this as non-overlapping sums:

$$\begin{aligned}
&(\tilde{x}_{k+2K} - \tilde{x}_{k+K}) - (\tilde{x}_{k+K} - \tilde{x}_k) \\
&= \frac{1}{W} \left(\sum_{j=k-W+2}^k ((k-j+1) - W)y_j - \sum_{j=k+1}^{k+K-W+1} W y_j \right. \\
&\quad + \sum_{j=k+K-W+2}^{k+K} (W - 2(k+K-j+1))y_j \\
&\quad \left. + \sum_{j=k+K+1}^{k+2K-W+1} W y_j + \sum_{j=k+2K-W+2}^{k+2K} (k+2K-j+1)y_j \right)
\end{aligned} \tag{67}$$

And compute the variance of each term so that:

$$Var((\tilde{x}_{k+2K} - \tilde{x}_{k+K}) - (\tilde{x}_{k+K} - \tilde{x}_k)) = 2K\sigma^2 \left(1 - \frac{1}{2K} \left(W - \frac{1}{W} \right) \right) \tag{68}$$

The bias is then:

$$bias' = \sqrt{\left(1 - \frac{1}{2K} \left(W - \frac{1}{W} \right) \right)} \tag{69}$$

Note that with this expression, we can choose to set the parameters K and W at different values. The parameter K is used in the under-sampling process and determines the time interval between successive phase fluctuation measurements. This parameter should be set according to the frequency offset f_{offset} at which phase noise has to be evaluated. More precisely, the Allan deviation is computed using phase fluctuation increments separated by $2K$ signal half-periods, which corresponds to an observation time $\tau = 2K \cdot \frac{T}{2} = KT$. In order to evaluate phase noise level at a frequency offset f_{offset} , the observation time should be set at $\tau = 1/f_{offset}$, which means that the parameter K should be set at:

$$K = \frac{f}{f_{offset}} \tag{70}$$

The parameter W corresponds to the size of the window for running average filtering and controls the low-pass cut-off frequency: the larger the window size, the lower the cut-off frequency. This parameter should be set as high as possible for efficient filtering of the quantization noise. However as it can be observed from eq.(71), $\frac{1}{bias'}$ diverges for $2K = \left(W - \frac{1}{W} \right) \approx W$. The choice of $W = K$

actually appears as a good compromise between this diverging zone and sensitivity of quantization noise.

As an illustration, figure 4.7 shows the result of phase noise estimation using eq.(65) according to the size of the running average window, in case of a simulated signal with injected phase noise $\sigma_f = 500\text{Hz}$. This graph clearly shows that the phase noise level is overestimated when W decreases towards 0 due to the quantization noise that isn't filtered, and when W approaches $2K$ due to the divergence of the $1/\text{bias}$ ' correcting factor. Correct estimation is obtained with $W = K$.

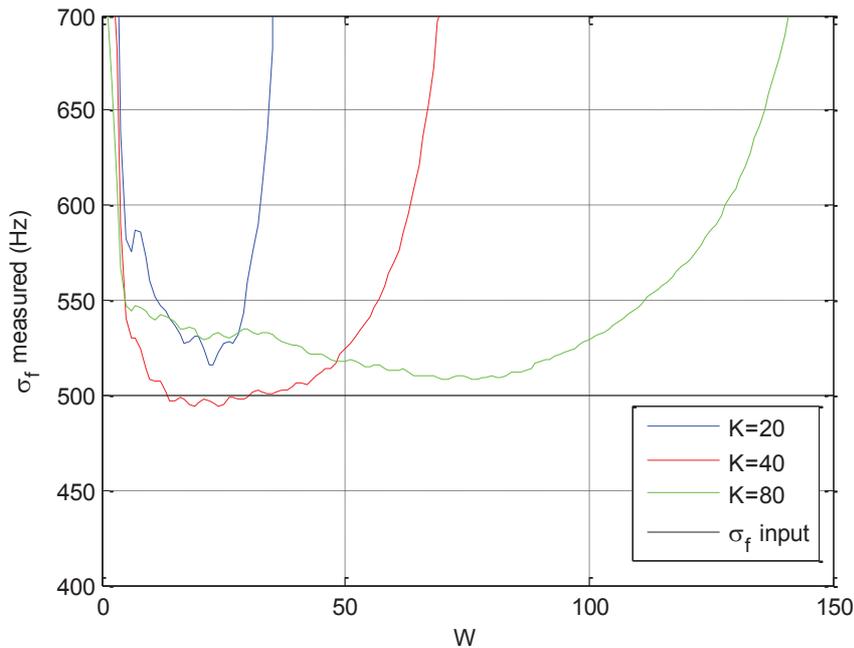


Figure 4.7 Phase noise estimation $\hat{\sigma}_f$ based on Allan deviation with bias correction vs. running average window size W , for 3 different values of K (injected phase noise level $\sigma_f = 500\text{Hz}$)

To summarize, the block diagram of the processing algorithm for phase noise estimation based on Allan deviation of phase fluctuation increments is given in figure 4.8. Note that correct operation of this algorithm does not require an integer oversampling ratio. Moreover, the algorithm is expected to be more robust to amplitude noise than the previous solution based on the mean absolute deviation of the instantaneous frequency.

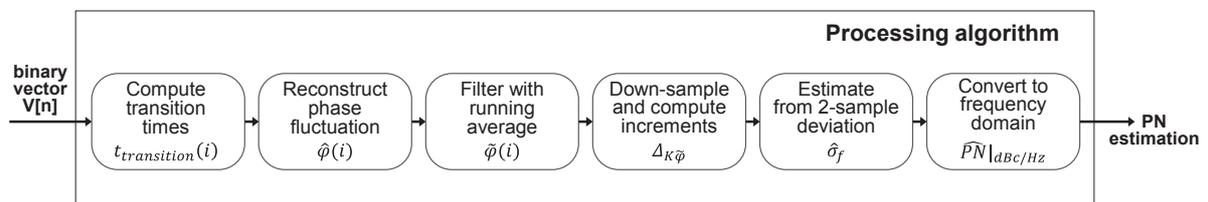


Figure 4.8 Block diagram of processing algorithm for phase noise estimation based on Allan deviation of phase fluctuation increments

Another remark has to be made regarding the running average as a quantization noise filter. One could choose other filters. The advantage of the running average is that the bias is easily algebraically calculable.

Finally it should be mentioned that the processing algorithm can also be modified to take into the account all the samples that have been discarded during the under sampling. In this case, estimation is based on an overlapping Allan deviation and offers less dispersion thus requiring fewer periods. However one has to be careful that, after filtering through the running average, the under samples are not independent when not sufficiently spaced in time.

4.2.4 Other approaches

Once the signal phase $\hat{\Omega}(i)$ is recovered, other approaches different from standard deviation or Allan deviation on phase fluctuation increments can also be envisaged for measuring phase noise. For instance, the signal phase estimate can be used to compute a virtual signal $\hat{s}(i)$, corresponding to an image of the noisy analog sine-wave:

$$\hat{s}(i) = \sin(\hat{\Omega}(i)) \quad (71)$$

The idea is then to apply FFT computation on this virtual signal and perform the classical phase noise measurement on the computed spectrum. However at this stage, we have a discrete estimation of the signal phase with only two samples per signal period, which is not sufficient to compute accurate spectrum. In addition, samples are non-uniformly distributed in times, introducing distortion in the computed spectrum. To palliate these limitations, we propose to compute a continuous-time estimation of the signal phase $\hat{\Omega}(t)$ based on linear interpolation:

$$\hat{\Omega}(t) = \hat{\Omega}(i) + \frac{\hat{\Omega}(i+1) - \hat{\Omega}(i)}{T(i+1) - T(i)}(t - T(i)) \quad (72)$$

The continuous-time virtual signal is then obtained by applying a sine function to the interpolated phase:

$$\hat{s}(t) = \sin(\hat{\Omega}(t)) \quad (73)$$

Finally, this continuous-time signal can be sampled at any desired frequency rate in order to compute the signal power spectrum through FFT calculation.

This was experimentally tested with the setup described in section 3.4.1. Figure 4.9 shows the signal spectrum resulting from a full capture with the mixed-signal oscilloscope as well as the signal spectrum resulting from 1-bit capture. A very good agreement is observed between both spectrums. PN measurements were carried out on these spectrums, and repeated a number of times in order to compare both techniques in terms of variability. Results demonstrated that accurate measurements can be achieved with the proposed technique, with a mean estimation error lower than 0.7dBc/Hz and a standard error in the same range than the conventional industrial technique.

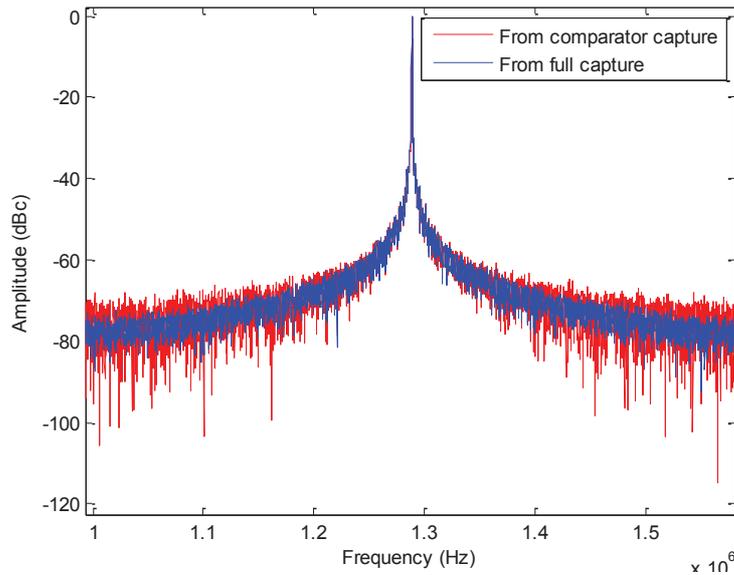


Figure 4.9 Signal spectrums computed with FFT on full signal capture and on virtual signal recovered from 1-bit capture

One could also choose to work on the reconstructed phase fluctuation $\hat{\varphi}(i)$. In particular, the phase noise spectrum may be obtained by directly applying a FFT on the reconstructed phase fluctuation. This approach was applied on an industrial device, i.e. a JN5168 wireless microcontroller fabricated by NXP. The 1.3125MHz sinusoidal signal delivered at the transceiver output was captured using both a PS1600 digital channel (2,688 periods captured at 32MHz sampling frequency) and a MBAV8 analog channel (52,500 periods captured at 200MHz sampling frequency) of Advantest 93K ATE. The digital capture was processed to reconstruct phase fluctuation and FFT applied on reconstructed data. The analog capture was processed through FFT and phase noise measurements performed on the resulting spectrum for different values of the measurement frequency offset. Results are summarized in figure 4.10 that reports the phase noise spectrum obtained by both methods. Here again a very good agreement is observed between both spectrums.

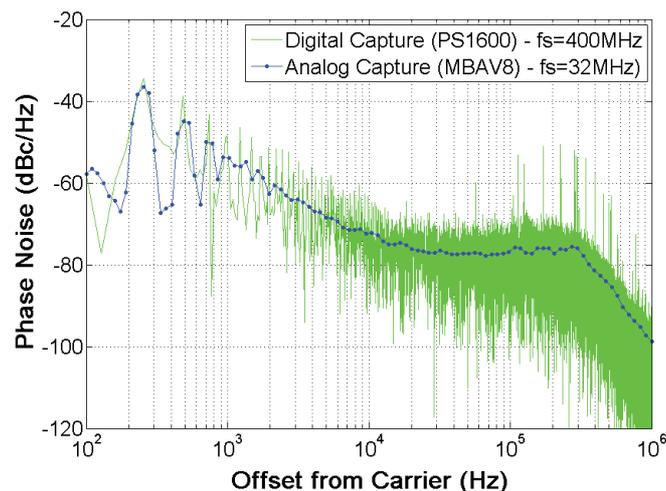


Figure 4.10 Fourier transform of the full signal and the reconstructed signal from phase estimation

These two methods were not deeper investigated because they necessitate the storage of the entire sequence and powerful processing resources for FFT calculation.

4.3 Performance of the measurement process based on Allan deviation

In this section, we analyze the performance of the measurement process based on Allan deviation. In the first part, we derive a model that allows the evaluation of the minimum phase noise level that can be measured according to given signal and test conditions. In the second part, we evaluate the measurement variability through a stochastic model of the measurement process.

4.3.1 Minimum phase noise level measurable

Simulations were performed to validate the phase noise estimator based on the Allan variance defined by eq.(65). As an illustration, figure 4.11 reports the estimated phase noise level $\hat{\sigma}_f$ with respect to the injected phase noise level σ_f , for $f = 1.3125\text{MHz}$, $f_s = 200\text{MHz}$, $N_{periods} = 12500$ and $K = W = 40$. On this log-log plot, we observe a good agreement on a large range of injected phase noise from 100Hz to 100kHz . However we observe that there is a measurement floor, indicating that the process is not able to measure too small noises.

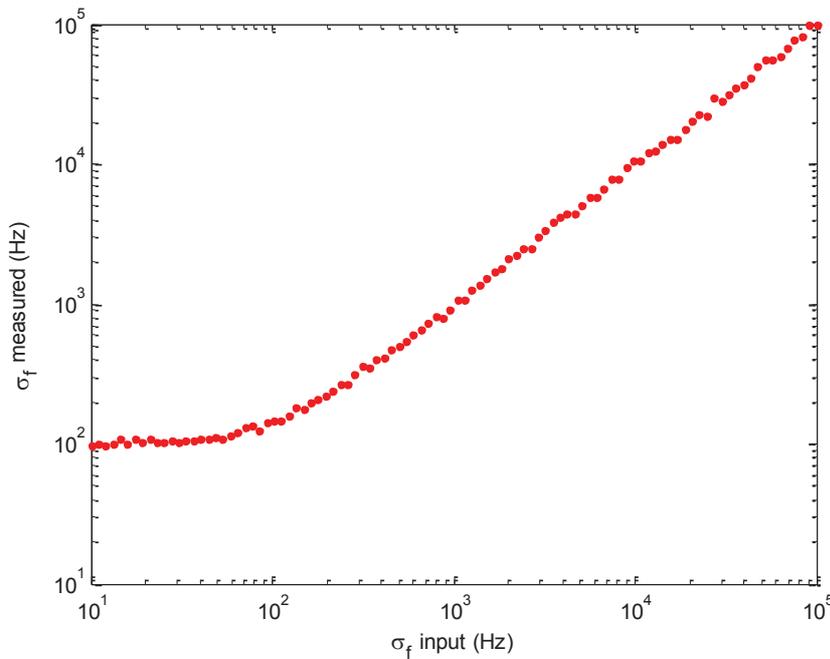


Figure 4.11 Estimated phase noise level $\hat{\sigma}_f$ vs. injected phase noise level σ_f

It is important to characterize σ_{min} the minimum phase noise level that can be measured and how it depends on the signal and test conditions. This has been done through a series of simulations and fit over different parameters, considering a sine-wave without phase noise ($\sigma_f = 0\text{Hz}$). The parameters studied where:

- The sampling frequency f_s
- The signal frequency f
- The under-sampling ratio K
- The running average window W
- The number of periods $N_{periods}$

- The relative amplitude noise B
- And jitter in the sampling clock J

Let us start with the sampling frequency.

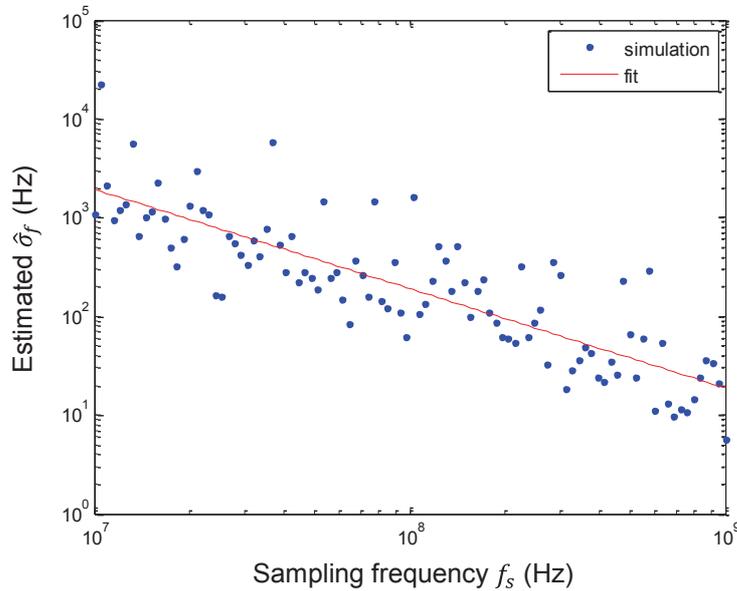


Figure 4.12 Estimated phase noise level $\hat{\sigma}_f$ vs. sampling frequency f_s on a sine-wave without phase noise ($\sigma_f = 0\text{Hz}$)

Figure 4.12 shows the results of the simulation with different f_s as well as a fit of these results. We have a fit of the form:

$$\sigma_{min} = B_0 f_s^{B_1} \quad (74)$$

Here $B_1 = 1$.

We can do that with the other parameters: f, W, K . And we obtain:

$$\sigma_{min} = 0.1 \frac{f^2}{K^{0.5} f_s W^{0.4}} \quad (75)$$

We checked that σ_{min} does not depend on the number of periods.

The fit identified for amplitude noise is different. As illustrate in figure 4.13, σ_{min} measured is of the form:

$$\sigma_{min} = B_0 (1 + B_1 \sqrt{\text{Amplitude Noise}}) \quad (76)$$

With $B_1 = 80$.

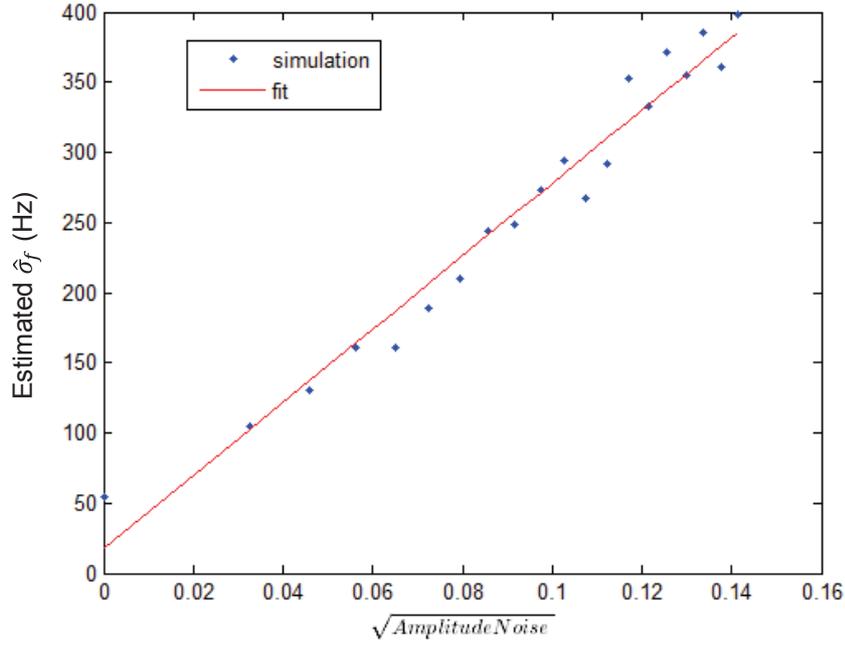


Figure 4.13 Estimated phase noise level $\hat{\sigma}_f$ vs. $\sqrt{\text{Amplitude Noise}}$ on a sine-wave without phase noise ($\sigma_f = 0\text{Hz}$)

The jitter, in seconds unit, can be considered as amplitude noise as well, with the slope coefficient at the zero crossing of $2\pi f$.

$$\sigma_{min} = B_0(1 + B_1\sqrt{jitter * 2\pi f}) \quad (77)$$

Finally combining together all these influences, the expression of the minimum phase noise level that can be measured under given conditions of signal and test parameters is:

$$\sigma_{min} = 0.1 \frac{f^2}{K^{0.5} f_s W^{0.4}} (1 + 80\sqrt{jitter * 2\pi f + B}) \quad (78)$$

where B is the relative amplitude noise (no units).

This expression has been validated through simulation under different conditions of signal and test parameters. These simulations have revealed that the sampling ratio also has also an influence on σ_{min} . Figure 4.14 illustrates one case, where only the sampling ratio is varied all other parameters being set at fixed values. The large majority of points are in accordance with the minimum measurable phase noise level computed with eq.(78). Results only diverge from σ_{min} when the sampling ratio n is close to even integers. Indeed in this case, the quantization noise is at its maximum and can hardly be reduced. Obviously this situation should be avoided, for instance by imposing $|(n + 1)\%2) - 1| > 0.1$.

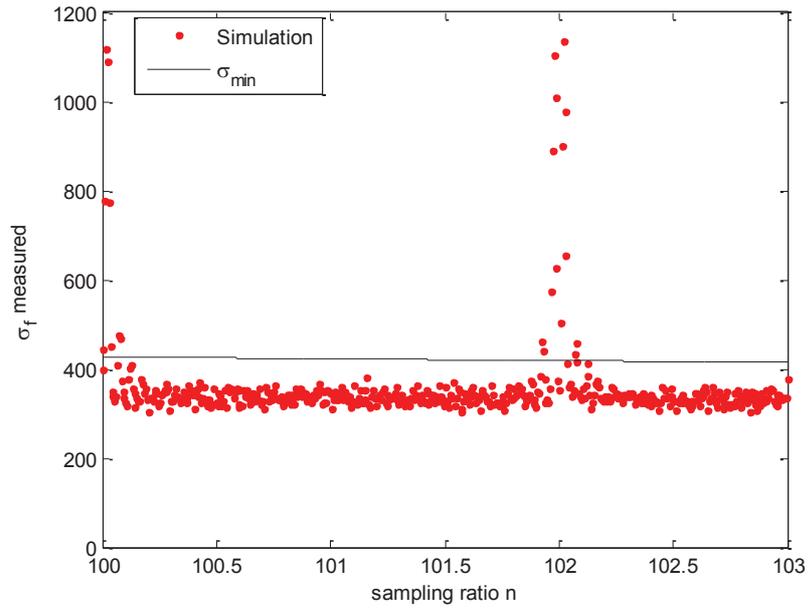


Figure 4.14 Estimated phase noise level $\hat{\sigma}_f$ vs. sampling ratio n on a sine-wave without phase noise ($\sigma_f = 0\text{Hz}$)

The robustness of the σ_{min} formula was evaluated by running a number of simulations with random parameter values. Even though K and W are different parameters, we here fix $K = W$ as advised in section 4.2.3. Results are summarized in figure 4.15 that plots the ratio between the phase noise level evaluated in simulation using the Allan deviation and the computed value of σ_{min} for the different random runs. Good agreement is observed, and the only values that significantly diverge from the computed σ_{min} correspond to a sampling ratio close to an even integer.

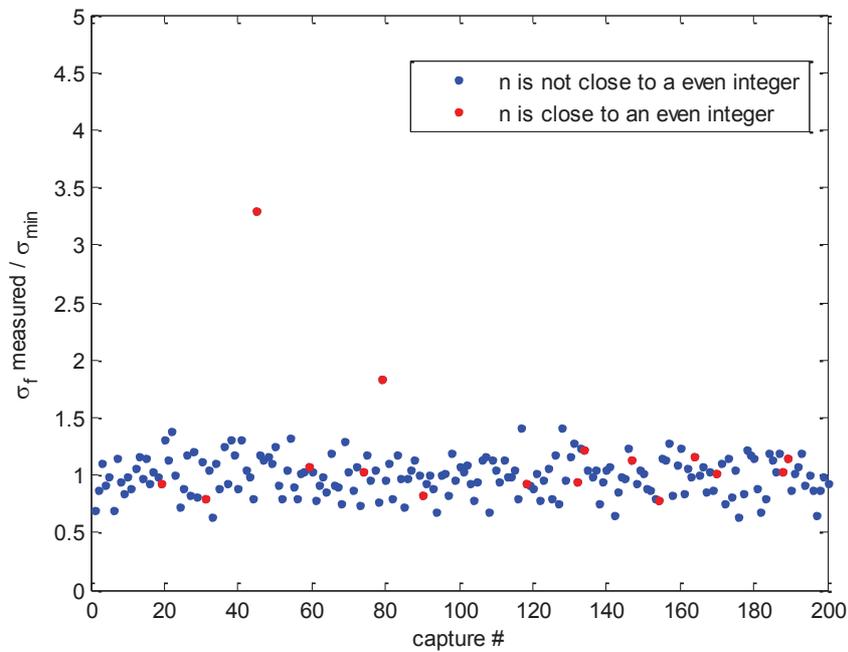


Figure 4.15 Random simulation runs of a sine-wave without phase noise

These results fully validate the analytical expression of σ_{min} established in eq.(78). This expression is very useful because it permits to verify whether a new product can be tested on a standard digital ATE. Indeed each application is characterized by a given signal frequency and a typical phase noise level to be measured at a given offset frequency. The signal to be analyzed may include amplitude noise. On the other side, the test equipment is characterized by its maximum operating frequency and sampling jitter. Knowing all these parameters, one can verify that the minimum phase noise that can be measured is smaller than the typical phase noise level for the product under consideration. If not, it also permits to specify what would be the required ATE performances to allow such measurement.

4.3.2 Measurement variability

A stochastic model has been developed to describe the behavior of the measurement process based on the Allan deviation in presence of phase noise modeled as a Brownian motion. The main idea behind the stochastic model development, is the ability to predict measurement variability according to given signal and test conditions. This study can be carried out with the help of the χ^2 (pronounced chi-squared) probability density function. This function describes the probability density of a sum of squared Gaussian variable.

Let us define the sum of squares Q of the Gaussian variable X_i described by $\mathcal{N}(\mu_i, \sigma_i)$. or the normalized Gaussian variables Y_i

$$Q = \sum_{i=1}^k Y_i^2 = \sum_{i=1}^k \left(\frac{X_i - \mu_i}{\sigma_i} \right)^2 \quad (79)$$

Q is then described by the density probability function :

$$f_{\chi^2}(x, k) = \frac{x^{\frac{k}{2}-1}}{\Gamma\left(\frac{k}{2}\right) 2^{\frac{k}{2}}} \exp\left(-\frac{x}{2}\right) \quad (80)$$

Where Γ is the gamma function. How does that relate to our measurement? Well, the square of the measurement we proposed is:

$$\hat{\sigma}_f^2 = \left(\frac{1}{bias} \frac{f}{2\pi} \right)^2 \frac{1}{M} \sum_{i=0}^M \left(\frac{\Delta\Delta_{K\tilde{\varphi}}(i)}{\sqrt{K}} \right)^2 \quad (81)$$

with $M = \left\lfloor \frac{N_{periods}}{K} \right\rfloor$ the number of differentiated increments $\Delta\Delta_{K\tilde{\varphi}}(i) = (\tilde{\varphi}(K(2i+2)) - \tilde{\varphi}(K(2i+1))) - (\tilde{\varphi}(K(2i+1)) - \tilde{\varphi}(K(2i)))$ the double difference of the phase fluctuation samples spaced of K .

Since we are in the hypothesis of a Brownian motion we can assume that $\overline{\Delta\Delta_{K\tilde{\varphi}}} = \mu = 0$. We recognize $X_i = \frac{f}{2\pi bias} \frac{\Delta\Delta_{K\tilde{\varphi}}(i)}{\sqrt{MK}}$ following $\mathcal{N}(0, \frac{\sigma_f}{\sqrt{M}})$. This would permit the change of variable from:

$$\hat{\sigma}_f^2 = \sum_{i=0}^M \left(\frac{f}{2\pi bias} \frac{\Delta\Delta_{K\tilde{\varphi}}(i)}{\sqrt{MK}} \right)^2 \quad (82)$$

To

$$Q = \sum_{i=1}^M \left(\sqrt{M} \frac{X_i}{\sigma_f} \right)^2 \quad (83)$$

Since Q follows the probability density $f_{\chi^2}(x, M)$, we conclude that $\hat{\sigma}_f^2$ follows the probability density $f_{\chi^2}\left(M \frac{x}{\sigma_f^2}, M\right)$ with x in HZ^2 .

The expected value of Q is M , so the expected value of $\hat{\sigma}_f^2$ is:

$$E[\hat{\sigma}_f^2] = \sigma_f^2 \quad (84)$$

And its variance:

$$Var[\hat{\sigma}_f^2] = \frac{2\sigma_f^4}{M} \quad (85)$$

However we are not interested in $\hat{\sigma}_f^2$ but rather in $\hat{\sigma}_f = \sqrt{\hat{\sigma}_f^2}$. $\hat{\sigma}_f$ will thus follow a χ probability law. Both cumulative distribution function of each law are related by :

$$F_{\chi}(x) = F_{\chi^2}(x^2) \quad (86)$$

Since their density probability functions are defined as

$$\frac{dF_{\chi}(x)}{dx} = f_{\chi}(x) \text{ and } \frac{dF_{\chi^2}(x^2)}{dx^2} = f_{\chi^2}(x^2) \quad (87)$$

We have the relation between the probability density functions:

$$f_{\chi}(x) = \frac{dF_{\chi}(x)}{dx} = \frac{dF_{\chi^2}(x^2)}{dx} = \frac{dx^2}{dx} \frac{dF_{\chi^2}(x^2)}{dx^2} = 2x f_{\chi^2}(x^2) \quad (88)$$

Our measurement $\hat{\sigma}_f$ will thus follow the probability density function:

$$f_{\chi}\left(\sqrt{M} \frac{x}{\sigma_f}, M\right) = \frac{2x\sqrt{M}}{\sigma_f} f_{\chi^2}\left(M \frac{x^2}{\sigma_f^2}, M\right) \quad (89)$$

Here x is in HZ .

The expected value of the measurement is then:

$$E[\hat{\sigma}_f] = E\left[\sqrt{\hat{\sigma}_f^2}\right] \cong \sqrt{E[\hat{\sigma}_f^2]} = \sigma_f \quad (90)$$

And its standard deviation can be calculated with identification. Let us compute:

$$\sqrt{E[\hat{\sigma}_f^2] \pm \sqrt{Var(\hat{\sigma}_f^2)}} = \sigma \sqrt{1 \pm \sqrt{\frac{2}{M}}} = \sigma_f \left(1 \pm \sqrt{\frac{2}{M}}\right)^{\frac{1}{2}} \quad (91)$$

Developed with a Taylor series:

$$\sqrt{E[\hat{\sigma}_f^2] \pm \sqrt{Var(\hat{\sigma}_f^2)}} \cong \sigma_f \left(1 \pm \frac{1}{2} \sqrt{\frac{2}{M}}\right) = \sigma_f \left(1 \pm \frac{1}{\sqrt{2M}}\right) \quad (92)$$

We can identify this last expression to:

$$E[\hat{\sigma}_f] \pm \sqrt{\text{Var}(\hat{\sigma}_f)} \quad (93)$$

Then it comes:

$$\Rightarrow \text{Var}(\hat{\sigma}_f) \cong \frac{\sigma_f^2}{2M} \quad (94)$$

These calculations can be verified by simulation with a Monte-Carlo process. We made 500 simulations of the measurement process with parameters, $f = 1.3125\text{MHz}$, $f_s = 200\text{MHz}$, $N_{\text{periods}} = 12500$, $K = W = 40$ and an input $\sigma_f = 4000\text{Hz}$. Figure 4.15 compares the obtained histogram to the stochastic model. We can see that the stochastic model agrees with the histogram.

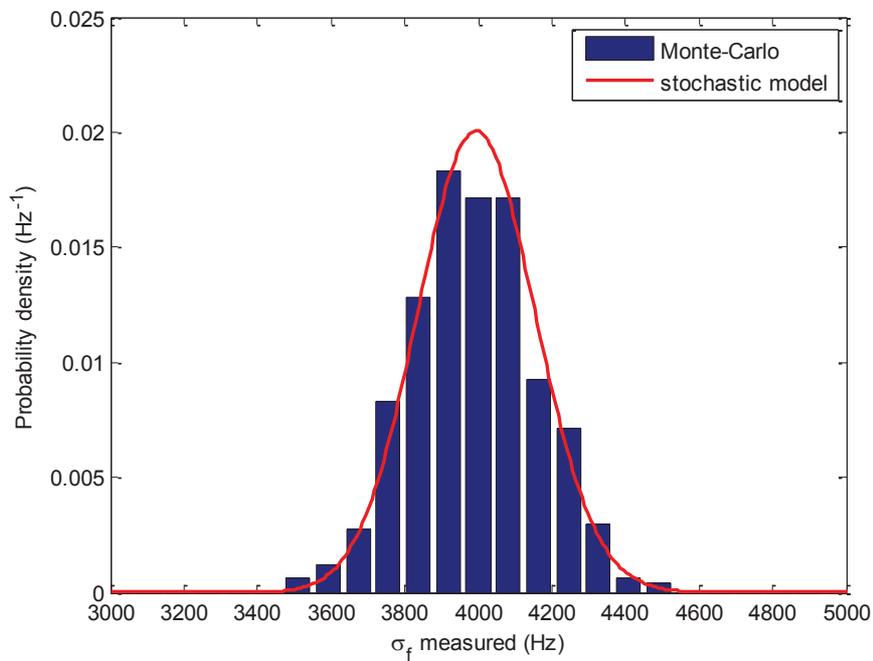


Figure 4.16 Validation of stochastic model for $\sigma_f = 4000\text{Hz}$

This calculation can be repeated over several values of σ_f . Figure 4.16 shows the color graph of the stochastic model density. Overlaid are some simulations. We see that the measurements of the simulations do indeed occur inside the most probable areas, validating the stochastic model.

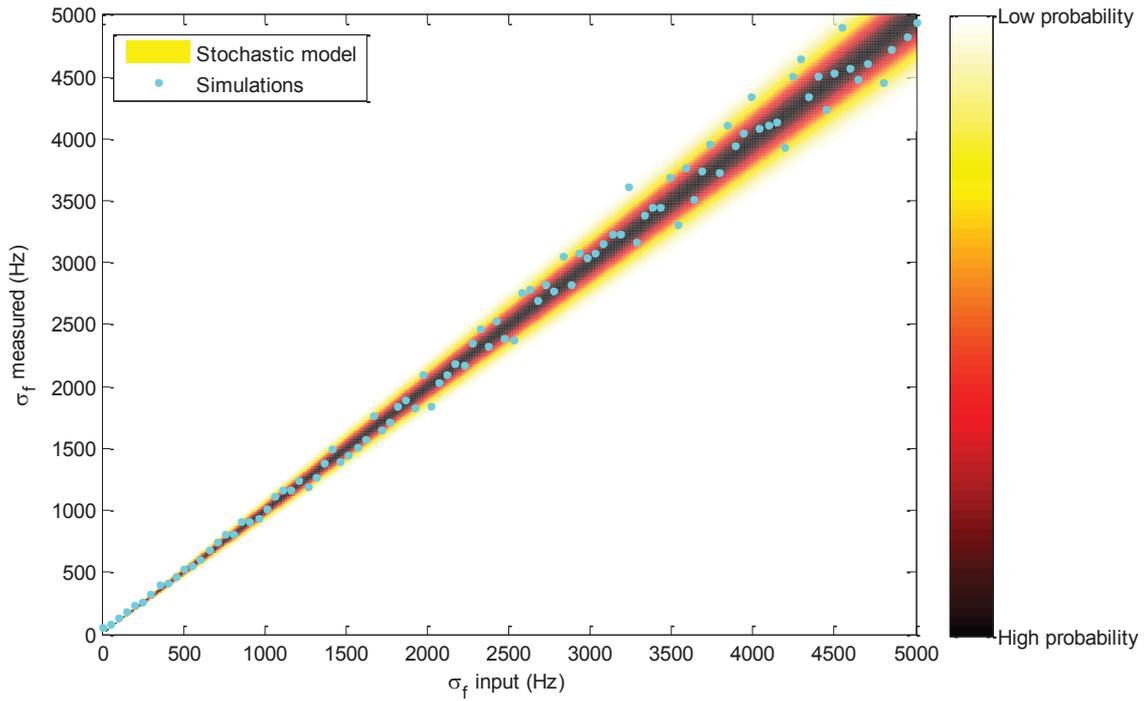


Figure 4.17 Validation of stochastic model

The interest of this development is that we now have an analytical expression for the variance of our phase noise estimator. Therefore, we can relate the standard error of the measurement $SE(\hat{\sigma}_f)$ to the signal and test conditions:

$$SE(\hat{\sigma}_f) = \sqrt{\text{Var}(\hat{\sigma}_f)} = \frac{\sigma_f}{\sqrt{2M}} = \sqrt{\frac{K}{2N_{\text{periods}}}} \cdot \sigma_f \quad (95)$$

This expression can be used by the test engineer to choose the appropriate test parameters for a given application. More specifically considering a given value of the phase noise level to be measured $\sigma_f|_{\text{typ}}$ and a desired measurement uncertainty $\Delta\varepsilon$, the targeted standard error can be defined with:

$$SE(\hat{\sigma}_f)|_{\text{target}} = \frac{|\Delta\varepsilon|}{n} \quad \text{for } \text{erf}(n/\sqrt{2}) \text{ confidence level} \quad (96)$$

The minimum number of signal periods to capture in order to satisfy this constraint is therefore given by:

$$N_{\text{periods}}|_{\text{min}} = \frac{K \cdot \sigma_f|_{\text{typ}}^2}{2} \cdot \left(\frac{n}{\Delta\varepsilon}\right)^2 \quad (97)$$

We can also differentiate equation (18) to consider the standard deviation in dBc/Hz:

$$dPN = \frac{20d\sigma_f}{\sigma_f \ln(10)} \quad (98)$$

So the same result can be obtained from the standard error in dBc/Hz :

$$N_{\text{periods}}|_{\text{min}} = \frac{K}{2} \left(\frac{20}{\ln(10)}\right)^2 \left(\frac{n}{\Delta\varepsilon|_{\text{dBc/Hz}}}\right)^2 \quad (99)$$

4.4 Experimental results

4.4.1 On synthesized signals

Hardware measurements in the lab have been performed to validate the proposed technique using the experimental setup described in figure 4.18. As introduced in section 2.2.1, generation of a Signal Under Test (SUT) featuring phase noise is accomplished through FM-modulation with white noise of a sinusoidal signal using an AWG and an RF generator. The phase noise level can be adjusted by means of the modulation depth f_{Δ} . This signal is then analyzed through two distinct flows. The first one corresponds to the conventional phase noise measurement, where a Mixed-Signal Oscilloscope (MSO) is used to capture digitized samples of the SUT and phase noise is measured on the signal spectrum computed through FFT. The second one corresponds to the proposed technique, where a Verigy 93K digital channel is used to perform 1-bit acquisition of the SUT and post-processing algorithm is applied to estimate phase noise from the captured binary vector. For the sake of comparison, we have also implemented the post-processing algorithm developed in chapter 3 that performs phase noise estimation from the analysis of instantaneous frequency distribution.

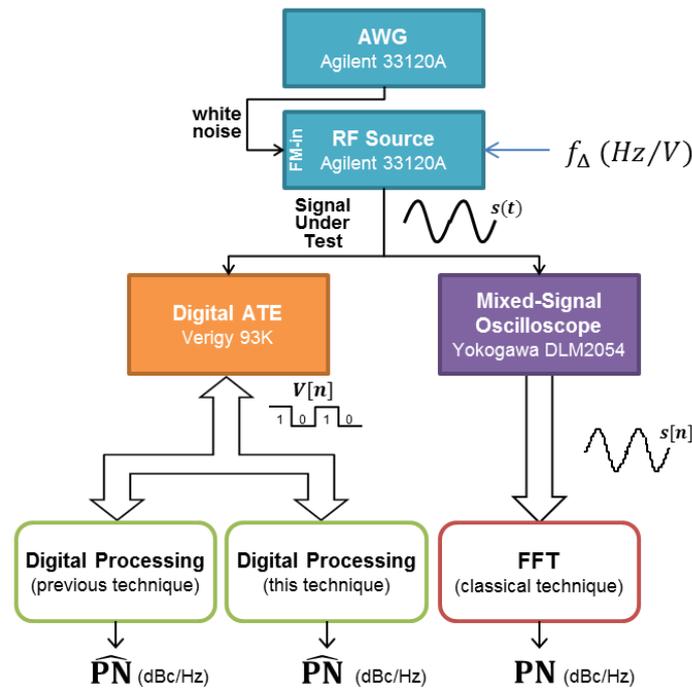


Figure 4.18 Experimental setup for hardware measurements on synthesized signals

Figure 4.19 summarizes measurement results obtained on a 1MHz signal sampled at 200MHz with analysis of 20,000 signal periods and phase noise evaluation at a frequency offset $f_{offset} = 12.5\text{kHz}$, so close to the signal carrier. Different levels of phase noise were injected varying the modulation depth f_{Δ} of the RF source from 100Hz up to 1MHz . These results show that both the previous digital technique and the new one have a good correlation with the conventional measurement on a large range of phase noise values from -40dBc/Hz to -80dBc/Hz . However while the previous technique diverges from conventional method for phase noise level below -80dBc/Hz , the new technique permits to achieve accurate evaluation down to -95dBc/Hz , which clearly demonstrates the superiority of the new technique.

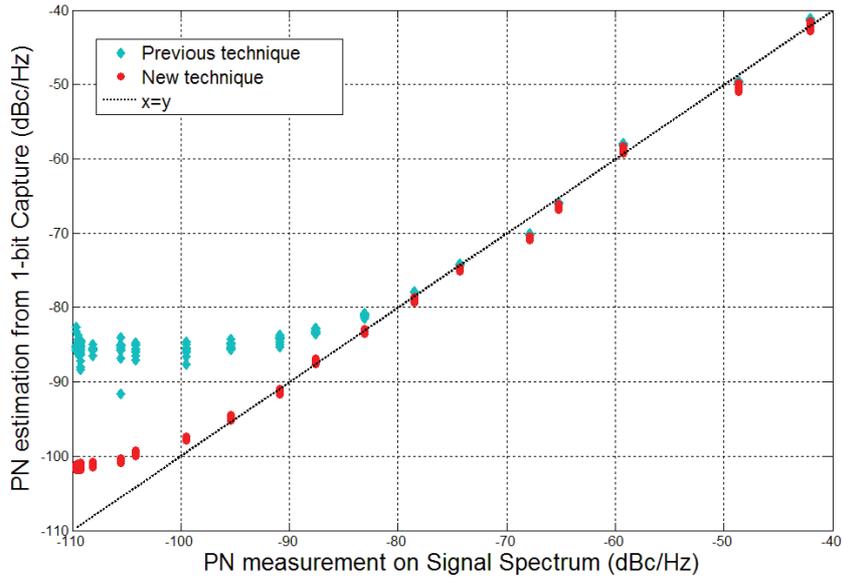


Figure 4.19 Comparison between PN estimation from digital capture and PN measurement on signal spectrum

4.4.2 On industrial devices

The proposed technique has also been evaluated in NXP’s development lab in Caen on an industrial device, a JN5168 wireless microcontroller. As illustrated in figure 4.19, both the conventional method and the new technique are implemented on Advantest 93K ATE using either an RF channel (with 32MHz sampling frequency of on 2,688 periods) or a digital channel (with 200MHz sampling frequency on 52,500 periods). Phase noise measurements were performed on the 1.3125MHz sinusoidal signal delivered on the transceiver output for different values of the measurement frequency offset.

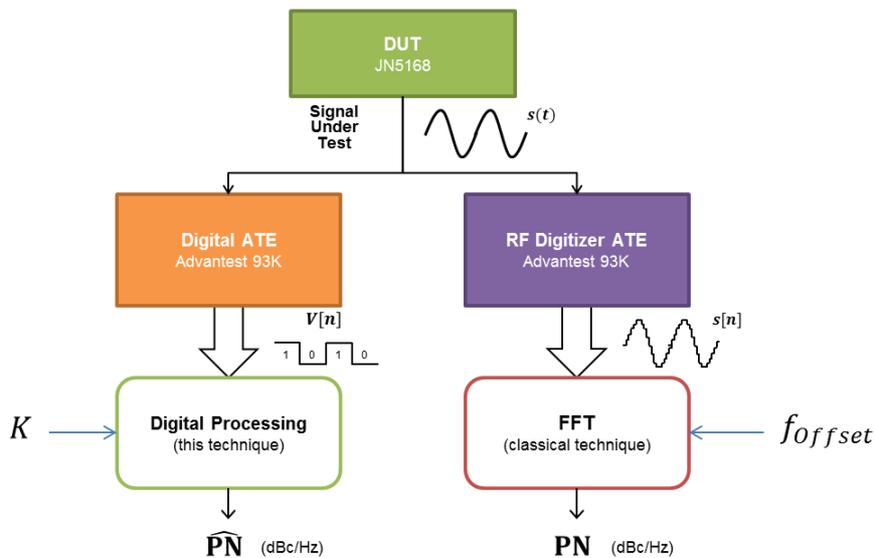


Figure 4.20 Experimental setup for industrial measurements

Results are summarized in figure 4.21 which reports the measured phase noise value with respect to the frequency offset. A very good agreement is observed between the digital technique and the conventional method on the complete range of frequency offset. Moreover it should be highlighted that the technique permits correct evaluation of frequency-domain phase noise characteristics, even when it differs from $1/f^2$ characteristics. This is a welcomed characteristic of the Allan deviation which permits identifying phase noise type from the slope of the phase noise level function to the frequency offset, or in case of the Allan deviation, to the observation interval $\tau = KT$.

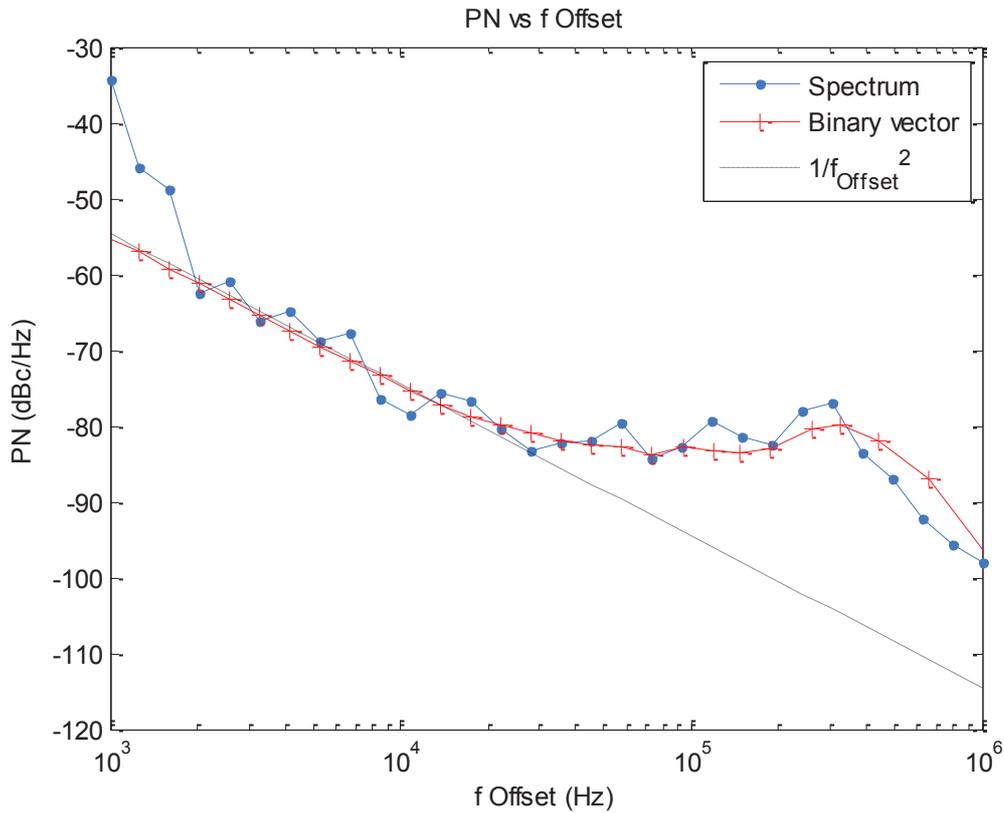


Figure 4.21 Comparison between PN estimation from digital capture and PN measurement on signal spectrum with different f_{Offset} on an industrial device

4.4.3 On industrial production line

With the help of NXP, the proposed method has been applied on 10,150 JN5168 devices out of the fab production line, by adding a new test method to the existing complete test flow. The measurement was done with the parameters: $f = 1.3125\text{MHz}$, $N_{periods} = 52,500$ and $K = 15$, which corresponds to a measurement frequency offset $f_{offset} = 87.5\text{kHz}$. All the calculations were carried out in the fab. Capture data were discarded, and only the final results were kept.

Obtained results are displayed in figure 4.22. A good correlation is obtained, with a Pearson correlation coefficient of 0.94846. The results show virtually no test escape but they show some possible yield loss. Concerning the test escape, no circuit were diagnosed with a $\Delta PN = PN_{FFT} - PN_{estimated} < -2\text{dBc/Hz}$ and only one was diagnosed with $\Delta PN < -1\text{dBc/Hz}$.

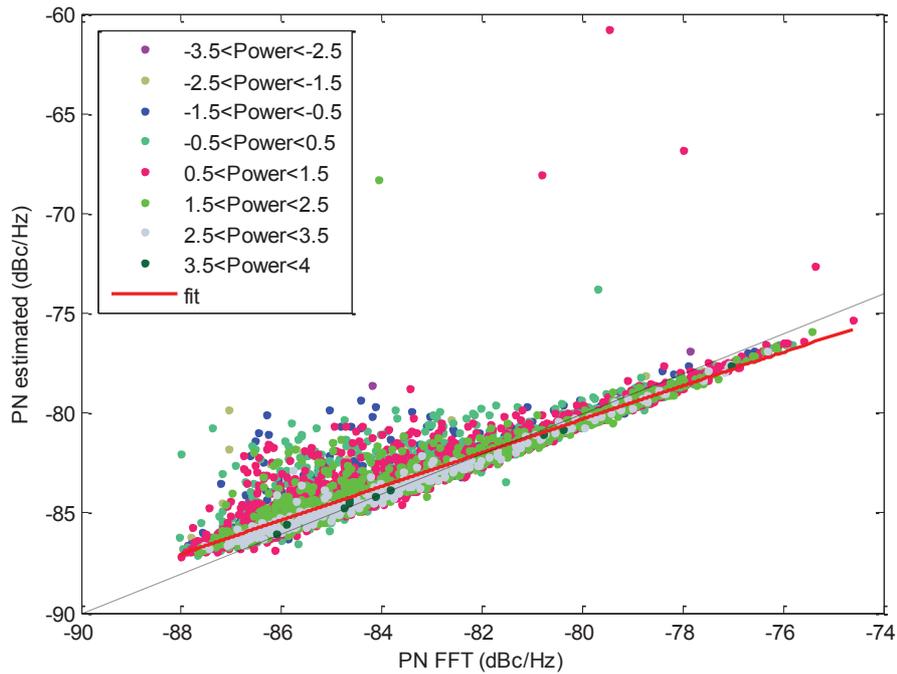


Figure 4.22 Results of measurements with classical technique versus proposed technique for 10 000 devices

Concerning yield loss, it has been noted that the power level of the signal is strongly correlated to the accuracy of the measurement as shown in figures 4.23.

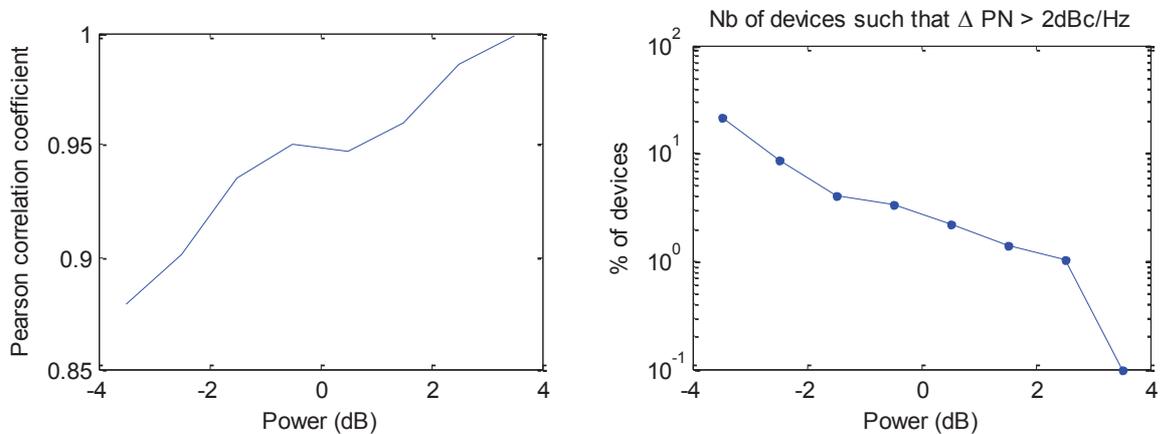


Figure 4.23 Pearson correlation coefficient and number of device with good measurement accuracy function of power

The devices with a measurement delta superior to $\Delta PN > 5 dBc/Hz$ were taken apart for further investigations in the NXP's lab. 64 devices over the 10150 evaluated fulfilled this criterion (0.6%). Based on additional measurements, it turned out that 60 to 80% percent of these devices would have been detected faulty by other tests.

Out of the remaining devices, with conflicting test results, a majority had instable (i.e. non repeatable) measures with the digital measurement technique. This instability has been resolved by truncating the first 5000 binary samples of the digital acquisition.

A minority of these devices still exhibit conflicting repeatable measurements. The reason of this conflict has been identified as an overestimation of the phase noise when measuring with an offset frequency between 20kHz and 100kHz as illustrated in figure 4.23. The reason for this overestimation (the bump observed around 40kHz) could not be explained.

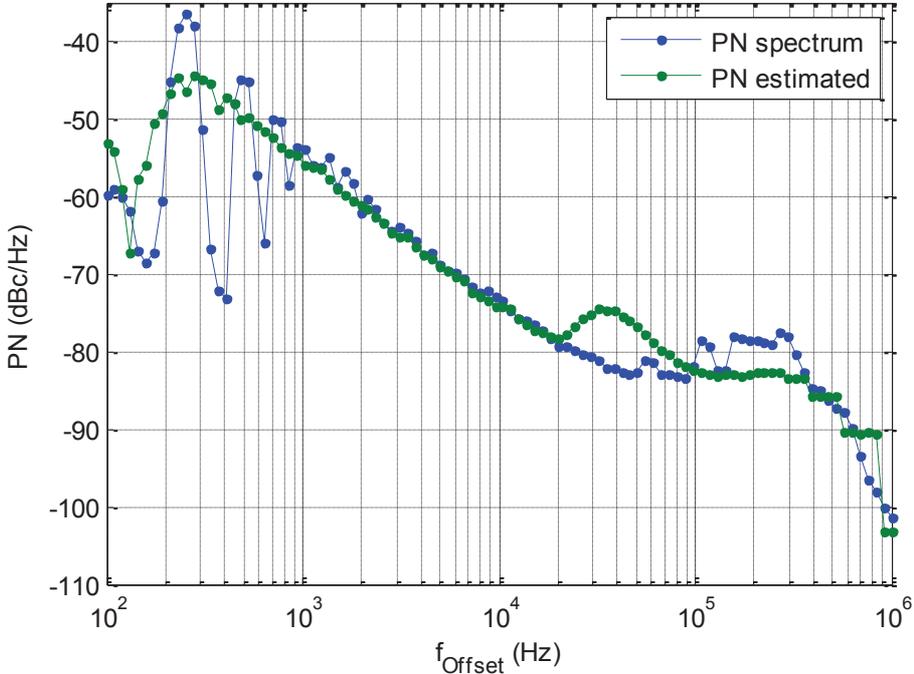


Figure 4.24 PN measurement on spectrum compared to PN estimated with proposed technique on a device with conflicting results

This issue, that concerns a very small percentage of the tested devices, can be solved using another value of f_{offset} for measurement. Unfortunately no other production run was planned to validate this statement. It would have been wiser to choose an f_{offset} in the $1/f^2$ slope section where the Brownian motion hypothesis for the noise holds.

Still, obtained results are very promising, and mostly validate the global approach with a real-scale experiment in production test conditions.

4.5 Conclusion

As a conclusion to this chapter, let us remind the main benefits of this new implementation in comparison to the previous one:

- The sampling frequency doesn't have to be a multiple of the signal frequency anymore
- The algorithm is robust whatever the sampling frequency and the amplitude noise
- One can identify the type of noise

The results are rather encouraging as less than 5 devices out of 10 000 tested devices exhibited an unexplained disagreement in the measurement. Moreover this disagreement might have been resolved by choosing better parameters and it involves yield loss which isn't as critical as test escape.

Chapter 5. On chip implementation

As mentioned in chapter 1, literature reports several attempts to move the required test resource from expensive ATE to embedded on-chip modules. In this chapter, we focus on the development of a digital BIST circuitry for on-chip phase noise evaluation of analog/RF signals. The proposed embedded test instrument relies on the phase noise measurement technique introduced chapter 4, i.e. the second proposed method based on Allan deviation.

Taking the form of an on-chip instrument, the measurement technique offer additional advantages compared to ATE implementation:

- The digital channels' performance and cost can be minimized since the measurement and process occurs on chip.
- The process can run in parallel on all the chips on the wafer
- Phase noise measurement can be done during the life time of the chip for aging control or parameter adjustments.

5.1 On chip algorithm

5.1.1 Processing only integers

In order to define an on-chip test instrument with small silicon area, we need to transform the measurement algorithm into a process that minimizes both algorithmic and memory resources. Regarding algorithmic resources, digital architecture implementation is actually made easier by using only integer numbers. Our objective is therefore to define a digital signature related to the phase noise level that can be computed using only integer numbers.

The starting point of the original post-processing algorithm is the detection of transitions in the binary vector, which occur at integer values of the sampling period T_s :

$$t_{transition}(i) = j_{transition}(i) \cdot T_s \quad (100)$$

where $j_{transition}(i)$ is the sample index at the transitions of the binary vector $V(n)$.

Times of extremum $t_{extr}(i)$ are then computed from these transitions according to eq.(53). Instead of working directly on the times of extremums, the idea is to work on integer values representative of these times. For this, we consider an index value $j_{extr}(i)$ defined by:

$$j_{extr}(i) = j_{transition}(i) + j_{transition}(i - 1) \quad (101)$$

$j_{extr}(i)$ can take only integer values. All the processing steps of the algorithm are then developed based on these indexes. Note that an interesting feature of the Allan deviation is that it is robust to bad estimation of the signal frequency f ; hence the computation of the phase fluctuation $\hat{\varphi}\left(i\frac{T}{2}\right)$ is not necessary and operations can be directly effectuated on j_{extr} as detailed hereafter.

First, we filter index values $j_{extr}(i)$ with a moving sum on W values:

$$\tilde{j}_{extr}(i) = \sum_{k=i}^{i+W-1} j_{extr}(k) \quad (102)$$

Then we compute increments between consecutive under-samples separated from K values:

$$\Delta_{K\tilde{j}_{extr}}(j) = \tilde{j}_{extr}(jK) - \tilde{j}_{extr}((j-1)K) \quad (103)$$

Then we compute the difference between successive non-overlapping increments:

$$\Delta\Delta_{K\tilde{j}_{extr}}(j) = \Delta_{K\tilde{j}_{extr}}(2j) - \Delta_{K\tilde{j}_{extr}}(2j-1) \quad (104)$$

And finally we compute the digital signature as the sum of squared differences:

$$SIG = \sum_{j=1}^M \Delta\Delta_{K\tilde{j}_{extr}}(j)^2 \quad (105)$$

where M corresponds to the number of computed differences between successive non-overlapping increments. This number can be an output of the processing algorithm, or simply estimated using:

$$M = \left\lfloor \frac{N_{points}}{K} \frac{f}{f_s} \right\rfloor \quad (106)$$

The signature SIG can be related to the phase noise estimator $\hat{\sigma}_f$ defined by eq.(65) in the previous chapter with:

$$SIG = \left(\frac{f_s}{f^2} \cdot 2W\sqrt{MK} \cdot bias' \cdot \hat{\sigma}_f \right)^2 \quad (107)$$

5.1.2 Semi-pipelined implementation

In order to minimize the required memory resources, the key idea is to implement on-the-fly processing of the 1-bit signal delivered at the output of the comparator. For this, we have chosen a pipelined implementation of the algorithm, where only a limited number of memory resources are required at each stage.

Here, the computation flow can be arranged in a semi-pipelined architecture. The word “semi” stands for all operations not occurring at every clock cycle. A possible implementation of the data flow is illustrated in Fig. 5.1.

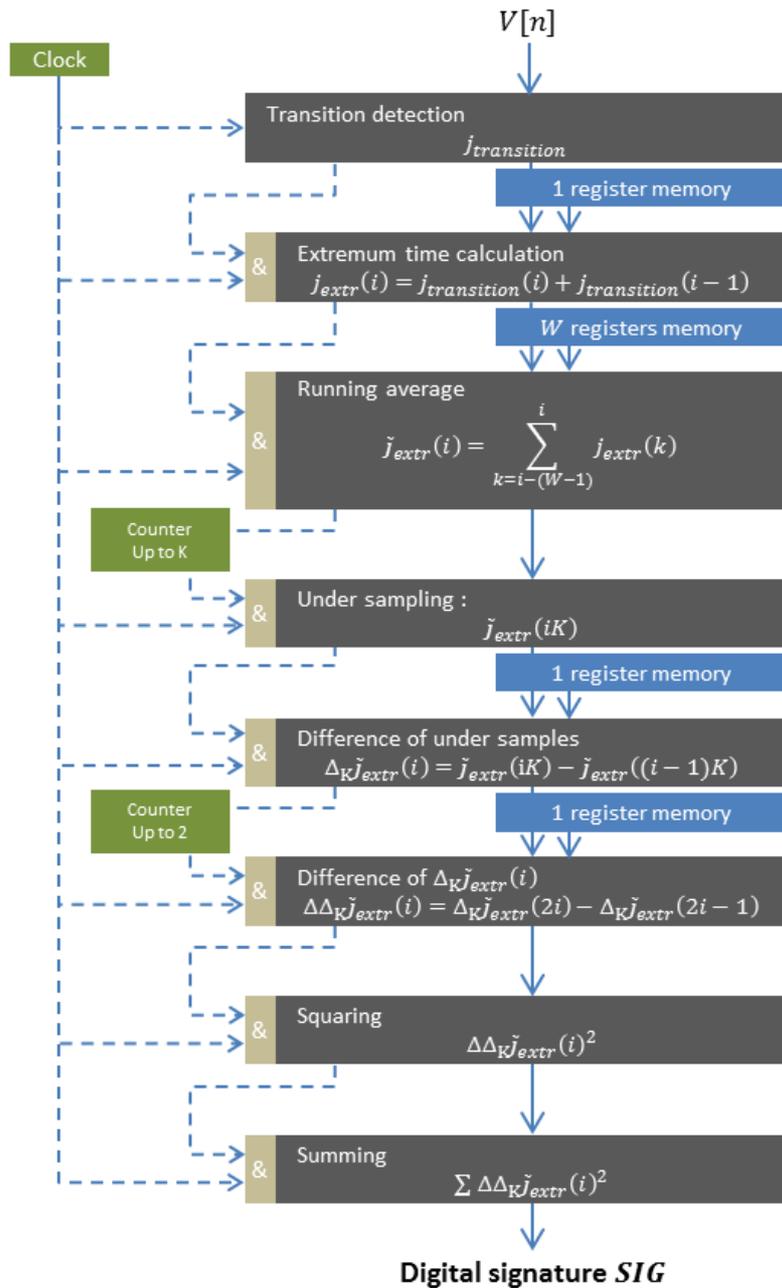


Figure 5.1 Example of semi-pipelined implementation for on-the-fly processing

Note that in addition of memory resource optimization, common benefit of pipelined architectures is also the increase of data throughput by making operations serial and thus minimizing delays of critical paths. Let us remark that latency produced by such implementation as no effect in this study.

5.1.3 Modular arithmetic

5.1.3.1 Limiting the number of bits

Limiting the data size (i.e. number of bits) is important to restrain both operator's silicon area, and internal delays.

Our technique for phase noise measurement relies on the oversampled acquisition of a large number of signal periods. $j_{transition}$ stores indexes of sampled points corresponding to the transition times. Given the capture length and the oversampling rate, this register would reach millions, thus requiring 30 to 40 bits to hold data without overloading. Subsequent operators and memory blocks would then be expensive in terms of delays and area.

Since the Allan deviation is based on differences, most significant bits (MSB) are most often being subtracted. We can then discard them by using a modular arithmetic approach, suitable with all the operations used in the proposed algorithm. In order to successfully implement modular arithmetic, a fixed width $Nbits$ is chosen for all registers in the data processing, until last operators (squaring and sum). Using modular arithmetic, all operations are done modulo 2^{Nbits} . Practically, this is simply done by discarding register overflows.

5.1.3.2 Computation limit

The drawback of using modular arithmetic is that too large phase noise can't be computed correctly because most significant bits are discarded.

More precisely, if $\sigma(\Delta\Delta_{K\tilde{J}_{extr}}) \ll 2^{Nbits}$ then the digital signature SIG will be correctly computed. On the other hand, if $\sigma(\Delta\Delta_{K\tilde{J}_{extr}}) \gg 2^{Nbits}$, we have a register overflow and the truncated values of $\Delta\Delta_{K\tilde{J}_{extr}}$ are almost uniformly distributed between $\llbracket -2^{Nbits-1}, 2^{Nbits-1} - 1 \rrbracket \approx \llbracket -2^{Nbits-1}, 2^{Nbits-1} \rrbracket$. The variance of a uniformly distributed variable on this range is $\frac{2^{2Nbits}}{12}$. Thus the maximum value of SIG that can be measured is given by:

$$SIG_{sup} = \frac{2^{2Nbits}}{12} M \quad (108)$$

This expression can be used to determine the appropriate register sizing ensuring that faulty chips will be correctly classified. For this, SIG_{sup} should be chosen such that the maximum value $\hat{\sigma}_{sup}$ of computed phase noise is sufficiently far from the Pass/Fail limit, as illustrated in figure 5.2. Indeed in this case, no matter whether the computed digital signature SIG corresponds to the actual device phase noise or to embedded instrument overload, the device will be correctly classified as a faulty one.

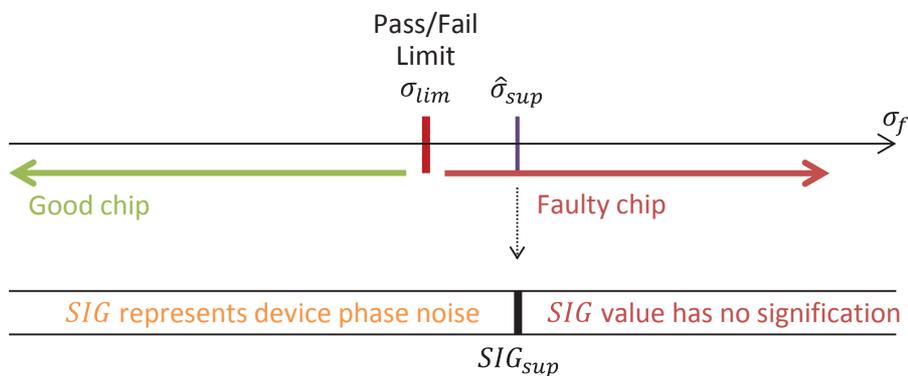


Figure 5.2 Choice of SIG_{sup}

Given the maximum value $\hat{\sigma}_{sup}$ of phase noise level to be correctly measured by the embedded instrument, the required number of bits for registers of the data processing is defined by:

$$2^{2Nbits} = 12 \left(\frac{f_s}{f^2} \cdot 2W\sqrt{K} \cdot bias' \cdot \hat{\sigma}_{sup} \right)^2 \quad (109)$$

5.1.4 From digital signature to PN value

The output of the embedded instrument is a digital integer representation of the phase noise. In the context of production test, the digital signature can be directly compared to established pass/fail limit. Still, it is possible to recover physical PN in terms of dBc/Hz using:

$$\hat{\sigma}_f = \frac{f^2}{f_s} \cdot \frac{1}{2W\sqrt{MK}} \cdot \frac{1}{bias'} \cdot \sqrt{SIG} \quad (110)$$

and eq. (18) to (20) established in chapter 2, which relate the phase noise level σ_f measured in the time domain to the phase noise value PN measured in the frequency domain.

Also note that as established in the previous chapter, the frequency offset of the measurement can be adjusted by setting the parameter $K = f/f_{offset}$.

5.1.5 Area estimation

In order to provide a rough understanding of the module complexity, an example of FPGA synthesis report is summarized in table 5.1. in terms of required hardware resources.

Qty	operator	Size of operators
1	MAC	13x13 to 48 bits
13	Adders/Subtractors	From 4 to 13 bits
4	Counters	From 11 to 41 bits
232	Registers and flip flops	
12	Comparators	From 1 to 41 bits

Table 5.1 Hardware resources involved in Processing Unit

The module has been entirely described using behavioral Verilog language and synthesized in a 65nm CMOS technology, under a frequency constraint of 200MHz. The resulting module area is found to be 34,000 μm^2 . For a device of about 9mm² in size, the embedded instrument would represent an overhead of approximately 0.3%.

5.2 FPGA implementation

In order to first validate above ideas, we have used a field-programmable gate array (FPGA) to implement the complete processing module. For availability reasons, the retained platform is a Digilent Nexys 3[®] board, featuring a Xilinx Spartan 6[®] FPGA. Architecture is written using Verilog language, and synthesized using Xilinx ISE[®] toolchain. For first experiments, registers have been arbitrarily sized to $N_{bits} = 13$. This achieves a good compromise between module dynamic range and maximum operating frequency. The main clock source is set at 125MHz by means of available phase locked loop. An UART interface is also implemented to control the instrument, and retrieve results.

The analog signal to be analyzed is interfaced to an FPGA digital input using a simple passive circuit that handles bias issue. Indeed FPGA only accepts input signals between $0V$ and $V_{dd} = 3.3V$. Moreover, the input buffer of the digital pin has probably a switching threshold around $\frac{V_{dd}}{2}$. A very basic RC network has been designed to ensure the correct adaptation of levels, as illustrated in figure 5.3. This network suppresses the DC component of the analog signal and adds a $\frac{V_{dd}}{2}$ offset to the signal. It therefore ensures that the signal applied to the FPGA digital input is centered on the switching threshold of the input buffer.

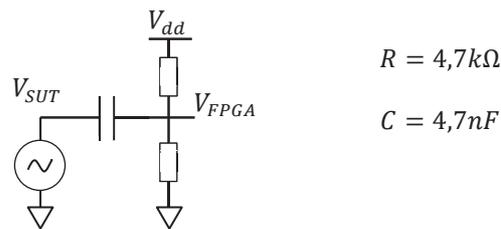


Figure 5.3 RC network for adaptation of voltage levels

Finally note that the input buffer that realizes the conversion of the analog signal into a binary signal may work asynchronously; it is therefore advisable to implement a latch register right after the input stage, in order to hold the converted data so that processing unit sees a stable input during a whole clock period.

5.3 Experimental results

5.3.1 Measurements on synthesized signal

First validations were performed on a synthesized signal affected with phase noise. The experimental setup is illustrated in figure 5.4. The signal under test is analyzed by two distinct flows. The first one corresponds to the conventional phase noise measurement. A Mixed-Signal Oscilloscope is used to capture digitized samples of the signal under test; FFT is then applied on these samples and phase noise is measured on the resulting spectrum. The second flow corresponds to the proposed technique where the digital test instrument is integrated on the FPGA of the Digilent Nexys 3[®] board. The test instrument delivers a digital signature SIG representative of the phase noise level present in the analog signal under test, which is converted into a phase noise value \widehat{PN} in dBc/Hz (cf. section 5.1.4) for comparison with the conventional phase noise measurement.

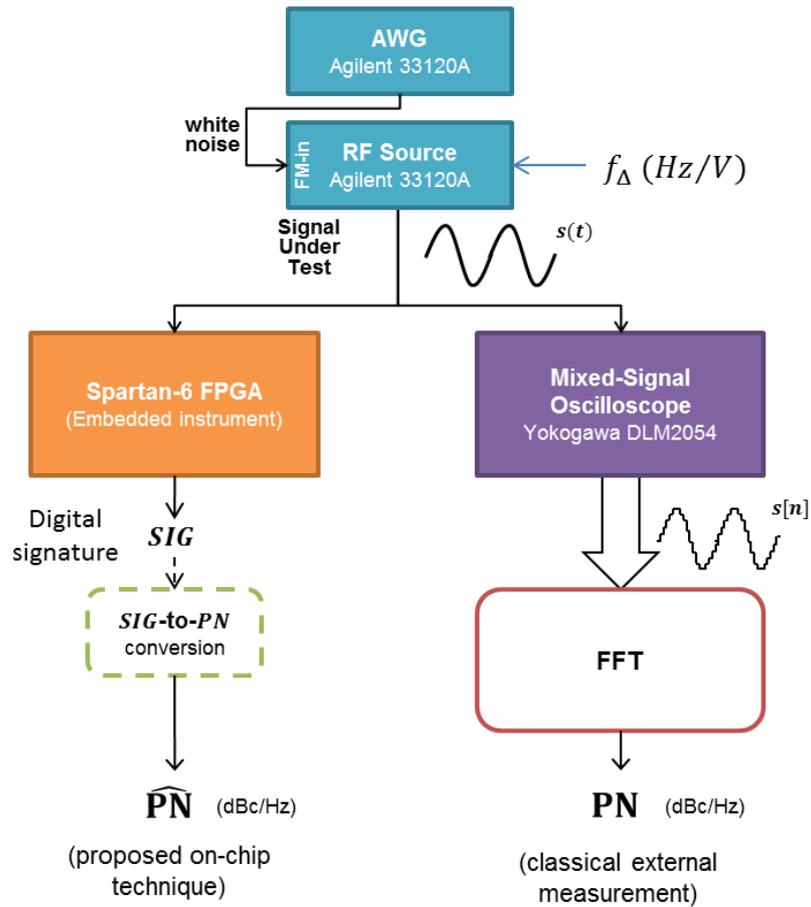


Figure 5.4 Experimental setup for measurements on synthesized signal

Measurements were performed on an arbitrary signal under test of 3V peak-to-peak amplitude and 1MHz frequency, for different phase noise levels corresponding to different values of the modulation depth f_{Δ} from 100Hz up to 1MHz. We set parameters $W = K = 15$, equivalent to a measurement frequency offset $f_{\text{Offset}} = 67\text{kHz}$. The sampling frequency of the oscilloscope matches the one of the FPGA at 125MHz. The number of sampling point is the same in both flows, equal to 1.25 million,

corresponding to 10,000 signal periods.

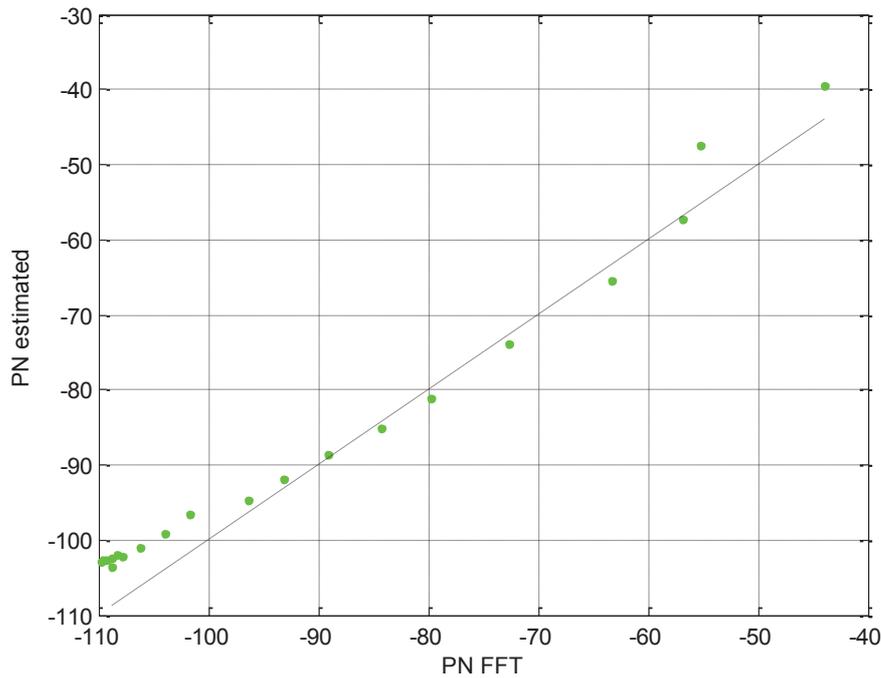


Figure 5.5 Hardware measurements on synthesized signal for different phase noise levels

Results are summarized in figure 5.5 that plots the PN value estimated using the digital test instrument according to the PN value measured using the classical technique. There is a good correlation on a very large range of injected phase noise. The classical technique is only better for a few dBc/Hz when measuring phase noise levels below $-100dBc/Hz$.

Figure 5.6 investigates effects of the running average window W . Results are produced by the integrated instrument only, for different values of the modulation depth f_{Δ} and $K = W$. In case of high modulation depth ($f_{\Delta} > 10kHz$), the estimated phase noise level $\hat{\sigma}_f$ is quite insensitive to the size of the running average window W , provided that is W not too large. Indeed as expressed by eq.(106), the maximum phase noise level that can be measured with this implementation depends on the register size ($Nbits$).

Note that there is a perfect agreement between the theoretical value $\hat{\sigma}_{sup}$ of the maximum phase noise level that can be measured and the observed one. In case of low modulation depth, we observe a saturation of the phase noise level that can be measured; the larger W and the lower the phase noise level that can be measured. Practically, W and K should be set according to the desired f_{offset} (cf. eq.(70)).

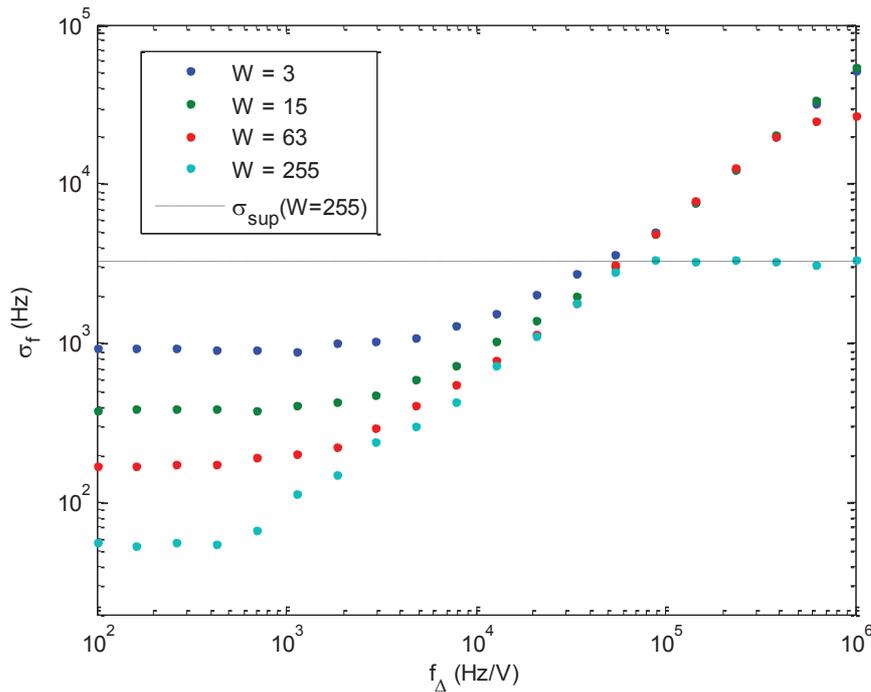


Figure 5.6 Hardware measurements on synthesized signal varying parameter W

5.3.2 Measurement on NXP silicon tuner

Next, we have evaluated the capability of the embedded instrument on NXP TDA18260 silicon tuners. In particular, we have tested a set of 10 faulty devices and 1 good device. The signal under test is the IF output signal from the silicon tuner with a frequency of 1MHz and a magnitude of 500mV . Phase noise measurement is performed using either the integrated instrument or the classical technique, as illustrated in figure 5.7.

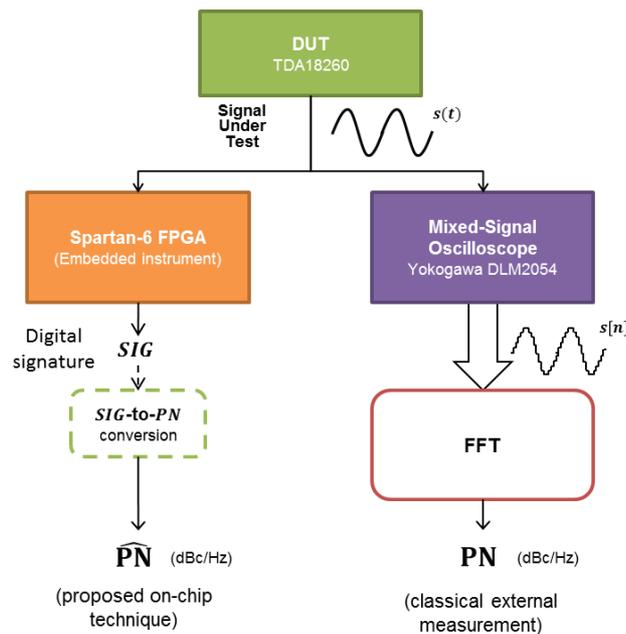


Figure 5.7 Experimental setup for measurements on NXP silicon tuner

First we have investigated whether the proposed method permits to recover the frequency-domain characteristics of phase noise, i.e. the phase noise spectrum. For this, data have been processed for different values of $W = K$, corresponding to different values of f_{Offset} . Results are reported in figure 5.8, which plots measured phase noise values with respect to the frequency offset. A very good agreement is observed between the proposed integrated digital measurement and the conventional one, both techniques revealing a $1/f^2$ phase noise characteristic for this product.

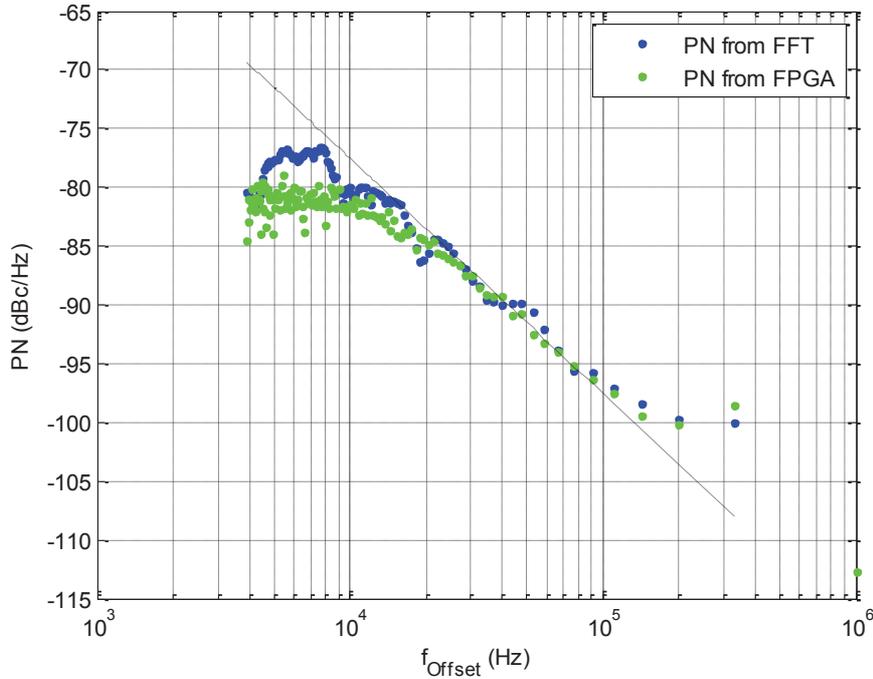


Figure 5.8 Hardware results for PN frequency domain characteristics on a faulty silicon tuner

Then we have investigated whether the proposed method permits to correctly sort the 11 available devices provided by NXP (1 golden device and 10 faulty devices). For this, we have set $f_{Offset} = 33kHz$ ($K = W = 33$) and we have analyzed 100,000 signal periods. Acquisition with the FPGA prototype is performed with $f_{clk} = 125MHz$ and $N_{samples} = 12.5 million$, while acquisition with the oscilloscope is performed with $f_s = 12.5MHz$ and $N_{samples} = 1.25 million$. The digital measurement with the FPGA is repeated 10 times in order to evaluate measurement variability. The conventional measurement is performed only once because of the long time required for transferring 1.25 million data.

Results are summarized in figure 5.9 that plots the PN value estimated using the digital test instrument according to the PN value measured using the classical technique. These results show that the integrated test instrument is able to clearly differentiate faulty devices from the good one. Again the classical method and the proposed technique show a strong correlation. The two approaches differ in results by only a few dBc/Hz . This may be due to aliasing phenomenon, occurring with the classical method, where data are sampled at lower rate.

The variability of results obtained with the FPGA is about $0.5dBc/Hz$ which is perfectly suitable with industrial requirements.

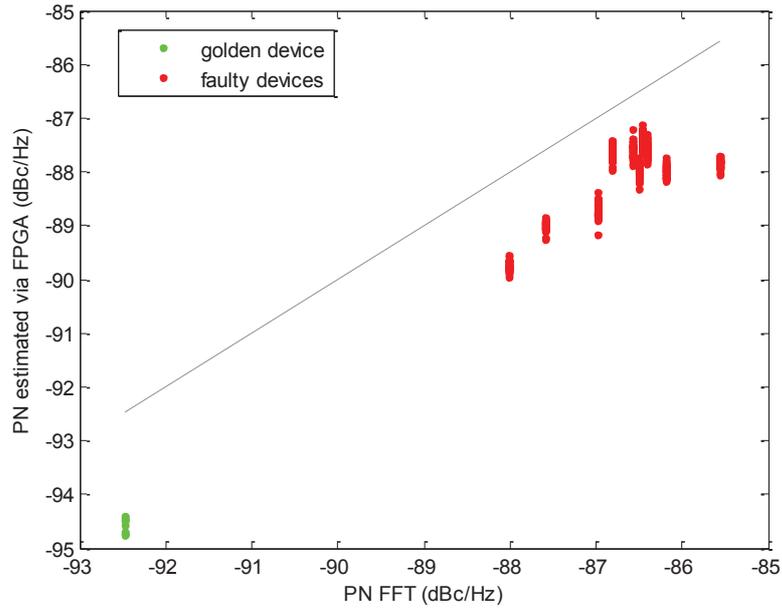


Figure 5.9 Measurements on several silicon tuner devices

5.4 Conclusion

In this chapter, we have developed an embedded instrument that enables on-chip phase noise measurement on analog/IF signals. This instrument relies on purely digital circuitry in order to minimize silicon area and maximize robustness to process variation. Moreover it is designed with a semi-pipeline architecture and processes only integers using modular arithmetic operators, which permits to minimize both algorithmic and memory resources.

The proposed instrument has been validated through a hardware FPGA implementation. Measurements have been performed, both on a synthesized signal and on manufactured devices. Results are very promising, in accordance with industrial requirements. In addition, the instrument is tuned by a simple set of parameters (W, K) and is tolerant to miscalibration.

General conclusion

The work presented in this thesis addresses the general need of test cost reduction in the context of analog and radio frequency signals. Indeed radiofrequency signals are everywhere, from your phone to your car remote control to ADSL and television signals. The spread of these devices lead industrials to look for new solutions to reduce cost of production for competition purposes. One of the main factors to be taken into account in the global fabrication cost is the cost of testing, especially in case of analog/RF circuits which usually necessitate the use of industrial testers equipped with dedicated and expensive resources in order to measure the device performances. An important performance to be verified for complex RF products is phase noise and this thesis proposes a new technique that permits phase noise measurement using only digital test resources. Essentially, these digital resources are used to perform 1-bit acquisition of the analog signal and dedicated post-processing algorithms have been developed in order to retrieve phase noise characteristics from simple processing of the captured binary vector.

In the first chapter, we have realized a brief overview of solutions proposed in the literature regarding phase noise testing. The objective of this work has then been defined and the proposed approach motivated with respect to the state of the art. It is concluded that few have tried characterizing phase noise through a binary capture. This thesis' positioning is to take advantage of the evolution of micro-electronic components. As transistors become smaller in size they allow higher working frequencies to the detriment of lower voltage resolution. We suggest exploiting the timing information from the zero-crossings to evaluate phase noise rather than measuring it in the frequency domain.

As a starting point for theoretical and simulation studies, a model that permits phase noise injection in a time-domain signal has been presented in the second chapter. This model relies on a one-dimension Brownian motion and corresponds to $1/f^2$ phase noise characteristics, which is typical of complex RF products that are the target of this thesis. The level of injected phase noise can be controlled by means of the standard deviation σ_f of the random variables involved in the Brownian motion. The validity of this model has been checked against a real device fabricated by NXP and an experimental setup allowing to synthesize a physical signal affected by phase noise has been developed. Finally, an analytical relationship between the phase noise level injected in the time domain and the phase noise level measured in the frequency domain has been established through a comprehensive simulation study.

In the third chapter, a first solution for measuring phase noise using only digital resources has been presented. 1-bit signal acquisition is performed with a sampling frequency that is an integer multiple of the signal frequency and the resulting binary vector is processed to derive instantaneous frequency estimates. Estimation of phase noise level is then accomplished by computing the mean absolute deviation of instantaneous frequency estimates. An integrated module allowing on-chip measurement has also been defined. Hardware measurements have demonstrated the validity of the approach but also unwanted sensitivity to amplitude noise. A refinement of the technique has been proposed that permits to extend the valid measurement range but constraints the oversampling ratio.

In the fourth chapter, a second solution has been presented, based on a different paradigm. This solution relies on phase fluctuation reconstruction instead of instantaneous frequency estimation. Estimation of phase noise level is then accomplished by computing the 2-sample deviation of phase fluctuation increments. As demonstrated from hardware measurements, this method presents several benefits from the previous one: it is robust to amplitude noise; the sampling frequency doesn't need to be a multiple integer of the signal frequency; and no fundamental limit on the sampling ratio is exhibited.

Finally in the last chapter, an embedded test instrument has been presented that allows on-chip phase noise measurement. This instrument relies on the phase noise estimator based on Allan deviation of phase fluctuation increments; however the processing algorithm has been redesigned in order to minimize the required memory and algorithmic resources. The module has been implemented on a FPGA and experimental results validate its ability to distinguish faulty devices from a golden one. This BIST solution permits even lower cost test equipment for testing the device as well as phase noise measurement during the life cycle of the chip.

To sum up, during this thesis, methods for measuring phase noise using only digital resources were proposed, thoroughly simulated, stochastically studied, redesigned for on-chip implementation, and experimentally validated in the lab and in the fab. The main benefit of the proposed methods resides in the dramatic reduction in the required test equipment since it can be implemented on a standard digital ATE and therefore permits to get rid of the expensive analog/RF tester resources required for the conventional method. Moreover because digital channels are usually available in a large number on a standard ATE, it also offers the possibility to implement multi-site testing in order to further reduce the testing costs. However it should be mentioned that since the proposed solution relies on an oversampled acquisition, it is not directed towards the analysis of signals with very high frequency above the gigahertz but targets products with an output in the intermediate frequency range.

This work opens interesting perspectives concerning low cost phase noise testing for analog and RF integrated circuits. Further investigations may be conducted to enlarge the application range of proposed methods towards circuits operating at higher frequency. One possibility is to exploit multiple level-crossings in order to decrease the quantization noise and therefore reduce the value of the required oversampling ratio. Another direction is to explore whether phase noise could be estimated from 1-bit coherent undersampled acquisition of the analog/RF signal. In this case, it would be possible to test circuits with operating frequency up to several GHz since current ATE can be equipped by digital test channels with data rate up to 8Gbps (for instance, Advantest Pin Scale 9G Digital Card).

Finally another interesting perspective is related to the possibility of performing on-chip measurement. Indeed the primary objective of a BIST solution is clearly the reduction of testing cost, but it can also be useful in a design context for calibration purpose. The availability of on-chip measurement facilities offers new options for designers and constitutes the first step towards the development of self-calibration and/or self-adaptation solutions.

Bibliographic references

- [1] "Test de circuits et systèmes intégrés" (Traité EGEM – Série Electronique et Microélectronique), *Hermès Science Publications, édité par C. Landrault*, ISBN 2-7462-0864-4, Mai 2004.
- [2] Poehl, F.; Demmerle, F.; Alt, J.; Obermeir, H., "Production test challenges for highly integrated mobile phone SOCs — A case study," *Proc. IEEE European Test Symposium (ETS)*, pp.17-22, 24-28 May 2010
- [3] Mittal, R.; Sontakke, A.; Parekhji, R., "Test time reduction using parallel RF test techniques," *Proc. IEEE VLSI Test Symposium (VTS)*, pp.40, 19-22 April 2010
- [4] Riley W.J., Howe D.A., "Handbook of Frequency Stability Analysis", *NIST Special Publication 1065*, July 2008
- [5] Hewlett & Packard, "Phase Noise Measurement Seminar", http://www.hparchive.com/seminar_notes/HP_PN_seminar.pdf, 1985
- [6] Feher K., "Telecommunications measurements, analysis and instrumentation", *Englewood Cliffs, NJ: Prentice-Hall*, 1987
- [7] Rohde, U.L.; Poddar, A.K.; Apte, A.M., "Getting Its Measure: Oscillator Phase Noise Measurement Techniques and Limitations," *IEEE Microwave Magazine*, vol. 14, no. 6, pp.73-86, Sept.-Oct. 2013
- [8] Angrisani, L.; D'apuzzo, M.; D'Arco, M., "A digital signal-processing approach for phase noise measurement", *IEEE Trans. on Instrumentation and Measurement*, vol. 50, no. 4, pp.930-935, Aug. 2001
- [9] Angrisani, L.; Baccigalupi, A.; D'Arco, M., "Method for close-to-the-carrier phase noise measurement," *IEE Proc. Science, Measurement and Technology*, vol. 150, no. 1, pp.35-39, Jan. 2003
- [10] Angrisani, L.; Baccigalupi, A.; D'Arco, M., "Evaluating phase noise power spectrum with variable frequency resolution", *IEEE Trans. on Instrumentation and Measurement*, vol. 53, no. 3, pp.685-691, June 2004
- [11] Angrisani, L.; D'Arco, M.; Greenhall, C.; Schiano Lo Moriello, R., "Enlarging the Input Frequency Range of a Phase Noise Meter through Bandpass Sampling Strategies," *Proc. IEEE Instrumentation and Measurement Technology Conference (IMTC)*, pp.1_6, 1-3 May 2007
- [12] Angrisani, L.; Schiano Lo Moriello, R.; D'Arco, M.; Greenhall, C.A., "A Digital Signal Processing Instrument for Real-Time Phase Noise Measurement," *IEEE Trans. on Instrumentation and Measurement*, vol. 57, no. 10, pp.2098-2107, Oct. 2008

- [13] Cosart, L.D.; Peregrino, L.; Tambe, A., "Time domain analysis and its practical application to the measurement of phase noise and jitter," *Proc. IEEE Instrumentation and Measurement Technology Conference (IMTC)*, vol. 2, pp.1430-1435, 1996
- [14] Cosart, L.D.; Peregrino, L.; Tambe, A., "Time domain analysis and its practical application to the measurement of phase noise and jitter", *IEEE Trans. on Instrumentation and Measurement*, vol. 46, no. 4, pp.1016-1019, Aug. 1997
- [15] Keller, Mark W.; Kos, A. B.; Silva, T. J.; Rippard, W. H.; Pufall, M. R., "Time domain measurement of phase noise in a spin torque oscillator", *Applied Physics Letters*, vol. 94 no. 19, p.193105-1-3, Nov. 2009
- [16] Ecker, A.; Blakkan, K.; Soma, M., "A digital method for phase noise measurement", *Proc. IEEE Int. Test Conference (ITC)*, paper 2.2, 2012
- [17] Kuo-Hsing Cheng; Shu-Yu Jiang; Zong-Shen Chen, "BIST for clock jitter measurements," *Proc. International Symposium on Circuits and Systems (ISCAS)*, vol. 5, pp.V-577-V-580, May 2003
- [18] Tian Xia; Lo, J.-C., "Time-to-voltage converter for on-chip jitter measurement," *IEEE Trans. on Instrumentation and Measurement*, vol. 52, no. 6, pp.1738-1748, Dec. 2003
- [19] Nelson, B.; Soma, M., "On-chip calibration technique for delay line based BIST jitter measurement," *Proc. International Symposium on Circuits and Systems (ISCAS)*, vol. 1, pp.1,944-7, 23-26 May 2004
- [20] Sunter, S.; Roy, A., "On-chip digital jitter measurement, from megahertz to gigahertz," *IEEE Design & Test of Computers*, vol. 21, no. 4, pp.314-321, July-Aug. 2004
- [21] Cazeaux, J.M.; Omana, M.; Metra, C., "Low-area on-chip circuit for jitter measurement in a phase-locked loop," *Proc. IEEE International On-Line Testing Symposium (IOLTS)*, pp.17-22, 12-14 July 2004
- [22] Maohsuan Chou; Jenchien Hsu; Chauchin Su, "A Digital BIST Methodology for Spread Spectrum Clock Generators," *Proc. Asian Test Symposium (ATS)*, pp.251-254, Nov. 2006
- [23] Sunter, S.; Roy, A., "Purely Digital BIST for Any PLL or DLL," *Proc. IEEE European Test Symposium (ETS)*, pp.185-192, 20-24 May 2007
- [24] Kahn Li Lim; Zilic, Z., "An undersampled duty cycle jitter BIST Circuit," *Proc. Midwest Symposium on Circuits and Systems (MWSCAS)*, pp.201-204, 5-8 Aug. 2007
- [25] Cheng, N.-C.D.; Yu Lee; Ji-Jan Chen, "A 2-ps Resolution Wide Range BIST Circuit for Jitter Measurement," *Proc. Asian Test Symposium (ATS)*, pp.219-223, 8-11 Oct. 2007
- [26] Jen-Chien Hsu; Chauchin Su, "BIST for Measuring Clock Jitter of Charge-Pump Phase-Locked Loops," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 2, pp.276-285, Feb. 2008
- [27] An-Sheng Chao; Soon-Jyh Chang, "A Jitter Characterizing BIST with Pulse-Amplifying Technique," *Proc. Asian Test Symposium (ATS)*, pp.379-384, 23-26 Nov. 2009

- [28] Jae Wook Lee; Ji Hwan Chun; Abraham, J.A., "A Random Jitter RMS Estimation Technique for BIST Applications," *Proc. Asian Test Symposium (ATS)*, pp.9-14, 23-26 Nov. 2009
- [29] Jianjun Yu; Dai, F.F., "On-chip Jitter Measurement Using Vernier Ring Time-to-Digital Converter," *Proc. Asian Test Symposium (ATS)*, pp.167-170, 1-4 Dec. 2010
- [30] Kinger, R.; Narasimhawsamy, S.; Sunter, S., "Experiences with parametric BIST for production testing PLLs with picosecond precision," *Proc. IEEE International Test Conference (ITC)*, pp.1-9, 2-4 Nov. 2010
- [31] Ching-Che Chung; Wei-Jung Chu, "An all-digital on-chip jitter measurement circuit in 65nm CMOS technology," *Proc. International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*, pp.1-4, 25-28 April 2011
- [32] Sakurai, M.; Niitsu, K.; Harigai, N.; Hirabayashi, D.; Oki, D.; Yamaguchi, T.J.; Kobayashi, H., "Analysis of jitter accumulation in interleaved phase frequency detectors for high-accuracy on-chip jitter measurements," *Proc. International SoC Design Conference (ISOCC)*, pp.146-149, 17-18 Nov. 2011
- [33] Abdel-Hafeez, S.; Harb, S.M.; Lee, K.M., "On-chip jitter measurement architecture using a delay-locked loop with vernier delay line, to the order of giga hertz," *Proc. International Conference on Mixed Design of Integrated Circuits and Systems (MIXDES)*, pp.502-506, 16-18 June 2011
- [34] Sunter, S.; Roy, A., "A Mixed-Signal Test Bus and Analog BIST with 'Unlimited' Time and Voltage Resolution," *Proc. IEEE European Test Symposium (ETS)*, pp.81-86, 23-27 May 2011
- [35] Niitsu, K.; Sakurai, M.; Harigai, N.; Yamaguchi, T.J.; Kobayashi, H., "CMOS Circuits to Measure Timing Jitter Using a Self-Referenced Clock and a Cascaded Time Difference Amplifier With Duty-Cycle Compensation," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 11, pp.2701-2710, Nov. 2012
- [36] Mendez-Rivera, M.; Silva-Martinez, J.; Sanchez-Sinencio, E., "On-chip spectrum analyzer for built-in testing analog ICs," *Proc. IEEE Int. Symposium on Circuits & Systems (ISCAS)*, vol. 5, pp.61-64, 2002
- [37] Erdogan, E.S.; Ozev, S., "A Robust, Self-Tuning CMOS Circuit for Built-in Go/No-Go Testing of Synthesizer Phase Noise", *Proc. IEEE Int. Test Conference (ITC)*, paper 33.1, 2006
- [38] Khalil, W.; Bakkaloglu, B.; Kiaei, S., "A Self-Calibrated On-Chip Phase-Noise Measurement Circuit With -75 dBc Single-Tone Sensitivity at 100 kHz Offset", *IEEE J of Solid-State Circuits*, vol. 42, no. 12, pp.2758-2765, Dec. 2007
- [39] Ouda, M.; Hegazi, E.; Ragai, H.F., "Digital on-chip phase noise measurement", *Proc. International Design and Test Workshop (IDT)*, pp. 1-5, 2009
- [40] Pous, N., "Analyse de signaux analogiques/radiofréquences à l'aide l'un équipement de test numérique", *Ph.D. thesis, Université Montpellier II*, 15 Nov. 2011

- [41] Pous, N.; Azais, F.; Latorre, L.; Rivoir, J., "On the use of standard digital ATE for the analysis of RF signals", *Proc. IEEE European Test Symposium (ETS)*, pp. 43-48, 2010
- [42] Pous, N.; Azais, F.; Latorre, L.; Rivoir, J., "A Level-Crossing Approach for the Analysis of RF Modulated Signals using only Digital Test Resources", *Journal of Electronic Testing: Theory and Applications (JETTA)*, vol. 27, no. 3, pp 289-303, 2011
- [43] Pous, N.; Azais, F.; Latorre, L.; Confais, G.; Rivoir, J., "Level-crossing based QAM demodulation for low-cost analog/RF testing", *Proc. IEEE Int. New Circuits and Systems Conference (NEWCAS)*, pp. 309-312, 2011
- [44] Azais, F.; Latorre, L., "Low-cost SNR estimation of analog signals using standard digital automated test equipment (ATE)," *Proc. IEEE Int. New Circuits and Systems Conference (NEWCAS)*, pp. 197-200, 2012
- [45] Su, C.C.; Chang, C.S.; Huang, H.W.; Tu, D.S.; Lee, C.L.; Lin, J.C.H., "Dynamic analog testing via ATE digital test channels," *Proc. Asian Test Symposium (ATS)*, pp.308-312, Nov. 2004
- [46] Logan, B. F., "Information in the Zero Crossings of Bandpass Signals", *Bell System Technical Journal*, vol. 56, pp. 487-510, April 1977
- [47] Watanabe, S.; Akimoto, Y.; Komatsu, T.; Saito, T., "A new stabilized zero-crossing representation in the wavelet transform domain and signal reconstruction," *International Conference on Image Processing*, vol.1, pp.37-40, vol.1, 23-26 Oct 1995
- [48] Foster, J.; Wang, T.-K., "Speech coding using time code modulation," *IEEE Proceedings of Southeastcon*, pp.861-863, vol.2, 7-10 Apr 1991
- [49] Raz, G.H.; Siegel, M., "Modeling and simulation of FM detection by zero-crossing in the presence of noise and FH interference," *IEEE Transactions on Broadcasting*, vol. 38, no. 3, pp.192-196, Sep 1992
- [50] Lee, E.K.B., "Zero-crossing baseband demodulator", *Personal, Indoor and Mobile Radio Communications*, pp 466-470, 1995.
- [51] Kwon, H.M.; Huan Yue; Weiguang H.; In-Ha Hyun, "Improved zero-crossing demodulator for minimum shift keying under static environments," *Proc. IEEE Global Telecommunications Conference (GLOBECOM)*, vol. 2, pp.1024-1028, Nov 1995
- [52] Lee, E.K.B.; Kwon, H.M., "New baseband zero-crossing demodulator for wireless communications. I. Performance under static channel," *Proc. IEEE Military Communications Conference (MILCOM)*, vol. 2, pp.543-547, Nov 1995
- [53] Kwon, H.M.; Lee, E.K.B.; Weiguang H.; Yun Bai, "New baseband zero-crossing demodulator for wireless communications. II. Performance under fading channel," *Proc. IEEE Military Communications Conference (MILCOM)*, vol. 2, pp.548-552, Nov 1995
- [54] Raz, G.H., "Modeling and simulation of FM detection by zero-crossing in the presence of noise and DS interference," *Proc. IEEE Vehicular Technology Conference (VTC)*, vol. 2, pp.1409-1413, 1996

- [55] Kwon, H.M.; Lee, E.K.B.; In-Ha Hyun, "Improved Zero-IF Zero-Crossing Demodulator For Gaussian Minimum Shift Keying Signals in Land Mobile Channels", *Proc. IEEE Vehicular Technology Conference (VTC)*, pp 297-301, 1996.
- [56] Kwon, H.M.; Weiguang H., "Performance Analysis of Narrowband Digital FM Demodulator With Zero-Crossing Post Processors", *Proc. IEEE Military Communications Conference (MILCOM)*, pp 275-279, 1996.
- [57] Kwon, H.M.; Poh Chun L., "Frequency Offset Compensator for Land Mobile Zero-Crossing Digital FM Receiver", *Proc. IEEE Military Communications Conference (MILCOM)*, pp 745-749, 1996.
- [58] Ismailoglu, N.; Yalcin, T., "Low-power design of a digital FM demodulator based on zero-cross detection at IF," *Proc. IEEE Vehicular Technology Conference (VTC)*, vol. 2, pp.810813, 1999
- [59] Mun, K.S.; Kwon, H.M., "Symbol Synchronization for Zero-Crossing Demodulator under Static and Mobile Fading Environments", *Proc. IEEE Military Communications Conference (MILCOM)*, pp 341-345, 1999.
- [60] Samadian, S.; Hayashi, R.; Abidi, A.A., "Demodulators for a Zero-IF Bluetooth Receiver", *IEEE Journal of Solid-State Circuits*, pp 1393-1396, 2003
- [61] Scholand, T.; Jung, P., "Novel receiver structure for Bluetooth based on modified zero-crossing demodulation," *Proc. IEEE Global Telecommunications Conference (GLOBECOM)*, vol. 2, pp. 729-733, Dec. 2003
- [62] Sayiner, N.; Sorensen, H.V.; Viswanathan, T.R., "A level-crossing sampling scheme for A/D conversion," *IEEE Transactions on Circuits and Systems II*, vol. 43, no. 4, pp.335-339, Apr 1996
- [63] Allier, E.; Sicard, G.; Fesquet, L.; Renaudin, M., "A new class of asynchronous A/D converters based on time quantization," *Proc. International Symposium on Asynchronous Circuits and Systems (ASYNC)*, pp. 196- 205, 12-15 May 2003
- [64] Akopyan, F.; Manohar, R.; Apsel, A.B., "A level-crossing flash asynchronous analog-to-digital converter," *Proc. International Symposium on Asynchronous Circuits and Systems (ASYNC)*, pp.11-22, 13-15 March 2006
- [65] Pekau, H.; Yousif, A.; Haslett, J. W., "A CMOS Integrated Linear Voltage-to-Pulse-Delay-Time Converter for Time-Based Analog-to-Digital Converters," *Proc. International Symposium on Circuits and Systems (ISCAS)*, pp. 2373-2376, 2006
- [66] Kim, J.; Cho, S., "A Time-Based Analog-to-Digital Converter Using a Multi-Phase Voltage-Controlled Oscillator," *Proc. International Symposium on Circuits and Systems (ISCAS)*, pp. 3934-3937, 2006.
- [67] Fusayasu, T., "A Fast Integrating ADC Using Precise Time-to-Digital Conversion," *Proc. IEEE Nuclear Science Symposium*, vol. 1, pp. 302-304, Oct. 2007

- [68] Staszewski, R. B.; et al, "All-digital TX frequency synthesizer and discrete-time receiver for Bluetooth radio in 130-nm CMOS," *IEEE Journal Solid-State Circuits*, vol. 39, no. 12, pp. 2278-2291, Dec. 2004.
- [69] Mörters, P.; Yuval P., "Brownian Motion", Cambridge Series in Statistical and Probabilistic Mathematics, *Cambridge University Press*, ISBN 9780521760188, March 2010
- [70] J. Yor-Chesney "Mathematical Model for Finance", Springer

Publications

Papers

ETS 2013 : Poster : Analog Measurements based on Digital Test Equipment for Low-Cost Testing of Analog/RF Circuits, Stephane David-Grignot, Florence Azais, Laurent Latorre, François Lefevre

NEWCAS 2014 : Phase noise measurement on IF analog signals using standard digital ATE resources, S. David-Grignot, F. Azaïs, L. Latorre, F. Lefevre

IMS3TW 2014 : Stochastic model for phase Noise measurement from 1-bit signal acquisition, S. David-Grignot, F. Azaïs, L. Latorre, F. Lefevre

ITC 2014 : Low-Cost Phase Noise Testing of Complex RF ICs using Standard Digital ATE, S. David-Grignot, F. Azaïs, L. Latorre, F. Lefevre

LATS 2015 : A digital technique for the evaluation of SSB phase noise of analog/RF signals, S. David-Grignot, F. Azaïs, L. Latorre, F. Lefevre

ETS 2015 : A New Technique for Low-Cost Phase Noise Production Testing from 1-bit Signal Acquisition, S. David-Grignot, F. Azaïs, L. Latorre, F. Lefevre

DDECS'15 : Embedded test instrument for on-chip phase noise evaluation of analog/RF signals, S. David-Grignot, F. Azaïs, L. Latorre, F. Lefevre

IMSTW 2015 : Digital on-chip measurement circuit for built-in phase noise testing, S. David-Grignot, F. Azaïs, L. Latorre, F. Lefevre

Patent

October 17, 2013 : Method And Apparatus For Measuring Phase Noise, Florence Azais, Laurent Latorre, Stéphane David-Grignot, François Lefevre

List of Tables

Table 1.1 Replacement cost with detection level in the integration process.....	9
Table 1.2 Phase noise power law response	12
Table 3.1 Hardware resources involved in Processing Unit	51

List of Figures

Figure 2.1 Example of time-domain phase fluctuation generated using a one-dimension Brownian motion	21
Figure 2.2 Spectrum comparison between simulated signal with injected phase noise for different sampling frequencies	22
Figure 2.3 Phase noise measurement on simulation model for different sampling frequencies.....	22
Figure 2.4 Phase noise measurement for different values of injected phase noise level σf	23
Figure 2.5 Experimental setup for phase noise measurement on a synthesized signal affected with phase noise.....	24
Figure 2.6 White noise from arbitrary waveform generator	24
Figure 2.7 Histogram of white noise signal captured on AWG output	25
Figure 2.8 Spectrum of white noise signal captured on AWG output.....	26
Figure 2.9 Spectrum of simulated and synthesized 1MHz signals affected by phase noise	27
Figure 2.10 Phase noise measurement of a 1MHz synthesized signal for different values of the modulation depth.....	27
Figure 2.11 Experimental setup for phase noise measurement on industrial chip	28
Figure 2.12 Comparison of spectrum of simulated signal with signal from industrial chip.....	29
Figure 2.13 PN measured in frequency domain vs. PN injected in time domain, for different values of the signal frequency with $f_{offset} = 25kHz$	30
Figure 2.14 PN measured in frequency domain vs. PN injected in time domain, for different values of the signal frequency with $f_{offset} = 100kHz$	31
Figure 2.15 PN measured in frequency domain vs. signal frequency, for different values of the measurement frequency offset with $\sigma f = 1kHz$ (low phase noise level).....	32
Figure 2.16 PN measured in frequency domain vs. signal frequency, for different values of the measurement frequency offset with $\sigma f = 50kHz$ (high phase noise level)	32
Figure 2.17 PN measured in frequency domain vs. measurement frequency offset for different values of the signal frequency with $\sigma f = 1kHz$ (low phase noise level).....	33
Figure 2.18 PN measured in frequency domain vs. measurement frequency offset for different values of the signal frequency with $\sigma f = 50kHz$ (high phase noise level).....	34
Figure 2.19 PN computed with analytical formulation vs. PN measured through FFT computation, $f_{offset} = 25kHz$	35

Figure 2.20 PN computed with analytical formulation vs. PN measured through FFT computation, $f_{offset} = 100kHz$	35
Figure 3.1 1-bit acquisition of the signal under test. The transitions of the binary vector are used to estimate the instantaneous period.	38
Figure 3.2 Amplitude noise may cause perturbation in the binary vector near transitions	39
Figure 3.3 Perturbation in the binary vector can be filtered with a running average	39
Figure 3.4 Standard deviation of estimated instantaneous frequency vs. injected phase noise level, for different values of the oversampling ratio	41
Figure 3.5 Effect of frequency quantization on distribution of estimated instantaneous frequency ..	42
Figure 3.6 Estimation of PN level based on Mean Absolute Deviation of estimated instantaneous frequency.....	43
Figure 3.7 The maximum distance between a zero-crossing event and the closest sampling point is reduced by choosing an odd oversampling ratio n	44
Figure 3.8 Distributions of σf estimations obtained in simulation with $n = 200$ and $n = 199$	44
Figure 3.9 Block diagram of the proposed digital technique based on instantaneous frequency estimation.....	45
Figure 3.10 Embedded Test Instrument for phase noise measurement	46
Figure 3.11 Principe of deviation signal generation (associated with SUT rise-to-rise transitions)	47
Figure 3.12 General architecture of the proposed ETI.....	48
Figure 3.13 Block diagram of the processing unit	49
Figure 3.14 “edge_detector” block	50
Figure 3.15 “ref_signal_generator” block	50
Figure 3.16 “cum_adder” block.....	51
Figure 3.17 VHDL simulation example.....	52
Figure 3.18 Layout of the test chip with embedded test instrument	52
Figure 3.19 Experimental setup for hardware measurements	53
Figure 3.20 Comparison of PN measured with proposed technique and conventional method.....	54
Figure 3.21 Deviation of the estimated instantaneous frequency in case of a sinusoidal signal without phase noise. Regions with multiple unwanted deviations are observed at regular time intervals.	54
Figure 3.22 Deviation of the estimated instantaneous frequency in case of a simulated sinusoidal signal without phase noise and imperfect setting of the oversampling ratio ($\Delta f = 10Hz$). Single unwanted deviations are observed at regular time intervals	55
Figure 3.23 Deviation of the estimated instantaneous frequency in case of a simulated sinusoidal signal without phase noise, imperfect setting of the oversampling ratio ($\Delta f = 10Hz$) and jitter in the sampling clock ($\sigma_{RJ} = 400ps$). Regions with multiple unwanted deviations are observed at regular time intervals.....	55
Figure 3.24 Deviation of the estimated instantaneous frequency in case of a simulated sinusoidal signal without phase noise, with voltage noise ($\sigma_{Anoise} = 0.007 \times A$) and imperfect setting of the oversampling ratio ($\Delta f=10Hz$). Regions with multiple unwanted deviations are observed at regular time intervals.....	56
Figure 3.25 A didactic example to illustrate the influence of jitter and voltage noise on the conversion of the analog signal: (a) crossing event in the middle between sampling points and, (b) crossing event close to the sampling point	56
Figure 3.26 Number of deviations w.r.t. signal frequency for 1-bit acquisition at $f_s = 125MHz$ of a sinusoidal signal without phase noise.....	57

Figure 3.27 Example of deviations observed in case of (a) a simulated sinusoidal signal affected by 0.7% voltage noise, and (b) a simulated sinusoidal signal affected by 1.4% phase noise	58
Figure 3.28 Filter operation in case of (a) a simulated sinusoidal signal affected by 0.7% voltage noise, and (b) a simulated sinusoidal signal affected by 1.4% phase noise	59
Figure 3.29 Evolution over time of the cumulative sum of deviations	60
Figure 3.30 Sum of absolute deviations after application of the filter w.r.t. phase noise level injected in the analog signal (without voltage noise).	60
Figure 3.31 Validation of the fitting models that relate the sum of absolute deviations observed after application of the filter to the phase noise level present in the analog signal: (a) low phase noise level and (b) high phase noise	62
Figure 3.32 PN estimation with digital technique vs. PN measurement with conventional method, for a simulated 3.0487MHz signal sampled at 125MHz with 0.7% voltage noise	62
Figure 3.33 Novel post-processing algorithm for phase noise estimation from digital capture	63
Figure 3.34 Filter operation for a 3.0487MHz signal without phase noise delivered by the RF generator without frequency modulation, and acquisition with an oversampling ratio $n = 41$	63
Figure 3.35 Sum of absolute deviation after filtering vs. oversampling ratio.....	64
Figure 3.36 Filter operation for a 2.77MHz signal without phase noise delivered by the RF generator without frequency modulation, and acquisition with an oversampling ratio $n = 45$	64
Figure 3.37 Hardware validation: comparison between PN estimation with digital technique and PN measurement with conventional method for a 3.0487MHz signal sampled at 125MHz.....	65
Figure 3.38 Deviations visualized with their density and image of the time-domain phase shift on hardware capture with oversampling ratio $n = 41$	66
Figure 3.39 Deviations visualized with their density and image of the time-domain phase shift on hardware capture with oversampling ratio $n = 45$	67
Figure 4.1 Recovering σf from φnTs phase fluctuation	69
Figure 4.2 1-bit acquisition of the signal under test. The transitions of the binary vector are used to determine extremums and estimate the phase.....	70
Figure 4.3 Reconstruction of phase fluctuation from 1-bit digital capture	71
Figure 4.4 Under-sampling of phase fluctuation and distribution of $\Delta K\varphi$ increments for two values of injected phase noise.....	72
Figure 4.5 Filtering phase fluctuation with running average	73
Figure 4.6 Input phase fluctuation and resulting histogram of $\Delta K\varphi$ increments with and without running average filtering	74
Figure 4.7 Phase noise estimation σf based on Allan deviation with bias correction vs. running average window size W , for 3 different values of K (injected phase noise level $\sigma f = 500Hz$)	77
Figure 4.8 Block diagram of processing algorithm for phase noise estimation based on Allan deviation of phase fluctuation increments	77
Figure 4.9 Signal spectrums computed with FFT on full signal capture and on virtual signal recovered from 1-bit capture	79
Figure 4.10 Fourier transform of the full signal and the reconstructed signal from phase estimation	79
Figure 4.11 Estimated phase noise level σf vs. injected phase noise level σf	80
Figure 4.12 Estimated phase noise level σf vs. sampling frequency f_s on a sine-wave without phase noise ($\sigma f = 0Hz$).....	81
Figure 4.13 Estimated phase noise level σf vs. <i>Amplitude Noise</i> on a sine-wave without phase noise ($\sigma f = 0Hz$).....	82

Figure 4.14 Estimated phase noise level σf vs. sampling ratio n on a sine-wave without phase noise ($\sigma f = 0\text{Hz}$).....	83
Figure 4.15 Random simulation runs of a sine-wave without phase noise	83
Figure 4.16 Validation of stochastic model for $\sigma f = 4000\text{Hz}$	86
Figure 4.17 Validation of stochastic model	87
Figure 4.18 Experimental setup for hardware measurements on synthesized signals	88
Figure 4.19 Comparison between PN estimation from digital capture and PN measurement on signal spectrum.....	89
Figure 4.20 Experimental setup for industrial measurements.....	89
Figure 4.21 Comparison between PN estimation from digital capture and PN measurement on signal spectrum with different $f\text{Offset}$ on an industrial device	90
Figure 4.22 Results of measurements with classical technique versus proposed technique for 10 000 devices.....	91
Figure 4.23 Pearson correlation coefficient and number of device with good measurement accuracy function of power.....	91
Figure 4.24 PN measurement on spectrum compared to PN estimated with proposed technique on a device with conflicting results	92
Figure 5.1 Example of semi-pipelined implementation for on-the-fly processing.....	95
Figure 5.2 Choice of $SIGsup$	96
Figure 5.3 RC network for adaptation of voltage levels.....	98
Figure 5.4 Experimental setup for measurements on synthesized signal.....	99
Figure 5.5 Hardware measurements on synthesized signal for different phase noise levels.....	100
Figure 5.6 Hardware measurements on synthesized signal varying parameter W	101
Figure 5.7 Experimental setup for measurements NXP silicon tuner	101
Figure 5.8 Hardware results for PN frequency domain characteristics on a faulty silicon tuner	102
Figure 5.9 Measurements on several silicon tuner devices	103
Figure a.1 Deviations occur when the signal triggers (or un-triggers) a captured bit that wouldn't have been there with an ideal signal	119
Figure a.2 Phase shift vs time representation.....	119
Figure a.3 a discrete model is used for high level noise regime	120
Figure a.4 a discrete model is used for low level noise regime without filter but a continuous model is used for the process with filter	120
Figure a.5 Markov chain	120
Figure a.6 Illustration of the probability density construction for the increments of the Brownian Motion.....	121
Figure a.7 Probability distribution of deviations with $\sigma = 10\text{kHz}, f = 4\text{MHz}, fs = 204\text{MHz}, Nperiods = 4096$	124
Figure a.8 Probability distribution of deviations with $\sigma = 1\text{kHz}, f = 4\text{MHz}, fs = 204\text{MHz}, Nperiods = 4096$	124
Figure annex.9 Comparison between simulations of the digital post-process and the stochastic model for different levels of noise Probability distribution for deviation with $f = 4\text{MHz}, fs = 204\text{MHz}, Nperiods = 4096$	125
Figure a.10 The Brownian motion evolves in the transition states until it reaches an absorbing state.	125
Figure a.11 Markov chain representation and probabilities associated with each transition.....	126

Figure a.12 The Brownian motion can cross a horizontal at a position ν between two measurements with ν following a uniform distribution between 0 and 1	127
Figure a.13 Illustration for a renewal process.....	127
Figure annex.14 Probability distribution of deviation for the low level noise stochastic model case $\sigma = 1kHz, f = 4MHz, fs = 204MHz, Nperiods = 4096$	128
Figure a.15 Illustration of Brownian motion with threshold of detections which can be modeled as corridors.	129
Figure a.16 Stochastic probability density compared to histograms on academic and simulated Process Monte Carlo	130
Figure a.17 Illustration of renewal process.....	130
Figure a.18 Stochastic model compared to histograms of simulated process and academic Brownian motion	131
Figure a.19 Stochastic model compared simulated process for different inputs of $\sigma noise$	133

Annex

1. Stochastic model

In this annex, we present a stochastic model of the measurement technique described in chapter 3. This stochastic model helps understanding the profound behavior of the measurement process, which can help seeing potential improvements. It also permits to compute algebraically the expected value of the phase noise estimator and to determine its probability distribution. These informations can then be used by the test engineer to parametrize the capture for a desired measurement precision. This annex is organized in four parts: first we describe the general behavior of the Brownian motion in the context of a sampled sinusoid signal, then we develop the stochastic model for three different cases corresponding to high phase noise level, low phase noise level without filter and low phase noise level with filter.

1.1. General Behavior

The signal to be analyzed is a sine wave affected by phase noise

$$signal = \sin(\Omega(j) + \Omega_0) \quad (111)$$

with Ω_0 a random initial phase and the assumption that the phase noise is a one dimension Brownian motion.

$$\Omega(n) = 2\pi T_s \sum_{j=0}^n (f + b r_j) \quad (112)$$

Where r_j are random values following a centered normal distribution $\mathcal{N}(0, \sigma^2)$ and $b = \sqrt{n}$ is a standardizing factor (cf chapter 2).

Since $r_j \equiv \mathcal{N}(0, \sigma^2)$, at each simulation sample, the Brownian motion is incremented of a step following $\mathcal{N}(0, (2\pi b T_s \sigma)^2)$. Instantaneous frequencies are estimated every period. At the scale of a period T , the Brownian motion gets incremented by a step following $\mathcal{N}(0, (2\pi b T_s \sigma \sqrt{n})^2) = \mathcal{N}(0, (2\pi T \sigma)^2)$. Indeed the variance of a Brownian motion step is proportional to the time.

As illustrated in figure a.1, we have a change in the estimation of the instantaneous frequency each time the signal phase is $\frac{2\pi}{n}$, that is, from the formula of the phase (112):

$$2\pi T_s \sum_{j=0}^m b r_j = \frac{2\pi}{n} \quad (113)$$

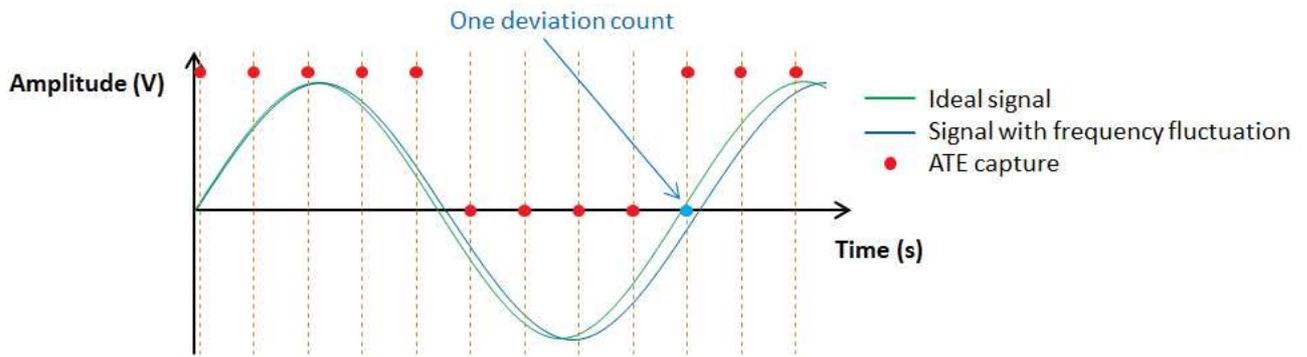


Figure a.1 Deviations occur when the signal triggers (or un-triggers) a captured bit that wouldn't have been there with an ideal signal

We can thus represent the Brownian motion as being in a corridor with barriers distant of $\frac{2\pi}{n}$. Or, because the detection occur only every period, we could rather represent the Brownian motion as being in a field with gates $\frac{2\pi}{n}$ wide and distant of T .

Over a period T , it is the same as studying a Brownian motion with an increment following $\mathcal{N}(0, (T\sigma)^2)$ with gates distant of $\frac{1}{n}$ or an increment following $\mathcal{N}(0, \sigma^2)$ with gates distant of $Q_f = \frac{f}{n}$.

One can represent the Brownian motion versus time as illustrated figure a.2. Points where the comparator is able to detect a change in the instantaneous frequency can be represented as gates. The integer multiple between f_s and f is preferred odd. This way, rising edge gates and falling edge gates aren't aligned with each other. This offers more accuracy and less redundancy.

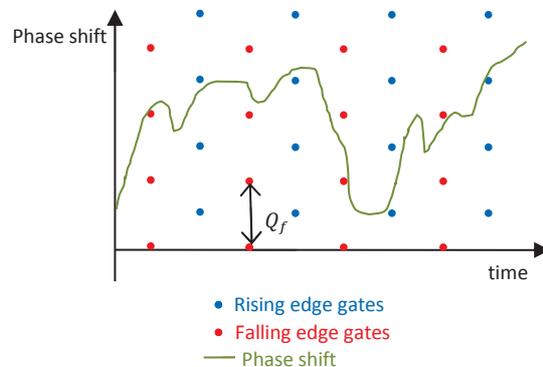


Figure a.2 Phase shift vs time representation

The measurement process exhibits distinct behaviors depending whether noise level is high or low. Let us first consider high noise levels. For high noise level, the same stochastic model applies with and without filtering, but this is not the case at low noise level.

At high noise level the Brownian motion is likely to trigger one or more detections at each period. The number of detections can be counted with an integer. The model for high noise level behavior is a discrete model based on a Markov chain. In this model the two gates behave independently.

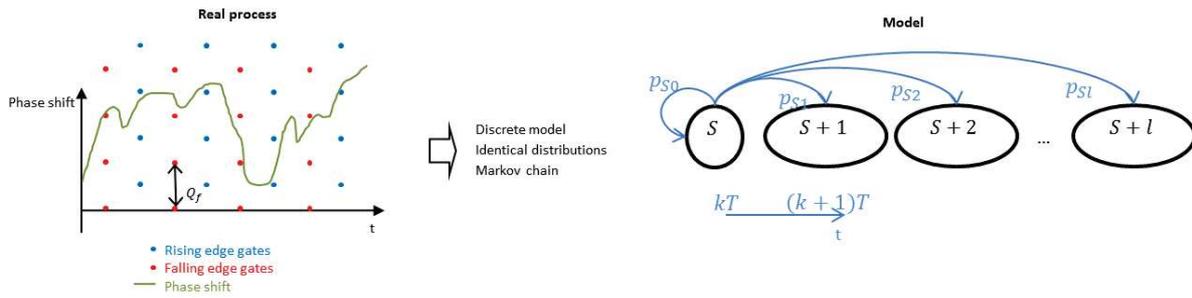


Figure a.3 a discrete model is used for high level noise regime

For low level of phase noise, behaviors differ with and without filter. Without filter, a discrete time Markov chain is used to analyze the path of the Brownian motion at each period. With filter a continuous time model is used instead. Discrete gates thus become continuous corridors. The stopping times of a Brownian motion in a corridor are calculated first and then we use renewal process to compute the detected number of deviations.

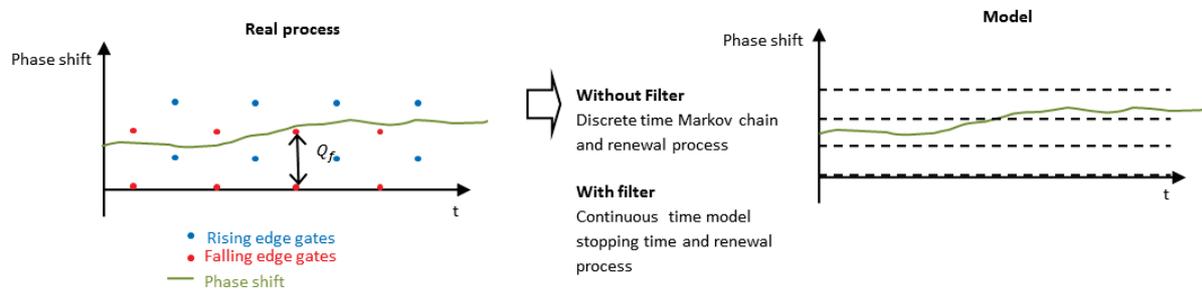


Figure a.4 a discrete model is used for low level noise regime without filter but a continuous model is used for the process with filter

1.2. High level noise

At high level of phase noise, i.e. high values of σ , for example over $10 Q_f$, every measurement is considered independent. For every measurement, the Brownian motion is just exiting the previous gate. As illustrated figure a.2, it has several possible outcomes: either it is entering into the gate right in front, or it's going to deviate (up or down on figure a.2) into a gate just on its side or into a gate further away. This model applies for both cases: process with or without filter. Indeed the filter only acts when the deviation is of just 0 or 1 unit; at high level noise this is unlikely to happen, deviations are the more often greater than 1.

We can now build a stochastic model based on a score number S . If the Brownian motion hits the gate right in front, it doesn't score any points, if it hits a gate just on its side, it scores 1 point. If it hits a gate further away it scores 2, 3 or more points depending on the distance. The points l add up with each measurement to give the score S . This constitutes a Markov chain illustrated in figure a.5

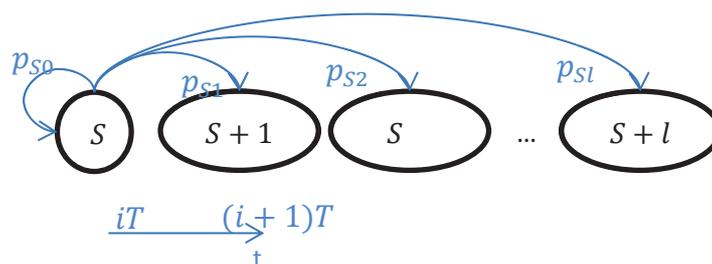


Figure a.5 Markov chain

Once we have the structure of the stochastic model, we need to compute the probabilities p_{Sl} for each transition. This can be done knowing that the Brownian motion's increments follow a normal distribution of standard deviation $\sqrt{\Delta t}\sigma$.

We also have to consider the Brownian motion can be anywhere into the gate of the previous measurement ranging from $-\frac{Q_f}{2}$ to $\frac{Q_f}{2}$. These two considerations are illustrated in figure a.6.

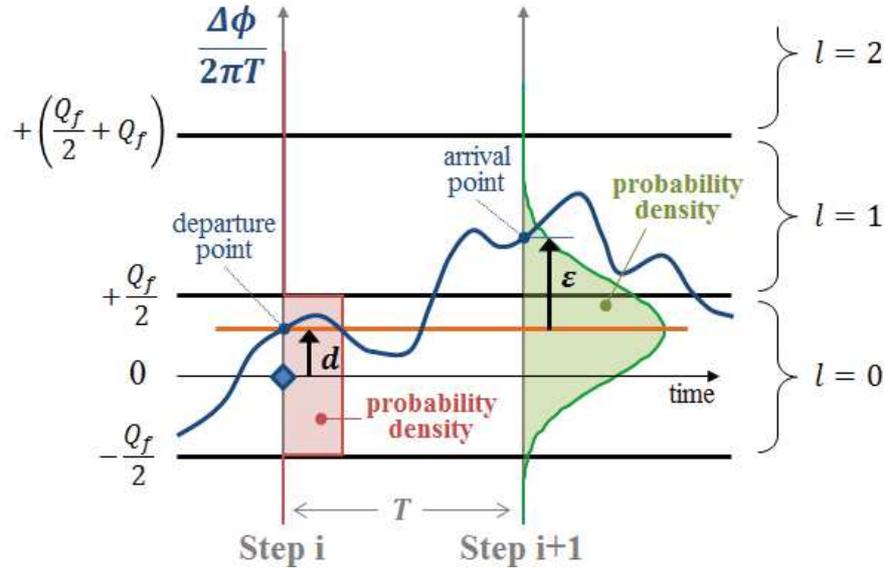


Figure a.6 Illustration of the probability density construction for the increments of the Brownian Motion

Let the index i indicates the i^{th} measurement, happening at the i^{th} period. Considering the probability of the Brownian motion being ϵ away from its departure point

$$p(\epsilon_i = \epsilon) = \frac{1}{\sigma \sqrt{2\pi}} \exp\left(-\frac{\epsilon^2}{2\sigma^2}\right) \quad (114)$$

and the probability of the position of its departure point :

$$p(d_i = d) = \begin{cases} \frac{1}{Q_f} & \text{for } d \in \left[-\frac{Q_f}{2}, \frac{Q_f}{2}\right] \\ 0 & \text{otherwise} \end{cases} \quad (115)$$

we can compute the probability of the arrival point being inside $\left[-\left(1 + \frac{1}{2}\right)Q_f, \left(1 + \frac{1}{2}\right)Q_f\right]$ with a double integral (the convolution is transparent here).

$$p_1 = \int \int \frac{1}{\sigma \sqrt{2\pi}} \frac{2}{Q_f} \exp\left(-\frac{\epsilon^2}{2\sigma^2}\right) d\epsilon dd$$

integrating on the area such that (116)

$$|\epsilon + d| < \left(1 + \frac{1}{2}\right)Q_f \text{ and } |d| < \frac{Q_f}{2}$$

That is :

$$\begin{aligned}
p_1 = (1 + 1) \operatorname{erf}\left(\frac{(1 + 1)Q_f}{\sigma\sqrt{2}}\right) - 1 \operatorname{erf}\left(\frac{1Q_f}{\sigma\sqrt{2}}\right) \\
+ \frac{\sqrt{2}\sigma}{\sqrt{\pi}} \frac{1}{Q_f} \left(\exp\left(-\frac{(1 + 1)Q_f^2}{2\sigma^2}\right) \right. \\
\left. - \exp\left(-\frac{1Q_f^2}{2\sigma^2}\right) \right)
\end{aligned} \tag{117}$$

With this expression, we can compute the probability for the Brownian motion to stay in the same state from one measurement to the next as the probability p_0 of the arrival point to be in the interval $\left[-\frac{Q_f}{2}, \frac{Q_f}{2}\right]$:

$$p_0 = \operatorname{erf}\left(\frac{Q_f}{\sigma\sqrt{2}}\right) - \frac{\sqrt{2}\sigma}{Q_f\sqrt{\pi}} \cdot \left(1 - \exp\left(\frac{Q_f^2}{2\sigma^2}\right) \right) \tag{118}$$

Similarly, we can compute the probability for the Brownian motion to step from one measurement to the next into an interval which is l away, as the probability of the arrival point to be in the interval $\left[-\left(1 + \frac{1}{2}\right)Q_f, -\left(1 - 1\right) + \frac{1}{2}\right)Q_f\right]$ or $\left[\left(1 - 1\right) + \frac{1}{2}\right)Q_f, \left(1 + \frac{1}{2}\right)Q_f\right]$, which corresponds to $p_1 - p_{1-1}$.

Finally knowing all these probabilities, can then compute the expected value $E(S)$ of the score at the end of the measurement process, which is given by:

$$E(S) = \sum_{l=1}^{N-1} (p_l - p_{l-1}) 2lN_{\text{periods}} = \sigma \frac{2N_{\text{periods}}}{Q_f} \sqrt{\frac{2}{\pi}} \tag{119}$$

Replacing p_l and p_{l-1} by their expression and using appropriate mathematical transformations, we finally obtain a simple analytical expression for the expected value of the score S at the end of the measurement process:

$$E(S) = \sigma \frac{2N_{\text{periods}}}{Q_f} \sqrt{\frac{2}{\pi}} \tag{120}$$

This expression relates the expected value of the score number S at the end of the measurement process to the level of phase noise σ present in the analog input signal. Note that this expression is in agreement with the analytical expression established in chapter 3 based on the MAD function, considering that the score S corresponds to the sum of absolute deviations from the ideal count computed from 1-bit acquisition.

The model can also be used to determine the probability distribution of the score S at the end of the measurement process. More precisely, the stochastic model is characterized by the following transition matrix P :

$$P = \begin{bmatrix} p_0 & p_1 - p_0 & p_2 - p_1 & \cdots & p_{N-1} - p_{N-2} & 1 - p_{N-1} \\ 0 & p_0 & p_1 - p_0 & \cdots & p_{N-2} - p_{N-3} & 1 - p_{N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & p_0 & p_1 - p_0 & 1 - p_1 \\ 0 & 0 & 0 & 0 & p_0 & 1 - p_0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (121)$$

with an initial distribution vector $\pi_0 = [1 \ 0 \ \dots \ 0]$ corresponding to a score number $S = 0$ at the beginning of the measurement process and N an arbitrarily chosen size of matrix, for example $2E(S)$.

The Markov matrix is applied at each measurement to the previous distribution vector. Since it is always the same, the distribution vector $\pi_{N\text{periods}}$ at the end of the measurement process can be computed with:

$$\pi_{N\text{periods}} = \pi_0 P^{N\text{periods}} \quad (122)$$

Finally since the two gates work independently but identically, the probability distribution of the score S , is the result of a convolution:

$$\pi(S) = \sum_{i=0}^S \pi_{N\text{periods}}(S-i) \pi_{N\text{periods}}(i) \quad (123)$$

Note that efficient computation of $P^{N\text{per}-1}$ can be implemented taking into account the specific properties of the transition matrix P which is a triangular superior Toeplitz matrix. Indeed since P^n shares the same properties as P for all n , only the first row is to be defined to know all the matrix.

Let us denote:

$$\pi P^n = [p_{ij}(n)] \quad (124)$$

From the properties just described:

$$p_{1j}(2n) = \sum_{k=1}^N p_{1k}(n) p_{kj}(n) = \sum_{k=1}^j p_{1k}(n) p_{kj}(n) \quad (125)$$

$$p_{1j}(2n) = \sum_{k=1}^j p_{1k}(n) p_{1,j-k+1}(n) \quad (126)$$

And the first row at the $2n$ iteration equates to:

$$r(2n) = r(n) * r(n) \quad (127)$$

We can validate the above calculation by comparing the probability distribution computed using the stochastic model to the one observed with a Monte Carlo simulation of the digital post-process (4000 runs). Results are reported in figures a.7 and a.8 in for two different values of phase noise corresponding to high and low noise level respectively.

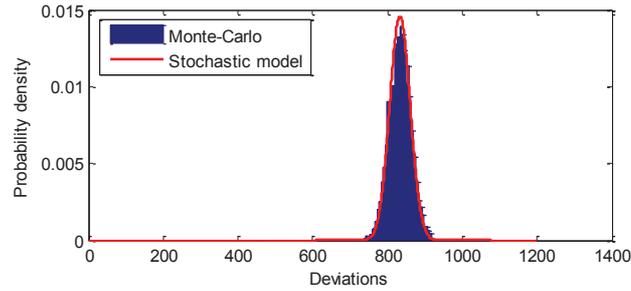


Figure a.7 Probability distribution of deviations with $\sigma = 10\text{kHz}$, $f = 4\text{MHz}$, $f_s = 204\text{MHz}$, $N_{\text{periods}} = 4096$

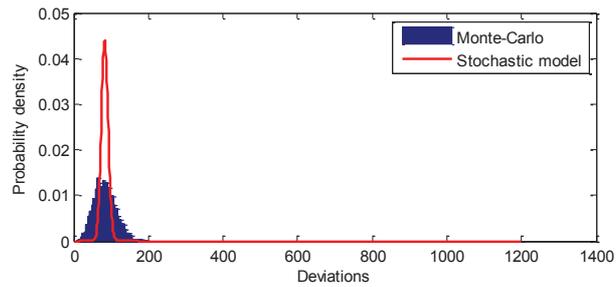


Figure a.8 Probability distribution of deviations with $\sigma = 1\text{kHz}$, $f = 4\text{MHz}$, $f_s = 204\text{MHz}$, $N_{\text{periods}} = 4096$

In case of the higher noise level, there is a perfect match between Monte-Carlo simulation and the stochastic model. However for the lower noise level, the dispersion predicted by the stochastic model is smaller than the observed one. This may be explained by the fact that for low noise level, the assumption of independent variables for the departure and the arrival points of the Brownian motion between two successive steps of the measurement process is not valid. Despite this divergence regarding dispersion, it is important to note that even for low phase noise, the validity of the phase noise estimator is preserved, since both probability distributions exhibit the same mean value which corresponds to the expected value of score S .

This is also visible on the graph of figure a.9. Indeed it can be observed that, on the complete range of injected phase noise, values obtained with simulations follow the linear trend defined by the expected value of the score S obtained with the stochastic model. Regarding dispersion, a good agreement is obtained between the plotted simulation results and the stochastic model for phase noise levels higher than $Q_f/10$, with all values obtained by simulation that fall in the region of high probability defined by the stochastic model. In case of lower level of noise, the dispersion observed with simulation appears higher than the one predicted by the stochastic model.

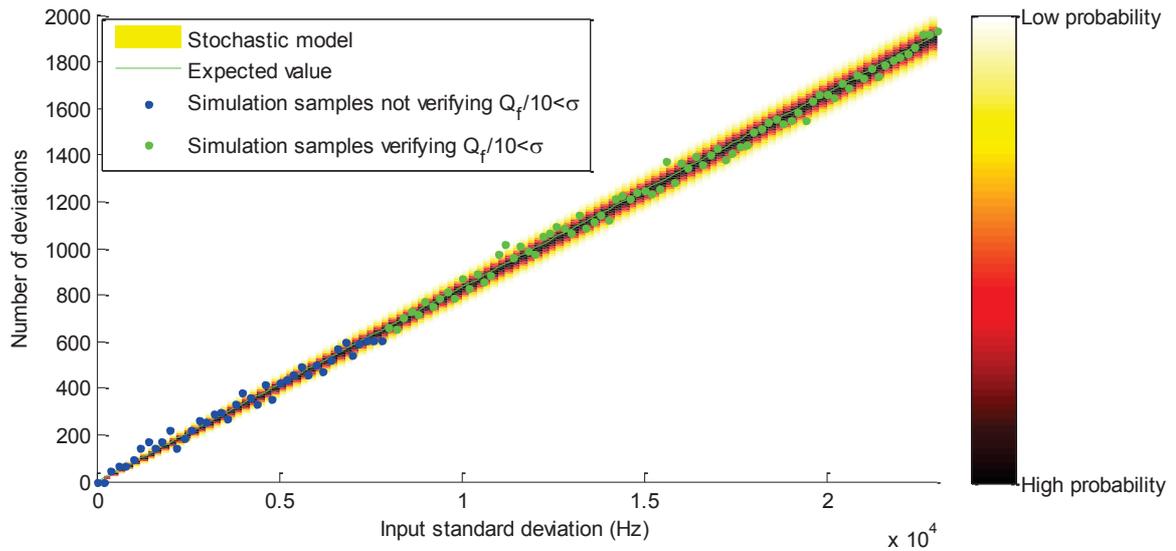


Figure annex.9 Comparison between simulations of the digital post-process and the stochastic model for different levels of noise Probability distribution for deviation with $f = 4\text{MHz}$, $f_s = 204\text{MHz}$, $N_{periods} = 4096$

1.3. Low level noise without filter

For lower noise another model can be developed. It implies that the Brownian motion can only score 0 or 1 point. At these levels of noise, the position at the previous measurement influences if the Brownian motion is going to cross or not a gate. Thus we want to track with care how the Brownian motion behaves in between two measurements.

So we are going to consider the position of the Brownian motion between two measured deviations. The space in which the Brownian motion evolves is broken up in discrete areas. This way the probability of the position of the Brownian motion can be computed at each discrete time, half periods, corresponding to a measurement. A discrete time representation is more realistic than a continuous one in our case because measurements are discrete.

The probability space is thus broken up in L areas called transition states where the Brownian motion is not detected and 2 areas representing one absorbing state as illustrated in figure a.10. The absorbing state is on the outside of the transition states. In the following calculations L is chosen to be 200.

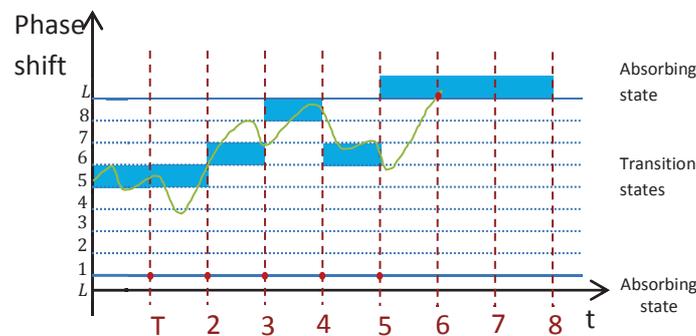


Figure a.10 The Brownian motion evolves in the transition states until it reaches an absorbing state.

Once the Brownian motion reaches an absorbing state, it stays there. So this can be modeled as a Markov chain in figure a.11.

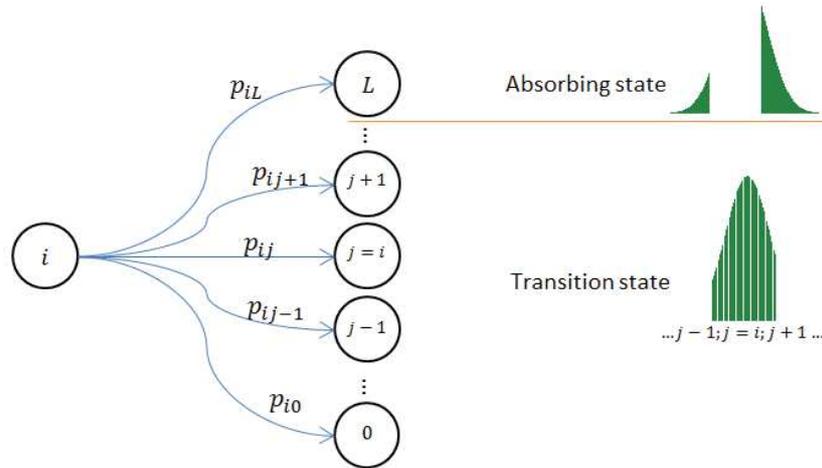


Figure a.11 Markov chain representation and probabilities associated with each transition

With a Markov matrix :

$$P = \begin{bmatrix} p_{ij} & p_{iL} \\ 0 & \dots & 0 & 1 \end{bmatrix} \quad (128)$$

With

$$p_{ij} = \frac{1}{2} \left(\operatorname{erf} \left(\frac{\left(j + \frac{1}{2} - i \right) \frac{Q_f}{L}}{\sqrt{2\sigma^2}} \right) - \operatorname{erf} \left(\frac{\left(j - \frac{1}{2} - i \right) \frac{Q_f}{L}}{\sqrt{2\sigma^2}} \right) \right) \quad (129)$$

$$p_{iL} = 1 - \frac{1}{2} \left(\operatorname{erf} \left(\frac{\frac{Q_f}{2} - i \frac{Q_f}{L}}{\sqrt{2\sigma^2}} \right) - \operatorname{erf} \left(\frac{-\frac{Q_f}{2} - i \frac{Q_f}{L}}{\sqrt{2\sigma^2}} \right) \right) \quad (130)$$

Naming the initial vector $V(0)$ at the very start of the measurement or $W(0)$ after a detection has happened, the distribution vector at the k step will be:

$$V(i) = V(0)P^i \text{ or } W(i) = W(0)P^i \quad (131)$$

Indeed there are two cases for computing $V(i)$ or $W(i)$. Either the Brownian motion is at the beginning of the capture; in this case, the probability distribution over the L transition states is uniform :

$$V(0) = \frac{1}{Q_f} [1 \dots 1 \quad 0] \quad (132)$$

Or the Brownian motion has just triggered a deviation. We make the hypothesis that the probability of crossing a deviation detection line is uniform over time as illustrated figure a.12.

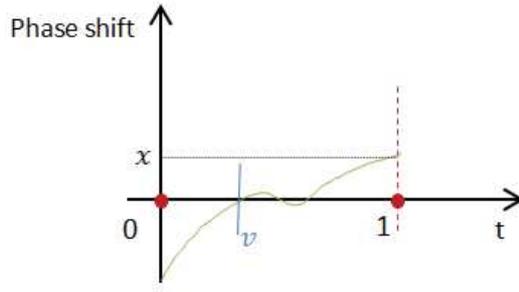


Figure a.12 The Brownian motion can cross a horizontal at a position v between two measurements with v following a uniform distribution between 0 and 1

So we can integrate over time

$$\begin{aligned}
 (W(v^+) = x) &= \int_0^1 \frac{2}{\sigma\sqrt{2\pi(1-v)}} \exp\left(-\frac{x^2}{2\sigma^2(1-v)}\right) dv \\
 &= \frac{2\sqrt{2}}{\sqrt{\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right) + \frac{2x}{\sigma^2} \left(\operatorname{erf}\left(\frac{x}{\sigma\sqrt{2}}\right) - 1\right)
 \end{aligned}
 \tag{133}$$

And the other initial vector $W(0)$ will be :

$$W(0) = \frac{1}{Q_f} \left[p \left(\operatorname{BM}(v^+) = i \frac{Q_f}{L} \right) \quad 0 \right]
 \tag{134}$$

We are only interested in the last element of $V(i)$ or $W(i)$ (but we need all of them to compute it). These last elements $V_L(i)$ or $W_L(i)$ represent the density probability function over time for the Brownian motion to generate a deviation.

We are interested in the number of deviations that are generated during the time of capture. A renewal process can help us with that.

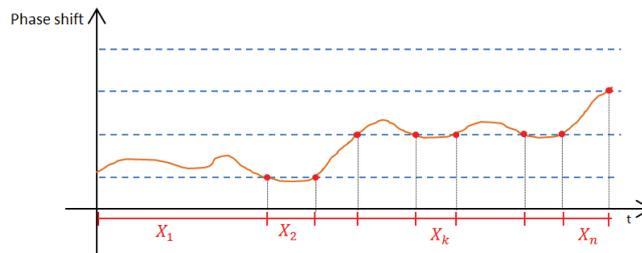


Figure a.13 Illustration for a renewal process

Let X_1 follow the probability distribution defined by V_L and $\{X_k, k \geq 2\}$ follow W_L and let's define the sum:

$$S_n = \sum_{k=1}^n X_k
 \tag{135}$$

What is the probability of having n deviations before the end of the time of capture t ?

$$\pi_{\text{sup}}(n) = p(S_n \leq t)
 \tag{136}$$

This renewal process problem solves its self with a convolution.

$$p(S_n = v) = [p_0 * p_1^{*(n-1)}]_{[0,v]}(v) \quad (137)$$

$$P(S_n \leq t) = \int_0^t p(S_n = v)dv \quad (138)$$

Just as in the previous high noise case, since the two gates work independently but identically, the probability of having a score S as the sum of both types of gates is the result of a convolution.

$$\pi(S) = \sum_{i=0}^S \pi_{N_{périodes}}(S - i) \pi_{N_{périodes}}(i) \quad (139)$$

The first case, no deviation detected at all, is better estimated with the probability which is the hitting time for Brownian motion in a corridor [70]:

$$\begin{aligned} P\{S_0 < t\} = & 2 - \frac{1}{b-a} \left(\sum_{i=-\infty}^{+\infty} 2 \left(i + \frac{1}{2}\right) Q_f \operatorname{erf} \left(\frac{\left(i + \frac{1}{2}\right) Q_f}{\sigma\sqrt{2t}} \right) \right. \\ & - 2iQ_f \operatorname{erf} \left(\frac{iQ_f}{\sigma\sqrt{2t}} \right) \\ & + 2\sigma \sqrt{\frac{2}{\pi}} t \left(\exp \left(-\frac{\left(i + \frac{1}{2}\right)^2 Q_f^2}{2t\sigma^2} \right) \right. \\ & \left. \left. - \exp \left(-\frac{i^2 Q_f^2}{2t\sigma^2} \right) \right) \right) \end{aligned} \quad (140)$$

With $a = 0$ and $b = \frac{Q_f}{2}$

The results have a good general figure but amplitudes of probability density function don't quite match.

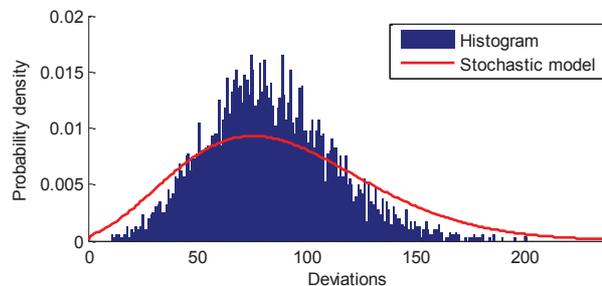


Figure annex.14 Probability distribution of deviation for the low level noise stochastic model case $\sigma = 1kHz, f = 4MHz, f_s = 204MHz, N_{periods} = 4096$

Anyhow, these results have to be taken with great care as jitter or amplitude noise is not taken in account. Such noises can have a great impact in the count of deviations particularly in the low level of

noise case. It is for this purpose that a filter has been developed as described in 3.6. The behavior of the process with this filter at low level noise is analyzed in the next paragraph.

1.4. Low phase noise level with filter

For lower noise another model can be developed. It also implies that the Brownian motion can only trigger 0 or 1 deviations. The Brownian motion has increments distributed according to a Gaussian distribution $\mathcal{N}(0, (2\pi T_s b \sigma)^2)$. The Brownian motion is described in a continuous time manor; that is with a corridor representation rather than gates. The deviations are triggered when the Brownian motion travels $\frac{2\pi}{2n}$ on one side or the other: that is when the Brownian motion hits one wall of the corridor. The corridor is thus $\frac{2\pi}{n}$ rad wide.

Over a time $\frac{T}{2}$ the Brownian motion has an increment distributed as $\mathcal{N}\left(0, \left(2\pi T_s \sqrt{\frac{n}{2}} b \sigma\right)^2\right)$ with $= \frac{f_s}{f}$.

This is equal to $\mathcal{N}\left(0, \left(2\pi T_s \sqrt{\frac{1}{2}} n \sigma\right)^2\right)$ still with a corridor $\frac{2\pi}{n}$ rad wide.

Dividing by $2\pi T_s n$, this is equivalent to a Brownian motion following $\mathcal{N}\left(0, \left(\sigma \sqrt{\frac{1}{2}}\right)^2\right)$ with a corridor

of width : $\frac{2\pi}{n 2\pi T_s n} = \frac{1}{nT} = \frac{f}{f_s/f} = Q_f$

When the Brownian motion hits a side of the corridor, it triggers a deviation. Once the deviation is triggered the process restarts. We seek the number of deviations that are triggered during the time of the measurement.

In figure a.15 bellow, the Brownian motion is represented by the orange line. When the orange line crosses a blue dotted line, a deviation detection is triggered. The Brownian motion can't trigger two consecutive deviation detections from the same line because of the filter design. The consecutive detections by the same blue line are discarded because their stopping time is not relevant to the Brownian motion's characteristics. So the blue lines are separated by $\frac{Q_f}{2}$ but the corridor is Q_f wide.

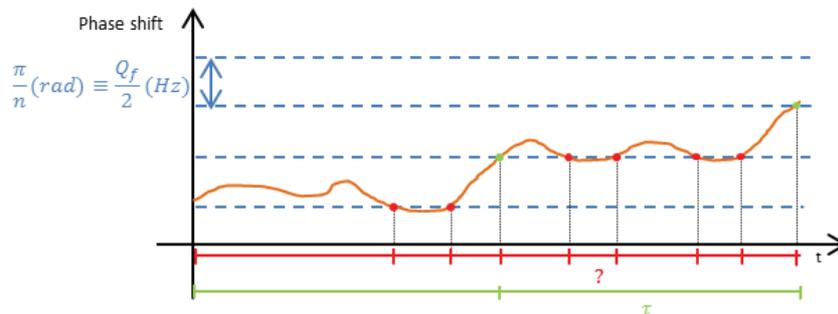


Figure a.15 Illustration of Brownian motion with threshold of detections which can be modeled as corridors.

The hitting times can be calculated knowing that they follow the density probability [70] :

$$p_x\{\tau = t\} = ss_t(b - x, b - a) + ss_t(x - a, b - a) \tag{141}$$

With :

$$ss_v(\alpha, \beta) = \frac{1}{\sigma_p \sqrt{2\pi v^3}} \sum_{i=-\infty}^{+\infty} ((2i+1)\beta - \alpha) \exp\left(-\frac{((2i+1)\beta - \alpha)^2}{2v\sigma_p^2}\right) \quad (142)$$

Setting $b = \frac{Qf}{2}$; $a = -\frac{Qf}{2}$; and $\sigma_p = \sqrt{\frac{1}{2}} \sigma_f$

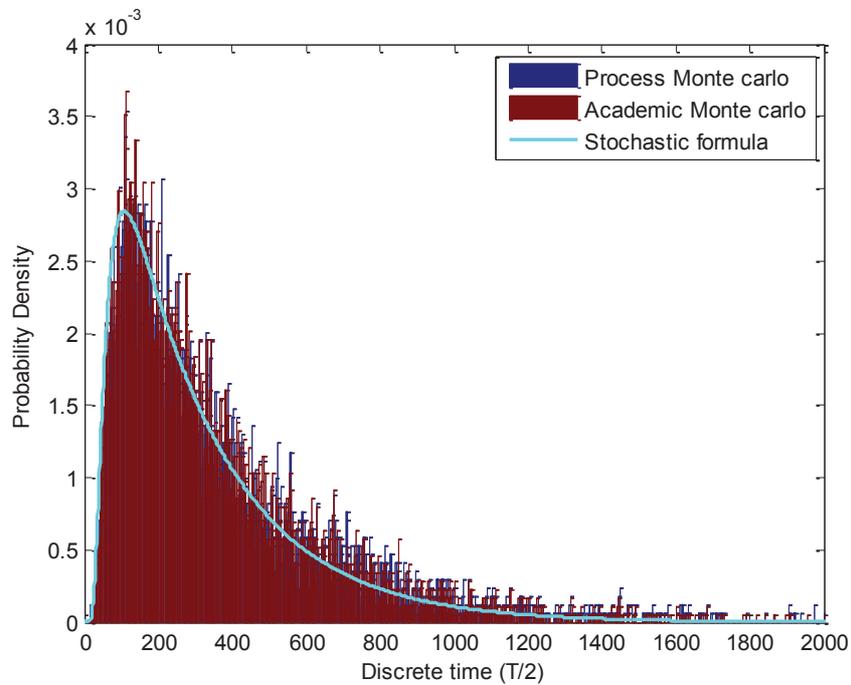


Figure a.16 Stochastic probability density compared to histograms on academic and simulated Process Monte Carlo

The times between detection have been recorded during a measurement simulation and an academic setup for Brownian motion. A histogram has been set up for both sets of results. We can see in the figure a.16 above that the stochastic formula matches the histograms.

From the density probability of hitting time, we can use a renewal process to count the number of detections during the time of experiment.

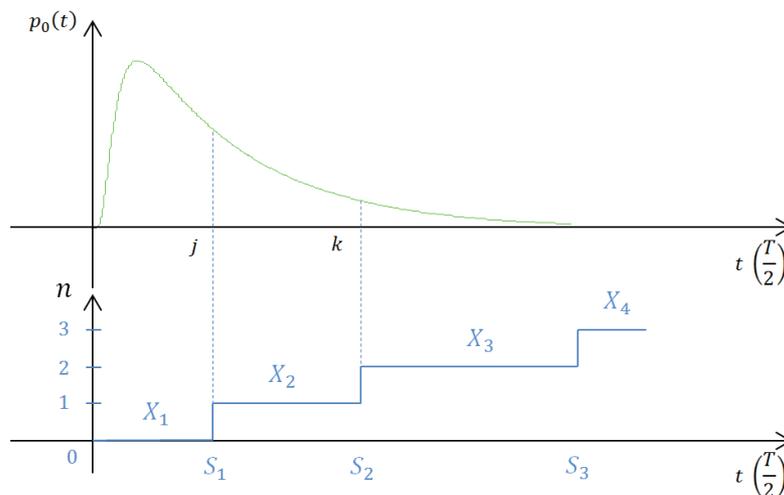


Figure a.17 Illustration of renewal process

Let us denote X_k the hitting time for the k^{th} detection triggered for which the probability density has been described just above. They are all identical.

$$p(X_k = t) = p_0\{\tau = t\} \quad (143)$$

S_n is the sum of the hitting times of n first transitions.

$$S_n = \sum_{k=1}^n X_k \quad (144)$$

The renewal process, illustrated in figure 6.17, tells us that the density probability of S_n is described by a n convolutions over $p(X_k = t)$.

$$p(S_n = t) = p^{*n}(t) \quad (145)$$

Note that the probability of having at least n deviations detected in the time of measurement t_{meas} is the integral of the probability density over the time :

$$P(S_n \leq t_{\text{meas}}) = \int_0^{t_{\text{meas}}} p(S_n = t) dt \quad (146)$$

We seek the number of transition that happened during the time of measurement:

$$N(t_{\text{meas}}) = \sup\{n : S_n \leq t_{\text{meas}}\} \quad (147)$$

The probability of $N(t_{\text{meas}})$ is then described by the probability of having exactly n deviations during the time of measurement t_{meas} :

$$P(N(t_{\text{meas}}) = n) = P(S_n \leq t_{\text{meas}}) - P(S_{n-1} \leq t_{\text{meas}}) \quad (148)$$

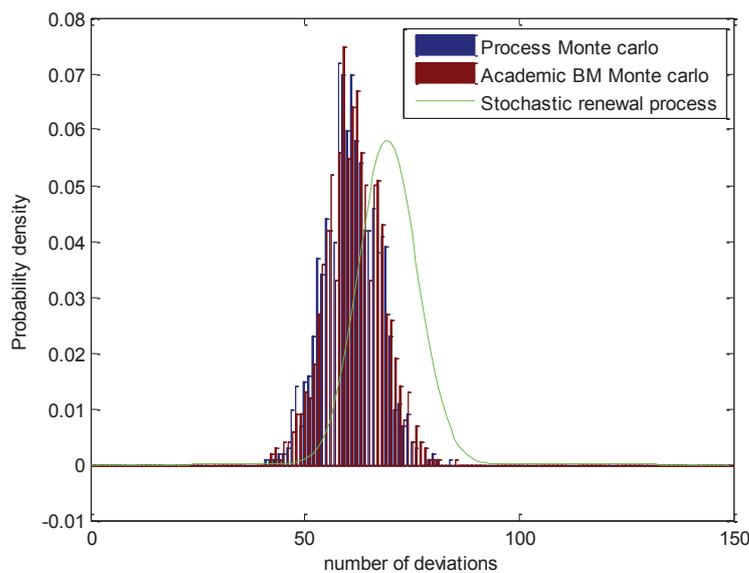


Figure a.18 Stochastic model compared to histograms of simulated process and academic Brownian motion

The process as well as an academic Brownian motion have been run in simulation for $\sigma = 1500\text{Hz}$, $f_s = 250\text{MHz}$, $f = f_s/99$ over a time of measurement corresponding to 5000 periods in figure 6.18. We can see that the stochastic model has the same shape as the histograms of the simulated process. However the stochastic model has a positive offset relative to the Monte-Carlo results. This could be expected as the model is in continuous time whereas the simulation process is discrete. Indeed, detection can only happen every period in simulation whereas the model considers them as continuous hitting time process.

The expected value of the hitting time mechanism is provided in [70]

$$\mathbb{E}_x(T^*) = (x - a)(b - x) \quad (149)$$

In our case $-a = b = \frac{Q_f}{2}$ et $x = 0$

So we have

$$\mathbb{E}(T) = \left(\frac{Q_f}{2}\right)^2 \quad (150)$$

In [70] the formula is normalized so, in our case the expected value is :

$$\mathbb{E}(T) = \left(\frac{Q_f}{2\sigma\sqrt{\frac{1}{2}}}\right)^2 \quad (151)$$

The expected value of the renewal process is then :

$$\mathbb{E}(N_{\text{écarts filtrés}}) = \frac{t_{\text{meas}}}{\mathbb{E}(T)} = \frac{4N_{\text{périodes}}\sigma^2}{Q_f^2} \quad (152)$$

To sum up, we run a calculation over several values of σ of the stochastic model and the expected value as well as the simulation to validate the model.

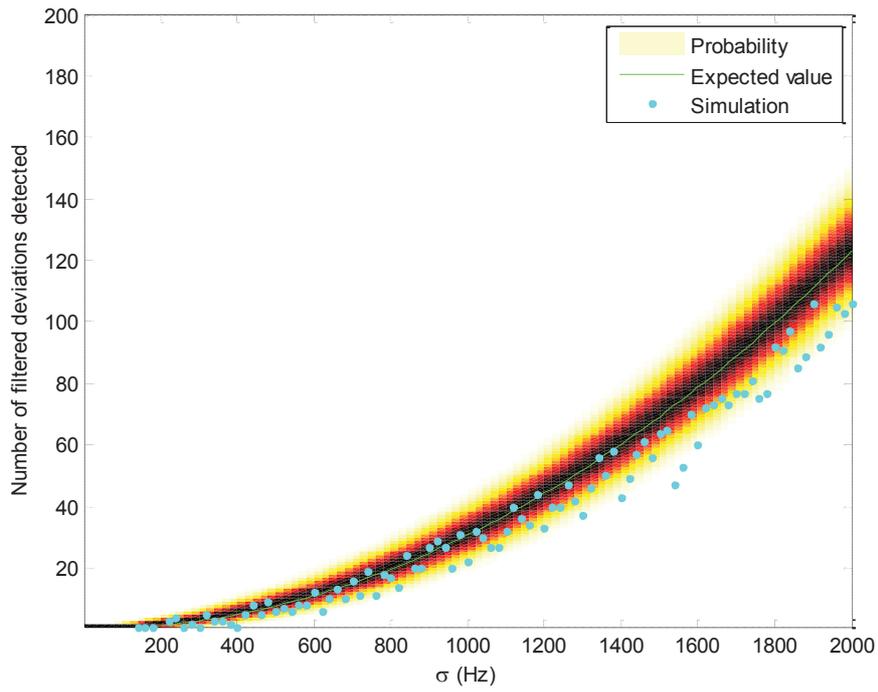


Figure a.19 Stochastic model compared simulated process for different inputs of σ_{fnoise}

The same remark apply in figure a.19, the expected value is overestimated due to the continuous time model which differ from the simulated process.

2. Algorithms

2.1. Signal Generation In simulation in Matlab

```
function Signal = SignalGeneration(fs,f,Nperiods,sigmaf)
%=====
%Signal Generation
%=====
%.....
% Parameters
%.....
L=floor(Nperiods*fs/f);
Ts=1/fs;
w=2*pi*f;

%.....
% standardizing coefficient
%.....
b=sqrt(fs/f);

%.....
% Gaussian random variables
%.....
va=randn(L,1)*sigmaf;

%.....
% Frequency noise
%.....
GaussianFrequencyNoise=va*2*pi()*b;

%-----
% Integration of frequency noise
%-----
%.....
% Integration
%.....
t=(1:L)*Ts;
t0=t;
PhaseNoiseBrownian=Ts*cumsum(GaussianFrequencyNoise);
phase=w*t0';
NoisyPhase=phase+PhaseNoiseBrownian;

%.....
% Initial phase
%.....
InitialPhase=2*pi()* (rand-0.5);

%.....
% Signal affected with phase noise
%.....
Signal=sin(NoisyPhase+InitialPhase);
```

2.2. Allan Variance Measurement In simulation in Matlab

```
function [EstimatedSigmaaf]= AllanDevMeasurement (Signal,fs,f,Navg,K)

Ts=1/fs;
%=====
%Comparator
%=====
%.....
% Comparator threshold
%.....
S = 0;
%.....
% Comparison
%.....
BinaryVector=NaN(size (Signal));
BinaryVector (Signal>=S)=1;
BinaryVector (Signal<S)=0;
%.....
% Running average on binary vector
%.....
RunAvgBinaryVect=tsmovavg (BinaryVector','s',Navg);
RunAvgBinaryVect=[RunAvgBinaryVect (round (Navg/2) :end) NaN (1, round (Navg/2) -
1)];
%.....
% Clean binary vector
%.....
BinaryVector (RunAvgBinaryVect>0.5)=1;
BinaryVector (RunAvgBinaryVect<0.5)=0;
BinaryVector (RunAvgBinaryVect==0.5)=BinaryVector (find (RunAvgBinaryVect==0.5
)-1);
%=====
% Extremums' position
%=====
%.....
% Zero crossing identification
%.....
StateChangeIdentification=diff (BinaryVector);
StateChangePosition=find (StateChangeIdentification);
%.....
% Extremum position
%.....
ExtremumsPosition=tsmovavg (StateChangePosition, 's',2,1)';

%=====
% Allan Deviation computation
%=====
W=K;
%-----
% Reconstruction
%-----
%.....
% Pase shift reconstruction
%.....
EstimatedPhase=pi () * (0:length (ExtremumsPosition));
EstimatedPhaseShift=EstimatedPhase (3:end) -
ExtremumsPosition (2:end) *Ts*2*pi () *f;

%.....
% Running average on phase shift
%.....
```

```

MovAvgEstimatedPhaseShift=tsmovavg (EstimatedPhaseShift, 's', W);
MovAvgEstimatedPhaseShift1=MovAvgEstimatedPhaseShift ((W+1+floor (rand () *K)) :
K:end);

%.....
% Increments
%.....
DeltaDeltaMovAvgEstimatedPhaseShift=diff (diff (MovAvgEstimatedPhaseShift1));
DeltaDeltaMovAvgEstimatedPhaseShift=DeltaDeltaMovAvgEstimatedPhaseShift (1:2
:end);
M=length (DeltaDeltaMovAvgEstimatedPhaseShift);

%.....
% Standard/Allan deviation
%.....
EstimatedSigmaf=sqrt (sum (DeltaDeltaMovAvgEstimatedPhaseShift.^2) / (M));
bias=1-1/ (2*K) * (W-1/W);

bias (bias<=0)=NaN;
EstimatedSigmaf=f/ (2*pi ()) *EstimatedSigmaf/sqrt (K*bias);

end

```

2.3. Simulation Example in Matlab

```

Signal = SignalGeneration (200e6, 1.3125e6, 15000, 500);
EstimatedSigmaf = AllanDevMeasurement (Signal, 200e6, 1.3125e6, 5, 30);

disp (EstimatedSigmaf)

```

2.4. Integrated module in Verilog

```

module SigmaEstimation(
  //// output
  varEstimateddraw, finished, FlagRavg, M,
  // Debug outputs
  op1, op2, op3, op4, op5, unknownValDetected, signalintest,
  N, W, // param input
  signal, // signal Input
  reset,
  activate,
  clk      // clock Input
);

//-----Input Ports-----
  input signal, reset, activate, clk;
  input [7:0] W;
  input [7:0] N;

  wire [7:0] K = W[7:0];
//-----Parameters-----
  parameter Navg = 5; // up to 15
  parameter Nbits = 12;
  parameter Nclkbases = 625000; //6550;

//----- Outputs and variables -----

```

```

reg [8:0] K2 = 30; //2*K
reg [40:0] Nclk = 1 * Nclkbases;

reg [40:0] counterclk=0;
output reg finished=0;

reg [3:0] Ravg = 0 ;
reg [4:0] Ravg2 = 0 ;
reg [(Navg-1):0] memNavg = 0;
reg FlagRavgPrev = 0;
output reg FlagRavg = 0;
output reg op1 = 0;
output reg op2 = 0;
output reg op3 = 0;
output reg op4 = 0;
output reg op5 = 0;
output reg unknownValDetected = 0;
output reg signalintest=0;
reg signalVerfied=0;
reg signalPrev=0;
reg [3:0] iAvg = 0;
reg [Nbits:0] t = 0;
reg [Nbits:0] ttransitionprev = 0;
reg [Nbits:0] textremum2 = 0;
reg [Nbits:0] RunSumPhaseShift = 0;
reg [Nbits:0] RunSumPhaseShiftTemp = 0;
reg [10:0] iW = 0;
reg [10:0] PhaseCount=0;
reg [Nbits:0] MemPhaseShift [256:0];
reg [Nbits:0] MemPhaseShiftTemp = 0;
reg [Nbits:0] PhaseShift0 = 0;
reg [Nbits:0] PhaseShiftK = 0;
reg signed [Nbits:0] DeltaK = 0;
reg signed [Nbits:0] Delta2K = 0;
reg signed [Nbits:0] DeltaDeltaPhaseShift = 0;
reg [(Nbits*2):0] DeltaDeltaPhaseShiftSquared = 0;
output reg [23:0] M = 0;
reg [10:0] jK = 0;
reg FlagjK = 0;
reg FlagPhaseCount=0;
reg cond0=0;
reg cond1=0;
reg cond2=0;
reg cond3=0;
output reg [47:0] varEstimateddraw = 0;

//-----Code Starts Here-----
always @ (posedge clk)

//-----
// Reset all variables
//-----
if (reset) begin
    counterclk<=0;
    signalVerfied<=0;
    signalPrev<=0;
    finished<=0;
    Ravg <= 0 ;

```

```

Ravg2 <=0 ;
memNavg <= 0;
FlagRavgPrev <= 0;
FlagRavg <= 0;
op1 <= 0;
op2 <= 0;
op3 <= 0;
op4 <= 0;
op5 <= 0;
unknownValDetected <= 0;
signalintest<=0;
iAvg <= 0;
t <= 0;
ttransitionprev <= 0;
textremum2 <= 0;
PhaseCount <= 0;
RunSumPhaseShift <= 0;
MemPhaseShiftTemp <=0;
RunSumPhaseShiftTemp <= 0;
iW <= 0;
PhaseShift0 <= 0;
PhaseShiftK <= 0;
DeltaK <= 0;
Delta2K <= 0;
K2 <= K*2;
Nclk <= N * Nclkbase;
DeltaDeltaPhaseShift <= 0;
DeltaDeltaPhaseShiftSquared <= 0;
M <= 0;
jK <= 0;
FlagjK <= 0;
FlagPhaseCount<=0;
varEstimateddraw <= 0;
cond0<=1;
cond1<=0;
cond2<=0;
cond3<=0;

// If the clock counter is still inferior to the number of specified in
Nclk and the process is being asked to be activated
end else if (cond0 && activate) begin
    // then the process hasn't finished yet
    finished<=0;
    // we increment th clock counter
    counterclk<=counterclk+1;
    cond0 <= counterclk < Nclk;
    // Buffer signal
    signalPrev<=signal;
    // Verification of signal integrity
    if (signalPrev===1'b1 || signalPrev===1'b0) begin
        signalVerfied<=signalPrev;
        unknownValDetected <= 0;
    end else begin
        unknownValDetected <= 1;
    end
    // Output for debugging
    signalintest<=signalPrev;

    // Running average
    Ravg <= Ravg + signalVerfied - memNavg[iAvg];
    Ravg2 <= 2*Ravg;

```

```

// the signal is kept in memory
memNavg[iAvg] <= signalVerified;

// then the memory index is incremented
if ((iAvg+1)==Navg) begin
    iAvg <= 0;
end else begin
    iAvg <= iAvg +1;
end
// time is incremented
t<=t+1;

// Running average is compared to Navg
if (Ravg2<Navg) begin
    FlagRavg <= 0;
end else if (Ravg2>Navg) begin
    FlagRavg <= 1;
end
// If state change then count a new semi-period and record time
FlagRavgPrev<=FlagRavg;
if (FlagRavgPrev!=FlagRavg) begin
    ttransitionprev <= t;
    PhaseCount <= PhaseCount+1;
    // Compute time of extremum
    textremum2 <= t+ttransitionprev; //coef
2, this should be divided by 2; t is time of transition (also actual time),
ttransitionprev is the time of previous transition
    // Indicate operation 1 is finished
    op1 <= 1;
end
// if operation 1 is finished, move on to next operation
if (op1) begin
    // reset op1
    op1 <= 0;

    // three conditions :
    // phase count hasn't reached window length (FlagPhaseCount is for
modular arithmetic on phase count)
    cond1 <= PhaseCount > 2 && PhaseCount < (W+3) && ~FlagPhaseCount;
    // phase count has just reached window length
    cond2 <= PhaseCount == (W+3) && ~FlagPhaseCount;
    // phase count has reached window length
    cond3 <= PhaseCount >= (W+4) || FlagPhaseCount;
    // Sum useful for running average on phase shift
    RunSumPhaseShiftTemp <= RunSumPhaseShift + textremum2;
    // get from memory for running average
    MemPhaseShiftTemp <= -MemPhaseShift[iW];
    // indicate operation 2 has finisher
    op2 <= 1;
end
// if operation 2 has finished move on to next operation
if (op2) begin
    // reset op2
    op2 <= 0;
    // sote in memory for running average
    MemPhaseShift[iW] <= textremum2;
    // counter for running average and memory index
    if ((iW+1)==W) begin
        iW <= 0;
    end else begin

```

```

iW <= iW + 1;
end

// RUNNING AVERAGE ON PHASE SHIFT
// if phase count hasn't reached window length
if (cond1) begin
// just sum (sum is done in op1)
RunSumPhaseShift <= RunSumPhaseShiftTemp;
// if phase count has just reached window length
end else if (cond2) begin
//sum and subtract with value in memory
RunSumPhaseShift <= RunSumPhaseShiftTemp + MemPhaseShiftTemp;
// if phase count has reached window length
end else if (cond3) begin
//sum and subtract with value in memory
RunSumPhaseShift <= RunSumPhaseShiftTemp + MemPhaseShiftTemp;
// raise flag that we have reached W phase count at laest once
FlagPhaseCount<=1;
// jK is index for Allan variance increment count
// if equal to 0 or K then store the value
if (jK==K) begin
PhaseShiftK <= RunSumPhaseShift;
end else if (jK==0) begin
// FlagjK is raised the very first time we come through
here, for not to get confused relative to 0 = 2K
if (FlagjK) begin
// compute increments
Delta2K <= RunSumPhaseShift - PhaseShiftK;
DeltaK <= PhaseShiftK - PhaseShift0;
// indicate operation 3 has finished
op3 <= 1;
end
FlagjK <= 1;
PhaseShift0 <= RunSumPhaseShift;
end
// counter for increments separated by K for Allan variance
if ((jK+1)==(K2)) begin
jK <= 0;
end else begin
jK <= jK +1;
end
end
end
// if operation 3 has finished
if (op3) begin
// reset op3
op3 <= 0;
// Difference between increments
DeltaDeltaPhaseShift <= Delta2K - DeltaK;
op4 <= 1;
end
// if operation 4 has finished
if (op4) begin
op4 <= 0;
// square
DeltaDeltaPhaseShiftSquared <=
DeltaDeltaPhaseShift*DeltaDeltaPhaseShift;
op5 <= 1;
end
// if operation 5 has finished
if (op5) begin

```

```
    op5 <= 0;
    // Allan variance result = SIG, relative to Eq 110
    varEstimateddraw <= DeltaDeltaPhaseShiftSquared + varEstimateddraw;
    M <= M+1;
    end
end else begin
    finished<=1;
end

endmodule
```

Abstract

In recent decades, the microelectronics industry has experienced a wide democratization of the use of telecommunication applications. The improved process design and manufacturing have produced complex and high performance analog, mixed and radio frequency circuits for these applications. However, the test cost of these integrated circuits still represents a large part of the manufacturing cost. Indeed, very often, analog testing is not just a functional test but needs measurements for specification validations. These measurements require the use of dedicated instruments expensive resources on standard industrial test equipment.

One of the essential but costly specifications to validate in RF circuitry is the phase noise level. The currently used industrial technique consists in capturing the signal from the circuit under test using an RF tester channel equipped with a high performance analog to digital converter; a Fourier transform is then applied to the digitized signal and the phase noise is measured on the resulting spectrum.

The approach proposed in this thesis is to achieve the phase noise measurement using solely digital low-cost resources. The basic idea is to perform 1-bit capture of the analog signal with a standard digital channel and develop post-processing algorithms dedicated for phase noise evaluation from the zero-crossings of the signal.

Two methods are presented. The first method is based on an estimate of the instantaneous signal frequency and an analysis of their dispersion induced by phase noise. This method imposes a strong constraint on the sampling frequency to be used and proved to be sensitive to noise amplitude, limiting the range of possible measures. A second method is then proposed to overcome these limitations. From the binary capture of the analog signal, a reconstruction of the instantaneous phase of the signal is carried out, then filtered and characterized by a common tool of frequency stability assessment: the Allan variance. This technique, robust to amplitude noise and jitter, can be parametrized and enables efficient characterization of phase noise without fundamental constraint.

In addition to the simulations, these techniques are subject to a stochastic study and are validated experimentally on different types of signals to be measured - artificially generated or from chips on the market - and with different measuring instruments - on oscilloscope or industrial tester, in laboratory and on a production line-. An On-chip implementation is also proposed and validated with a FPGA prototype.

Keywords: Analog and radio frequency integrated circuits, Phase noise test, Allan Variance, Built-in-self-test

Résumé

Au cours des dernières décennies, l'industrie de la micro-électronique a connu une large démocratisation de l'utilisation des applications de télécommunication. L'amélioration continue des procédés de conception et de fabrication ont permis de produire des circuits analogiques, mixtes et radiofréquences complexes et haute performances pour ces applications. Toutefois, le coût de production de ces circuits intégrés est dominé par le coût de leur test en fin de chaîne. En effet très souvent tester des fonctions analogique ne se résume pas à un test fonctionnel mais signifie mesurer les spécifications du circuit.

Une des spécifications essentielle mais couteuse à caractériser pour les circuits RF est son niveau de bruit de phase. En effet les mesures des paramètres analogiques et radiofréquences nécessitent généralement des instruments couteux et sophistiqués. La technique actuellement utilisée en industrie consiste à capturer le signal à l'aide des ressources analogiques onéreuses de l'équipement de test automatisé. Celui-ci est équipé de convertisseurs analogique-digital hautes performances qui permettent d'évaluer le bruit de phase sur la transformée de Fourier du signal.

La solution proposée dans cette thèse n'utilise que des ressources digitales faibles coût. Le travail présenté montre qu'une évaluation des temps de passages à zéro du signal dans le domaine temporel grâce à un simple comparateur permet une mesure du bruit de phase.

Deux méthodes sont présentées. La première méthode est basée sur l'évaluation de l'écart absolu moyen d'une population de mesures de la fréquence instantanée du signal. Bien que validée en simulation, la méthode présente une contrainte forte quant à la fréquence d'échantillonnage à utiliser et s'est révélée ne pas être robuste au bruit d'amplitude lors de l'évaluation expérimentale.

Une seconde méthode est alors proposée afin d'en résoudre. La phase du signal est reconstituée grâce la capture du vecteur binaire avant d'être filtrée puis caractérisée grâce à un outil usuel d'évaluation de stabilité fréquentiel : la variance d'Allan. Cette technique, robuste au bruit d'amplitude et au jitter, peut être paramétrée et permet de caractériser les signaux analogiques efficacement et sans contrainte fondamentales.

En plus des simulations cette technique fait l'objet d'une étude stochastique et est validée expérimentalement en plusieurs configurations : avec différents types de signaux à mesurer – généré artificiellement ou provenant de puces sur le marché – ou avec différentes conditions mesures – sur oscilloscope ou sur testeur, en laboratoire et en production –. Une implémentation sur puce est aussi proposée avec une validation avec un prototype FPGA.

Mots clés : Circuits analogiques et radio fréquence, Test du bruit de phase, Variance d'Allan, Test sur puce