ABSTRACT

PARIKH, NEHAL NILESH. Effect of Nonlinear System with Time Delayed Feedback Generating Flicker Noise in Oscillators. (Under the direction of Dr. Michael Steer.)

Oscillator phase noise is presented for a simple microwave Colpitts oscillator. This model consists of a nonlinear system with time delayed feedback. It is found that the origin of oscillator phase noise is chaos which derives from the dynamics of nonlinear system with time delayed feedback. Such a system produces long term memory effects. The dynamics of chaos are evident as a long duration autocorrelation characteristic. Baseband white noise affects the dynamics of chaos presenting what is analogous to upconversion of baseband white noise to the oscillation frequency.

This approach presents a first attempt at using quad precision to simulate phase noise in oscillators. This places minimal error and high accuracy on the magnitude and the nature of noise present in a circuit. Such an effort captures the effect of nonlinear interactions between signal and noise. This work introduces associate discrete modeling, a novel technique for modeling of discrete elements in circuit simulators. This technique is implemented using Finite difference time domain method of numerical simulation. Such an implementation has the advantage to simulate as a standalone C++ code, instead of incorporating with existing circuit simulators. Backward euler form is employed for solving numerical integrations.

Even though, there are no noise sources incorporated in the oscillator, results show clearly differentiated phase noise regions with slopes of $1/\Delta f$, $1/\Delta f^3$, where $\Delta f$ is the frequency offset from the carrier. These slopes suggest presence of flicker noise in oscillators. Addition of baseband white noise causes $1/\Delta f^2$ slope to be added to the spectrum. Contrary to the conventional thinking that this represents up conversion, the baseband noise affects the dynamics of chaos.

As has been experimentally observed, phase noise is affected by the oscillator $Q$. Increase in quality factor, causes a decrease in the level of phase noise of the oscillator. The delay added to the oscillator, causes an increase in the long term memory of the oscillator output. The magnitude of delay governs the intermittency of the system. An increase in delay causes the phase noise level to decrease. The level of baseband white noise added to the system causes the magnitude of phase noise to increase proportionately.

This research presents a fundamental simulation of phase noise in a microwave oscillator at close to first principals. This work is a step in developing a complete understanding of phase noise in oscillators. This understanding would help to predict the effect of phase noise on the noise of the microwave system, mitigate the negative effects generated by it and in the process improve the performance of the microwave system.
Effect of Nonlinear System with Time Delayed Feedback Generating Flicker Noise in Oscillators

by
Nehal Nilesh Parikh

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Electrical Engineering

Raleigh, North Carolina

2015

APPROVED BY:

__________________________  ____________________________
Dr. Jacob Adams            Dr. Mohammed Zikry

__________________________
Dr. Michael Steer
Chair of Advisory Committee
DEDICATION

To my mother Dharmistha Parikh, my father Nilesh Parikh and my mentors, Jenyl Shah and Bijal Shah who provided me with the opportunity to make something of my life.

To my fate, for allowing it to happen.
BIOGRAPHY

Nehal Parikh was born in the city of Mumbai, India in 1990. He received his Bachelor of Engineering degree in Electronics and Telecommunication from University of Mumbai in 2012. In Fall 2012, he joined Master of Science program in Electrical Engineering at North Carolina State University. His research interests include analog circuit design for microwave systems. His other interests include cricket, astrophysics, music, healthy living and contributing to humanity.
ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor, Dr. Michael Steer, for his support, patience, and encouragement throughout my research. His technical and editorial advice was essential to the completion of this thesis and has taught me innumerable lessons and insights on the workings of academic research in general. Above all and the most needed, he provided me with unflinching encouragement and support in various ways. His truly scientist intuition has made him a constant oasis of ideas and passions in science, which exceptionally inspire and enrich my growth as a student, a researcher and a scientist want to be. I am highly indebted to him.

I would also like to thank Dr. Jacob Adams, Dr. Mohammed Zikry and Dr. M. B. Steer for serving on my committee.

My parents receive my deepest gratitude and love for their dedication and the many years of support during my undergraduate studies that provided the foundation for this work. I would like to thank my mentors Jenyl Shah and Bijal Shah, for not losing hope and helping me through difficult times. Words fail me to express my appreciation to my friend Tanwee Gandhi for constantly being my support and encouragement.
TABLE OF CONTENTS

LIST OF TABLES ......................................................................................... vii

LIST OF FIGURES ....................................................................................... viii

CHAPTER 1 INTRODUCTION ........................................................................ 1
  1.1 Objectives ......................................................................................... 1
  1.2 Motivation ......................................................................................... 1
    1.2.1 Treatment of Flicker noise is oscillators ........................................ 3
  1.3 Original Contributions ...................................................................... 5
  1.4 Overview ......................................................................................... 5
  1.5 Summary ......................................................................................... 6

CHAPTER 2 LITERATURE REVIEW ................................................................. 7
  2.1 Introduction ..................................................................................... 7
  2.2 Thermal Noise ................................................................................. 7
  2.3 Shot Noise ....................................................................................... 10
  2.4 Flicker Noise ................................................................................... 12
    2.4.1 A $1/f^\alpha$ response deriving from the superposition of relaxation processes ......................................................... 14
    2.4.2 Transmission line model of $1/f$ noise ........................................ 15
    2.4.3 Noise in diffusion processes ...................................................... 16
  2.5 Phase Noise ..................................................................................... 17
    2.5.1 Importance of phase noise ......................................................... 21
    2.5.2 Observations of Oscillator noise in the Frequency Domain .......... 23
    2.5.3 Leeson Model ........................................................................... 24
    2.5.4 Hajimiri and Lee Model ............................................................. 30
    2.5.5 Unified theory of oscillator phase noise .................................... 37
  2.6 Summary ......................................................................................... 40

CHAPTER 3 DESCRIPTION OF SIMULATION ............................................ 41
  3.1 Introduction ..................................................................................... 41
  3.2 Architecture .................................................................................... 42
    3.2.1 Introduction ............................................................................ 42
    3.2.2 Preliminaries ........................................................................... 42
    3.2.3 Colpitts Oscillator .................................................................. 43
    3.2.4 Summary ................................................................................ 46
  3.3 Implementation-Intelligent Model ..................................................... 47
    3.3.1 Introduction ............................................................................ 47
    3.3.2 Numerical Integration of Ordinary Differential Equations ........ 50
    3.3.3 Associated Discrete Modelling using the Backward Euler form .... 51
    3.3.4 Summary ................................................................................ 53
  3.4 Oscillator modeling ......................................................................... 53
    3.4.1 Introduction ............................................................................ 53
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table B.1</th>
<th>Data files-Part 1</th>
<th>114</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table B.2</td>
<td>Data files-Part 2</td>
<td>115</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1.1</td>
<td>Measured phase noise of a 50 MHz BJT varactor-based VCO[32]. Three phase noise regions are identified as $f^{-3}$ (having a slope of $-9$ dB/octave), $f^{-2}$ (having a slope of $-6$ dB/octave), and $f^{-1}$ (having a slope of $-3$ dB/octave) Figure is taken from [32].</td>
</tr>
<tr>
<td>Figure 2.1</td>
<td>Thermal noise equivalent circuits. Figure is reproduced from [31].</td>
</tr>
<tr>
<td>Figure 2.2</td>
<td>Lumped RC transmission line excited by a white noise current source. Figure is reproduced from [29].</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>1D Brownian motion realization. Figure is taken from [16].</td>
</tr>
<tr>
<td>Figure 2.4</td>
<td>Uncertainty in carrier frequency due to phase noise.</td>
</tr>
<tr>
<td>Figure 2.5</td>
<td>Components of noise.</td>
</tr>
<tr>
<td>Figure 2.6</td>
<td>Varactor tuned VCO. Figure is taken from [38].</td>
</tr>
<tr>
<td>Figure 2.7</td>
<td>Transmitter. Figure is taken from [38].</td>
</tr>
<tr>
<td>Figure 2.8</td>
<td>Receiver interference.</td>
</tr>
<tr>
<td>Figure 2.9</td>
<td>Constellation Diagram. Figure is taken from [1].</td>
</tr>
<tr>
<td>Figure 2.10</td>
<td>Measured phase noise of low-frequency oscillators: (a) instrument noise floor; (b) HP 5087A frequency distribution amplifier at 5 MHz (used to drive the external reference input of several test instruments using a single high-quality oscillator); (c) TADD-1 frequency distribution amplifier at 10 MHz; (d) TADD-1 frequency distribution amplifier at 5 MHz; (e) Spectracom 8140T frequency distribution amplifier at 10 MHz. Five phase noise regions are identified as $f^{-5}$, $f^{-4}$, $f^{-3}$, $f^{-1}$, and white noise. The spurious signals are related to injected harmonics of the 60 Hz power mains [56].</td>
</tr>
<tr>
<td>Figure 2.11</td>
<td>A typical plot of the phase noise of an oscillator having low Q versus offset from the carrier. Figure is taken from [35].</td>
</tr>
<tr>
<td>Figure 2.12</td>
<td>Negative Resistance model. Figure is taken from [38].</td>
</tr>
<tr>
<td>Figure 2.13</td>
<td>Equivalent circuit.</td>
</tr>
<tr>
<td>Figure 2.14</td>
<td>(a) Impulse Injected at the peak, (b) Impulse injected at zero crossing, and (c) effect of nonlinearity on amplitude and phase of the oscillator in state-space. Figure is taken from [21].</td>
</tr>
<tr>
<td>Figure 2.15</td>
<td>Impulse injection at different time instants. Figure is taken from [38].</td>
</tr>
<tr>
<td>Figure 2.16</td>
<td>ISF for different VCO output waveforms. Figure is taken from [38].</td>
</tr>
<tr>
<td>Figure 2.17</td>
<td>Conversion of noise to phase fluctuations and phase-noise sidebands. Figure is taken from [21].</td>
</tr>
<tr>
<td>Figure 2.18</td>
<td>Convolving in Frequency domain. Figure is taken from [38].</td>
</tr>
<tr>
<td>Figure 2.19</td>
<td>Frequency contributions. Figure is taken from [38].</td>
</tr>
<tr>
<td>Figure 2.20</td>
<td>Vector-based description of an oscillator’s field under noise perturbation. Figure is taken from [36].</td>
</tr>
<tr>
<td>Figure 3.1</td>
<td>Block model. Figure is reproduced from [44].</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>Basic oscillator.</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>Colpitts oscillator.</td>
</tr>
</tbody>
</table>
Figure 3.4 Hyperbolic tangent function. Figure plotted in Maple 18. .......................... 45
Figure 3.5 Oscillation build up due to tanh function. .................................................. 46
Figure 3.6 Nodal analysis of a generic circuit. ............................................................... 48
Figure 3.7 Associated Discrete Model for a Conductance .......................................... 49
Figure 3.8 Associated discrete model for a linear capacitor based on the BE form. ........ 52
Figure 3.9 Associated discrete model for a linear inductor based on the BE form. ......... 53
Figure 3.10 Discretized Oscillator model using associate discrete modeling. ................. 54
Figure 3.11 Transient start up for Oscillator. ............................................................... 56
Figure 3.12 Transient start up for Oscillator. ............................................................... 59
Figure 4.1 Autocorrelation for normally distributed white noise. There is an impulse of amplitude 1 at lag=0. ................................................................. 61
Figure 4.2 Autocorrelation of the oscillator output for delay, $\tau = 100$ ps (time step is set to 2.5 ps). ................................................................. 62
Figure 4.3 Autocorrelation of the oscillator output for zero delay, $\tau = 0$ (time step is set to 2.5 ps). ................................................................. 62
Figure 4.4 Phase noise including transient effects. The phase noise at 100 Hz offset from the carrier is 9.8 dB below the level of the carrier ($Q=400$, time step = 2.5 ps). ... 65
Figure 4.5 Number of cycles skipped to minimize error due to transient. .................... 66
Figure 4.6 Phase noise of Colpitts oscillator with $Q = 15$ noiseless circuit elements ($L = 28$ nH, $C_1 = 9.3$ pF and $C_2 = 1$ pF). 0 db corresponds to -9.8 dBc. ................. 66
Figure 4.7 Phase noise implemented using double precision. 0 db corresponds to -9.8 dBc. 69
Figure 4.8 Phase noise with SNR ................................ ................................. 72
Figure 5.1 Phase noise of oscillator with delay 100 ps and $Q=400$ ($L = 25$ nH, $C_1 = 1$ pF, $C_2 = 1.26$ nF). 0 dB corresponds to -9.8 dBc. ................................. 72
Figure 5.2 Effect of Delay on phase noise, $Q = 400$ ($L = 25$ nH, $C_1 = 1$ pF, $C_2 = 1.26$ nF). ... 73
Figure 5.3 Autocorrelation of the oscillator output with a delay of 200 ps (time step = 2.5 ps). 73
Figure 5.4 Equivalent circuit for the oscillator loop gain with simplifying assumptions, no transistor feed-back and resistive loading across the output terminals only. ... 74
Figure 5.5 Effect of Q on phase noise. 0 dB is considered as -9.8 dBc. ......................... 77
Figure 6.1 Statistical properties of white noise. Figure is taken from [15]. .................... 80
Figure 6.2 Autocorrelation vs PSD of white noise. Figure is reproduced from [3]. ........ 81
Figure 6.3 Simple first order low pass RC filter, $R = 5$ k$\Omega$ and $C = 1$ pF. ............... 82
Figure 6.4 Bode plot and pole-zero plot for RC filter, with $R = 5$ k$\Omega$, $C = 1$ pF (implemented in Maple 18). ......................................................... 83
Figure 6.5 Baseband noise model. Figure is taken from [48]. .................................... 83
Figure 6.6 Baseband discretized model. ................................................................. 84
Figure 6.7 Phase noise with addition of baseband white noise, $Q = 3.7$, carrier-to-noise ratio = 100 dB ($L = 30.4$ nH, $C_1 = 1$ pF and $C_2 = 10$ pF). 0 dB is considered as -9.8 dBc. ......................................................... 85
Figure 6.8 Phase noise with $Q = 17.8$, with and without white noise (SNR = 88 dB). 0 dB is considered as -9.8 dBc. ......................................................... 85
Figure 6.9 Phase noise with $Q = 17.8$ and SNR = 88 dB. 0 dB is considered as -9.8 dBc. ... 87
Figure 6.10 Phase noise with SNR = 88 dB, $Q=15,17.8$ and 20. 0 dB is considered as -9.8 dBc. 88
1.1 Objectives

In this thesis, simulation of oscillator phase noise is reported. A simple model of an oscillator is analyzed in time and the frequency domain. The phase noise characteristics of the oscillator are calculated.

The purpose of the research described is to present a very simple model of an oscillator that presents the observed characteristics of flicker noise. The work demonstrates that flicker noise originates from a very simple process: time-delayed feedback of a weakly nonlinear system. The same process that describes chaos and long memory effects.

1.2 Motivation

Noise in electrical circuits presents a practical limit on the performance of electrical circuits and microwave systems. The sources of noise are numerous and in most cases, their origins are well known and several models for these sources exist in the literature. In all cases, the models for representing noise are random in nature and the only characterization of these sources of noise is statistical. Noise sources can be broadly classified as thermal, shot and flicker type. Thermal
noise is associated with the random motion of carriers in a material and the extent of the motion is proportional to the resistance of the material and its temperature. Shot noise is generally found in junction semiconductors, although it was originally observed in vacuum tubes, its existence is attributed to the motion of charges across a junction formed by joining two semiconductor materials with opposite charge concentrations.

The origin of flicker noise, also referred to as $1/\Delta f$ noise where $\Delta f$ is the frequency offset from the carrier, is still somewhat ambiguous in the sense that there is no general agreement on its sources. Flicker noise is a more general form of power law noise or a $1/\Delta f^\alpha$ noise where $\alpha$ is considered to have discrete integer values between 0 and 5. Its various manifestations are found not just in electrical circuits but also in a wide spectrum of scientific situations which is perhaps the biggest reason for there being no unified explanation on the origins of such a noise source. Examples of flicker noise that have been observed includes vacuum tubes [52], height of the floods of the river Nile [18], self-organized criticality and sandpile slides [4], fluctuations in neuro-membranes [45], sunspot numbers in a 11 or 22 year period of the solar cycle [50]. Other examples can be found in [53].

![Figure 1.1 Measured phase noise of a 50 MHz BJT varactor-based VCO[32]. Three phase noise regions are identified as $f^{-3}$ (having a slope of $-9$ dB/octave), $f^{-2}$ (having a slope of $-6$ dB/octave), and $f^{-1}$ (having a slope of $-3$ dB/octave) Figure is taken from [32].](image)

The most puzzling noise observed with oscillators is the noise observed at a small frequency
offset from the carrier (i.e., the average oscillation signal). For example, the phase noise spectrum of a 50 MHz BJT varactor-based VCO is shown in Figure 1.1 has regions with slopes of $\Delta f^{-1}$, $\Delta f^{-2}$, and $\Delta f^{-3}$. The phase noise of oscillators and also of amplifiers have been observed to have slopes of $\Delta f^{-4}$, $\Delta f^{-5}$ as well [55].

### 1.2.1 Treatment of Flicker noise in oscillators

The first substantial treatment of flicker noise in electronic oscillators was due to Leeson [35]. Leeson determined that the oscillator phase noise has a region with $\Delta f^{-3}$ dependence that is due to low-frequency $\Delta f^{-1}$ noise, a $\Delta f^{-2}$ region due to white noise in the bandwidth of the oscillator’s tank circuit, and also a white noise region outside the bandwidth of the tank circuit. He provided an important observation that increasing the $Q$ of an oscillator feedback path reduces phase noise.

The next advance that influenced and lead to improvements in oscillator design was the phase noise model introduced by Hajimiri and Lee [21]. In essence this model identified a source of flicker noise as white noise converted from harmonics of the oscillation frequency. Their time-invariant model provides a richer description of phase noise, but it does not describe $\Delta f^{-1}$ noise or $\Delta f^{-n}$, $n > 3$, noise that is observed with oscillators.

Various physical theories have proposed that flicker noise originates from nonlinear dynamics and chaos [10, 17, 32, 54, 61, 62]. In this model flicker noise derives from a nonlinear process with delayed feedback. In this research, we show that this is correct and we present an easily understood model of a microwave oscillator.

A common approach to the analysis of noise in oscillator circuits, and indeed to circuits with periodicity, is a perturbation analysis [36, 37]. In this analysis, a free–running oscillator described by

$$\dot{x}(t) = f(x(t))$$  \hspace{1cm} (1.1)

where $x(t)$ and $\dot{x}(t)$ are $N$–dimensional independent variables of the circuit of oscillation frequency $f_0$, is perturbed by a small, zero-mean Gaussian noise source $n(t)$ to modify the analysis equation to

$$\dot{x}(t) = f(x(t)) + B(x(t))n(t).$$  \hspace{1cm} (1.2)

In general, $B(.)$ is a function of $x(t)$. The solution of this perturbed equation can be split into two terms – the first term corresponds to the variation of the tangential components of the oscillator’s limit cycle or orbit, and the second term corresponds to the variation of the orbit’s transversal components. Assuming that the noise term is purely additive and is of a small magnitude, it can be shown that the first term gives rise to amplitude modulation (AM) of the oscillator’s response, and
the second term gives rise to phase modulation (PM) of the response. This analysis is performed by the Floquet theory of linear time-periodic differential equations, and under the restrictive conditions imposed on the nature of the noise, it can be used effectively to model the noise in a circuit simulation environment. However, if the circuit has larger levels of noise or if the noise has multiplicative effects on the state variables so as to violate the linearity condition, this method of analysis breaks down [48].

Other approaches to modeling oscillator circuits with flicker noise are based on replacing the ordinary differential equations in a circuit simulator by stochastic differential equations and using intermittency-based noise sources [32, 33], with intermittency being a characterization of chaotic behavior. A key feature of any chaotic intermittent function is that it exhibits regular laminar sections separated by intermittent bursts of seemingly random behavior and this response can be generated using a simple nonlinear iterative deterministic function. Certain intermittent chaotic functions have been used to generate sequences with long memory characteristics. These sequences have slowly decaying correlations, which is an essential property of flicker noise. When these intermittent functions are used as sources of flicker noise in circuits, the differential equations representing the circuit are now stochastic. One of the main differences between Ordinary Differential Equations (ODEs) and Stochastic Differential Equations (SDEs) is that when integrating an SDE to find a solution, the point in the interval where the integral is evaluated matters. For a random function $h : [0, T] \rightarrow \mathbb{R}$ with intervals $0 = t_0 < t_1 < t_2 \ldots < t_N = T$, evaluating a integral at the starting point of each interval (the so-called Itô form), gives a different result if the evaluation occurs at the midpoint of each interval (the so-called Stratonovich form). While the interpretive nature of SDEs can seem troubling at first, it has been shown [11] that when a stochastic function involves multiplications of stochastic functions – the more general case – the Stratonovich interpretation is the correct one. Another upside of using the Stratonovich form is that conventional rules of calculus can be used to solving a system of SDEs (the Itô form requires modified calculus rules), which makes it easy to implement these ideas in a circuit simulator. While this accurately models the flicker noise observed in microwave oscillators down to very low frequencies, the complexity of the approach does not lend itself to the development of intuition.

The current phase noise models are not capable of explaining all the observations of noise in oscillators. This suggests some gaps in complete understanding of phase noise in oscillators. It is necessary to fill in these gaps so as to obtain a complete understanding of phase noise, mitigate the negative effects due to it and in the process, improve the performance of the microwave system. Most of the material in this section is taken from [48].
1.3 Original Contributions

This work represents a first attempt at using quad precision to simulate phase noise in microwave oscillators. It is found that flicker noise has one of its sources lying in the dynamics of a nonlinear system with time delayed feedback. Such a system is inherently chaotic in nature and possesses long term memory effects. The dynamics of chaos are evident as a long duration autocorrelation characteristic. As has been experimentally observed, phase noise is affected by oscillator $Q$. Baseband white noise affects the nonlinear dynamics of chaos presenting what is analogous to upconversion of baseband white noise to the oscillation frequency. The phase noise of a microwave oscillator is presented as being the result of a chaotic process.

The oscillator model developed to implement such a system is illustrated in Chapter 3. The corresponding frequency domain results clearly show $1/\Delta f^3$ and $1/\Delta f$ slopes where $\Delta f$ is the frequency offset from the carrier. Varying the $Q$ of the oscillator tank circuit, causes a change in the dynamics of the nonlinear system. An increase in delay causes the phase noise level to decrease. Such an increase also causes the intermittency and hence long term memory of the system to increase. This has not been observed and analyzed by previous phase noise models. Baseband white noise changes the dynamics of the system. It leads to $1/\Delta f^2$ slope in the spectrum of the oscillator.

1.4 Overview

Chapter 2 is a concise review of a large body of literature on noise in systems. The attempts of various researchers to provide physical explanations for the noise phenomenon has been explored and a brief overview on these efforts is provided. Also reviewed are the approaches to modeling noise and in particular, phase noise in oscillators.

Chapter 3 introduces a simple model of a microwave Colpitts oscillator. This model is implemented using a nonlinear function with time delayed feedback. The oscillator is tuned to operate at 1 GHz. Its lumped element components are modeled using an associate discrete modeling technique and implemented using backward euler form. The different tools and software used to simulate are briefly reviewed.

In chapter 4, the various trade-offs implemented for simulation of phase noise are discussed. Time domain properties of the oscillator are illustrated by its autocorrelation function. Impact of the usage of quad precision over double precision for phase noise simulation is illustrated. The importance of omission of the initial transient and the corresponding effect on phase noise is presented.

Chapter 5 provides the effect of quality factor and delay on phase noise of oscillator. The quality
factor of the oscillator is derived using the conventional definition of Q factor. Intermittency variation with delay is briefly discussed with corresponding impact on phase noise of the oscillator. Chapter 6 adds baseband white noise to the modeled oscillator. White noise is generated using the normal distribution function in C++. This noise is passed through a first order low pass RC filter to generate baseband white noise. The effect of addition of baseband white noise on phase noise level of oscillator is discussed. Finally Chapter 7 summarizes this work and provides guidelines to design oscillators with low phase noise.

1.5 Summary

In this chapter, the motivations and outline of the thesis were presented. It is important to understand phase noise in oscillators so that an accurate prediction of noise in microwave systems can be achieved. The major contributions of this research work along with an overview of the thesis were discussed.
2.1 Introduction

This chapter is a brief survey of the vast amount of literature available on the history of the understanding of noise and noise processes. This chapter is by no means comprehensive but it makes an attempt to mention a fair number of references on the subject ranging from the historical origins of noise to some more recent approaches to phase noise modelling.

Section 2.2 is a brief introduction to thermal noise found in electrical circuits followed by Section 2.3 which is an introduction to shot noise. Section 2.4 surveys theories of the physical origin of flicker noise and various mathematical models to describe flicker noise processes. Section 2.5 describes phase noise in oscillators. It covers various topics ranging from importance in practical systems, negative effects due to it, various observation in practical oscillator systems. In the end it covers major literature on various models aimed to describe phase noise in electronic oscillators.

2.2 Thermal Noise

Thermal noise is attributed to the random motion of free electrons in a conductor which are in a state of constant thermal agitation at temperature $T$. Such random fluctuations results in a corresponding
random voltage or current generated across the terminals of the conductor, with resistance $R$, as in Figure 2.1

The power spectral density (PSD) of $V_n(t)$ and $I_n(t)$ are related as:

$$S_{V_n}(f) = R^2 S_{I_n}(f). \quad (2.1)$$

We can derive the Power Spectral Density (PSD) of the voltage spectrum using the principles of thermodynamics in accordance with the developments of Nyquist in 1928, [46]. This development is briefly discussed below and yields the amount of thermal noise power in a signal. The results is dependent, to a good approximation, only on the temperature $T$, and the bandwidth of the receiver.

Suppose then that the receivers bandwidth is $B$ Hz, and consider that the noise $n(t)$ is restricted to the time interval $[t_1, t_2]$. By the sampling theorem, this signal can be represented by its samples spaced $1/(2B)$ seconds apart, and so the entire signal can be represented by $2B \Delta t$ samples, where $\Delta t = t_2 - t_1$. We can say that the noise signal has $2B \Delta t$ degrees of freedom. Now there is a theorem from statistical mechanics (Boltzmann’s Law) that for any motion, for every degree of freedom, there is an average kinetic energy of $\frac{1}{2} k_B T$ joules, where $k_B$ is the Boltzmann’s constant, i.e., $1.38 \times 10^{-23}$ J/K. Thus the (average) energy in our noise signal is $B \Delta t k_B T$ joules and the power (energy per unit time) is $B k_B T$ watts. The bandwidth $B$ is arbitrary, so we can divide it out and conclude that the power per unit of bandwidth is $k_B T$ watts. This value is called the power spectral density of thermal noise, and is usually denoted by the symbol $N_0$. In summary, the power spectral density of thermal noise is

$$N_0 = k_B T. \quad (2.2)$$
\( N_0 \) is sometimes called the one-sided noise spectral density, since only positive frequencies are considered.

We stated above that the average kinetic energy per degree of freedom in any motion is \( \frac{1}{2}k_B T \), a result due to Boltzmann. But Boltzmann’s law in fact gives not only the average energy of the motion, but the distribution of the energy as well. Indeed, the probability density function for the energy in a single oscillator at temperature \( T \) is the exponential density,

\[
p(x) = \frac{1}{k_B T} e^{-\frac{x}{k_B T}} \quad (2.3)
\]

That is, the probability that the energy lies between the limits \( x_1 \) and \( x_2 \) is

\[
Pr \{ x_1 \leq E \leq x_2 \} = \int_{x_1}^{x_2} p(x) \, dx \quad (2.4)
\]

It is an easy exercise in calculus to show that 2.4 implies that the average energy is \( k_B T \), i.e.,

\[
\langle E \rangle = \int_0^{\infty} x p(x) \, dx = k_B T \quad (2.5)
\]

Here the notation ‘\( \langle x \rangle \)’ indicates the ensemble average of \( x \). The energy \( E \) is the total energy, i.e., the sum of the potential and kinetic energies. On average they are equal, so by Equation (2.5) the average kinetic energy is \( \frac{1}{2}k_B T \), just as we stated above.

Planck showed that the distribution of the energy is not given by the exponential density 2.3 but rather by a discrete density of a similar form. He showed that \( E \) can assume only the discrete set of values \( 0, E_1, 2E_1, 3E_1, ... \), where \( E_1 = hf \), where \( f \) is the frequency of motion, and \( h \) is Planck’s constant, \( h = 6.62 \times 10^{-34} \) J sec. There are two important differences from Boltzmann’s density: the possible values for the energy are discrete, and the distribution is dependent on the frequency. The exact form of Planck’s density function is the geometric density,

\[
p_n = Pr \{ E = nE_1 \} = \alpha e^{-nE_1/k_BT}, \quad (2.6)
\]

which should be compared to the Boltzmann density in Equation (2.3). The constant \( \alpha \) in 2.6 is determined by the condition \( \sum_{n \geq 0} p_n = 1 \), and it turns out to be,

\[
\alpha = (1 - e^{-hf/k_BT}), \quad (2.7)
\]

Using the Planck formula, Equation (2.6) it is an easy exercise to find the average energy in the oscillation. We omit the algebra, but the result is
\[
\langle E \rangle = \sum_{n \geq 0} n E_1 p_n = k_B T \frac{x}{e^x - 1},
\]

where \( x = hf / k_B T \). If we recall that \( \langle E \rangle \) is the sum of the average kinetic energy and the average potential energy, and that the thermal noise is due to the kinetic energy of the electrons in the receiver, it follows that for every degree of freedom in the motion there is an average kinetic energy of,

\[
\frac{1}{2} k_B T \frac{x}{e^x - 1}
\]

Hence the power spectral density of thermal noise is,

\[
N_0 = k_B T \left( \frac{hf / k_B T}{e^{hf / k_B T} - 1} \right)
\]

which has the SI units of \( \text{W} / \text{Hz} \).

Finally, we should add that while the power spectral density as given by Equation (2.10) tells us about the variance of thermal noise, (its mean is zero), it does not tell us more about what kind of distribution the noise has. However, since thermal noise is caused by the independent contributions of many identical particles, the central limit theorem suggests, and experiment verifies, that thermal noise is a Gaussian process [15]. Thus we may say that thermal noise is a mean-zero, gaussian random process with power spectral density given by Equation (2.10).

For a more thorough examination of thermal noise in a resistor the reader is refered to Nyquists original work [46].

With inductors and capacitors, thermal noise is mainly attributed to noise contribution from its parasitic resistances. There has been some recent advanced discussion in [19] and [41].

The power spectrum is simply related to the autocorrelation function by the Wiener - Khinchin theorem, found by taking the inverse fourier transform of the power spectrum. This suggests that regardless of the probability distribution function of the signal, a flat power spectrum will correspond to an autocorrelation of a delta function. Hence we can say that thermal noise has zero correlation with its past events, and therefore forms a memoryless system.

### 2.3 Shot Noise

Shot noise is due to the corpuscular nature of current transport. In 1918, Walter Schottky discovered Shot noise in vacuum tubes and developed Schottky’s theorem [52]. Shot noise is always associated with direct current flow. In fact, it is required that there be dc current flow or there is no Shot noise.
Electrical current do not flow uniformly and does not vary smoothly in time like the standard water flow analogy. Current flow is not continuous, but results from the motion of charged particles (i.e. electrons and/or holes) which are discrete and independent. At some (supposedly small, presumed microscopic) level, currents vary in unpredictable ways. It is this unpredictable variation that is called shot noise.

The carrier entering the junction (or channel) must do so as a purely random event and independent of any other carrier crossing this point. If the carriers are not constrained in this manner then the resultant thermal noise will dominate and the Shot Noise will not be observed. A physical system where this constraint holds is a pn-\textsuperscript{+} junction. The passage of each carrier across the depletion region of the junction is a random event, and because of the energy barrier the carrier may travel in only one direction. Every carrier that crosses the depletion region generates a pulse. This arrival rate of pulses can be modelled by a Poisson arrival process of rate $\lambda$. The power spectral density can be derived using Van der Ziel's technique discussed in [64], which is briefly discussed below.

In order to find the fluctuation, first define $N$ as the number of carriers passing a point in a time interval $\tau$ at a rate $n(t)$. Then

$$N = \int_0^\tau n(t)dt$$

$$\bar{N} = \bar{n}\tau$$

Where $\bar{N} = \langle N \rangle$ and $\bar{n} = \langle n \rangle$ are ensemble averages and this result follows from the fact that time averages equal ensemble averages (the Ergodic theorem). If we define a new random variable $\Delta N$ such that

$$\Delta N = N - \bar{N}$$

Then we have removed the d.c. term leaving only the fluctuation. If we also define the random process $X_\tau$ then for sufficiently large $\tau$ we have

$$X_\tau = \frac{\Delta N}{\tau}$$

Note that for a Poisson process $\bar{N} = \text{var} N = \Delta \bar{N}^2$ and $\bar{n} = \text{var} n$ [20], where var is the variance of the process, therefore

$$X_\tau^2 = \frac{\Delta \bar{N}^2}{\tau^2} = \frac{\text{var} N}{\tau^2} = \frac{\bar{N}}{\tau^2} = \frac{\bar{n}\tau}{\tau^2} = \frac{\text{var} n}{\tau}$$

Hence,
\[ \text{var } n = \tau \bar{X}^2 \] \hspace{1cm} (2.16)

Now applying the Wiener-Khinchin theorem yields,

\[ S_n(0) = \lim_{\tau \to \infty} 2\tau \bar{X}^2 = 2\text{var } n \] \hspace{1cm} (2.17)

To convert to current we prove Schottkys theorem. The spectral intensity of the fluctuating current \( I(t) \) of average \( \bar{I} \) is

\[ S_I(0) = 2q \bar{I} \] \hspace{1cm} (2.18)

Where \( S_I(0) \) is the spectral density of the current fluctuations.

This derivation can be enhanced by assuming a time-varying rate for the Poisson process and that successive pulses have a degree of overlap, but in all cases it can be shown that the shot noise process has a white PSD and follows Gaussian distribution [15].

Thus, we can see that Shot noise has its autocorrelation, a delta function (similar to thermal noise). Hence, shot noise also forms a memoryless system.

### 2.4 Flicker Noise

Flicker noise, or \( 1/f \), noise in electrical circuits has generally been accredited to charge trapping and detrapping in conductors or transistors. Unlike other noise sources, a widely accepted explanation of the source of \( 1/f \) noise has eluded scientists. The first spectral density measurement of \( 1/f \) noise was published by J. B. Johnson [28]. In his 1925 paper Johnson was experimentally studying Schottky’s prediction of shot noise in vacuum tubes. He found that at high frequencies the measured noise agreed with prediction but at lower frequencies its deviation from prediction was substantial. The magnitude of this excess spectral density varied as the current squared and, although he made no comment on the frequency dependence, his published data show the excess noise to be proportional to \( 1/f \). He ascribed the effect to irregular temporal changes in the cathode emissivity. Since then this type of low-frequency noise has been observed in a variety of engineering and scientific situations and perhaps the only thing that is universally agreed upon is the ubiquity of the phenomenon. Forms of \( 1/f \) noise are observed in many of areas in [25], [59]

- Voltages or currents of:
  - Vacuum tubes, diodes, transistors.

- resistance of:
carbon microphones semiconductors metallic thin-films aqueous ionic solutions

- the frequency of quartz crystal oscillators
- average seasonal temperature
- annual amount of rainfall
- rate of traffic flow
- the voltage across nerve membranes and synthetic membranes
  - item the rate of insulin uptake by diabetics [13]
- economic data [39]
- the loudness and pitch of music [61]

The presence of $1/f$ noise in such a diverse group of systems (plus others not mentioned) has led researchers to speculate that there exists some profound law of nature that applies to all nonequilibrium systems and results in $1/f$ noise. Numerous specific models have been proposed, but not one can account for the presence of $1/f$ noise in even most of the systems listed. Perhaps the only similarity among these systems is the mathematical description that leads to $1/f$ noise.

Many attempts has been made to understand the characteristics of $1/f$ noise [9]. Let us discuss some major statistical properties of $1/f$ noise.

### 2.4.0.1 Is $1/f$ noise presence at equilibrium?

The work of many physicist and in particular that of F. N. Hooge and collaborators [24], produced several empirical formulas for $1/f$ noise in resistors, and in particular Hooge [23] showed that the $1/f$ voltage spectral density can be parametrized by the formula

$$S_v(f) = \gamma \frac{V_{DC}^{2+\beta}}{N_c f^\alpha}$$  \hspace{1cm} (2.19)

Where $\alpha$, $\beta$ and $\gamma$ are constants, $V_{DC}$ is the applied voltage and $N_c$ is the total number of charge carriers in the sample. This formula relates $1/f$ noise to the passage of current in the sample, and so people asked whether the noise was still present without a driving current. The problem was settled experimentally by Clarke and Voss [60] who found that $1/f$ noise was indeed present at equilibrium and this result was later confirmed by Beck and Spruit [5]. In their experiment there was no driving current, but there was also no guarantee of thermal equilibrium.
2.4. FLICKER NOISE

2.4.0.2 Is the $1/f$ noise process Gaussian?

There are two important reasons to answer this question: in the first place a Gaussian process is completely characterized by its average value and by the spectral density; secondly linear processes, like simple diffusion, are always Gaussian, therefore Gaussianity is an important indicator of linearity at the microscopic level.

To demonstrate linearity Voss [58] proved that $1/f$ noise showed linearity and Gaussianity in several conductors. However, it was later demonstrated that the noise processes observed by Voss were reasonably Gaussian and did not show linearity. Later several groups reported non-Gaussian signals, but Gaussianity can always be recovered at the macroscopic level from the superposition of many microscopic processes (this is a manifestation of the central limit theorem). From the theoretical point of view linearity is not expected, since many attempts to understand $1/f$ noise are based on the behavior of nonlinear processes.

Notice that the properties that we have discussed in this section have been tested on conductors only, i.e. on one very special subset of the systems that display $1/f$ noise, therefore we cannot draw any general conclusion from these observations.

There are several mathematical models in the literature that have been proposed for modeling $1/f$ noise. While some of them use physical intuition to explain this phenomenon, others use a broad array of mathematical functions to produce a model for $1/f$ noise. Below we briefly discuss some of these models.

2.4.1 A $1/f^\alpha$ response deriving from the superposition of relaxation processes

Schottky [52] developed a simple explanation of $1/f^\alpha$ noise in vacuum tubes. His explanation is that a free carrier is immobilized or trapped when it falls into a recombination center (trap). When several such carriers are trapped, it means that they are not available for conduction and as a result, the resistance of the semiconductor is modulated. If in the simplest case, a single trap is considered, then the kinetics of the fluctuation are characterized by a single relaxation time or time constant $\tau_z$. If this trapping process obeys a Poissonian statistic, then the correlation of this process is purely exponential and its spectrum is Lorentzian. The contribution to the vacuum tube current from the cathode surface trapping sites, which release the electrons according to a simple exponential relaxation law $N(t) = N_0 e^{-\lambda t}$ for $t \geq 0$ and $N(t) = 0$ for $t < 0$.

The Fourier transform of a single exponential relaxation process is [52]:

$$F(\omega) = \int_{-\infty}^{\infty} N(t) e^{-i\omega t} dt = N_0 \int_{-\infty}^{\infty} e^{-\lambda t} e^{i\omega t} dt = \frac{N_0}{\lambda + i\omega}$$

(2.20)
therefore for a train of such pulses \( N(t, t_k) = N_0 e^{-\lambda(t-t_k)} \) for \( t \geq t_k \) and \( N(t, t_k) = 0 \) for \( t < t_k \), we find

\[
F(\omega) = \int_{-\infty}^{\infty} N(t, t_k) e^{-i\omega t} \, dt = N_0 \sum_k e^{i\omega t_k} \int_{0}^{\infty} e^{-\lambda t + i\omega t} \, dt = \frac{N_0}{\lambda + i\omega} \sum_k e^{i\omega t_k} \tag{2.21}
\]

and the spectrum is

\[
S(\omega) = \lim_{T \to \infty} \frac{1}{T} |F(\omega)|^2 = \frac{N_0^2}{\lambda^2 + \omega^2} \lim_{T \to \infty} \frac{1}{T} \sum_k e^{i\omega t_k} = \frac{N_0^2 n}{\lambda^2 + \omega^2} \tag{2.22}
\]

where \( n \) is the average pulse rate and the triangle brackets denote an ensemble average. This spectrum is nearly flat near the origin, and after a transition region it becomes proportional to \( 1/\omega^2 \) at high frequency. This was sufficient for Schottky, who had found such a dependence in Johnsons data and provided a crude explanation of the frequency slopes existing in the plots. Later, he found that a single relaxation process was not enough and that there had to be a superposition of such processes, with different relaxation rates \( \lambda \) [7]. Thus, using this theory three characteristic regions: a white noise region at very low frequency, a \( 1/f \) noise intermediate region and a \( 1/f^2 \) region at high frequency were predicted.

### 2.4.2 Transmission line model of \( 1/f \) noise

The author in [29] introduced a simple RC transmission line model which when fed with white noise produced \( 1/f \) noise at its output. The circuit is shown in Figure 2.2.

The circuit consists of an infinitely long transmission line and fed with a white noise current source of magnitude \( I \) at its input. The impedance of this line is

\[
Z(f) = \sqrt{\frac{R}{j2\pi f C}} \tag{2.23}
\]

where \( R \) and \( C \) are the resistance and capacitance per unit length respectively. The PSD of the voltage at the output of the line is

\[
S_X(f) = \frac{I^2}{j2\pi f C} \cdot \frac{R}{j2\pi f C} \tag{2.24}
\]

Assuming a line of infinite length, \( S_X(f) \) is proportional to \( 1/f \) down to zero frequency. On the other hand, if the line is of finite length, then there will exist a lower frequency below which \( S_X(f) \) is white and its value is

\[
f_{\text{low}} = \frac{1}{2\pi RC\ell^2} \tag{2.25}
\]
2.4. FLICKER NOISE

with $\ell$ being the length of line. Keshner went on to derive an autocorrelation function for this model and proved that it consists of a sum of two terms: a nonstationary one and a stationary one, making the overall function nonstationary. Keshner also showed that if the time of observation was much smaller than the time elapsed since the system was turned on, then this autocorrelation function could be considered almost stationary.

2.4.3 Noise in diffusion processes

Most of the early noise studies were carried out on resistors, operational amplifiers or other electronic equipment and systems. A special emphasis was placed on resistors, and thus it was quite natural to identify simple random processes like the simple random walk and more general diffusion processes as possible origins of $1/f$ noise. Let us start with the simple stochastic differential equation for the Wiener process

$$\frac{dx}{dt} = n(t) \quad (2.26)$$

that describes Brownian motion in one space dimension, where $x(t)$ is the position of the Brownian particle, and $n(t)$ is a Gaussian white noise process with standard deviation $\sigma$, so that autocorrelation of $n(t) = \sigma^2 \delta(\tau)$. The spectral density of $n(t)$ is just $\sigma^2 / 2\pi$, and since Equation (2.26) implies the following relation between the Fourier Transforms $X(\omega)$ and $N(\omega)$ of the two processes

$$-i\omega X(\omega) = N(\omega) \quad (2.27)$$

We see that the spectral density of $x$ is

$$S_x = \frac{\sigma^2}{2\pi \omega^2} \quad (2.28)$$
And thus we conclude that Brownian motion has a $1/f^2$ spectrum. An example of a sample realization of a 1D brownian motion is shown in Figure 2.3. Brownian motion is considered scale invariant. Intuitively this means that a sample path realization as in Figure 2.3 looks the same in a probabilistic sense under any resolution. If the same simulation is run with a different timestep, the resulting sample path will have identical random statistical properties. These formulas show that neither Brownian motion in position space nor Brownian motion in velocity space can adequately describe $1/f$ noise, at least in one space dimension only the spectrum falls off too fast. In [16] it is shown that using a random driving term in the diffusion equation can lead to a $1/f^\alpha$ spectra.

### 2.5 Phase Noise

The phenomenon of Phase noise is mostly of concern with oscillators and is characterized as a frequency-domain phenomenon. Phase noise basically is the uncertainty in the phase of a signal. It represents the uncertainty in determination of the carrier frequency and instead of appearing as a perfect delta function at the carrier frequency in the frequency domain, it occupies a band of frequencies about the carrier whose PSD exhibits a power-law characteristic. This is represented graphically in the frequency domain, in Figure 2.4 where $\Delta f$ represents the band of frequencies.
2.5. *PHASE NOISE*  

Figure 2.4 Uncertainty in carrier frequency due to phase noise.

Phase noise has many definitions of which the most widely accepted is based on single-sideband phase noise, relative to carrier. This is stated as the power relative to the carrier in 1 Hz bandwidth at some frequency offset from the carrier. This definition will be used to quantify phase noise in all cases described here.

The instantaneous output of an oscillator can be represented by the equation

\[ V_{\text{output}}(t) = V_0 \cos[2\pi f_c t + \phi_n(t)] \]  

(2.29)

The function \( \phi_n(t) \) affects the phase of the output signal and its random nature with respect to time, gives rise to phase noise. The PSD of the oscillator can be related to the PSD of the phase noise according to:

\[ S_{\text{output}}(f) = \frac{P}{2} [S_\phi(f + f_c) + S_\phi(f - f_c)] \]  

(2.30)

where \( P \) is the power of the carrier. Pragmatically, the oscillator would contain amplitude noise (not modelled in the Equation (2.29) and phase noise. The amplitude noise can be added by modifying the equation above as:

\[ V_{\text{output}}(t) = V_0[1 + \alpha(t)] \cos[2\pi f_c t + \phi_n(t)]. \]  

(2.31)

Where the term \( \alpha(t) \) represents the amplitude fluctuations present in the circuit. In design of oscillators we ensure that the loop gain is greater than 1 and that it is limited with increasing oscillation amplitude. The amplitude, initially grows and then saturates. Since the amplitude is
saturated or limited, the oscillator circuit rejects amplitude noise and thus phase noise $\phi_n(t)$ is the relevant source of noise in oscillator circuits.

We can express Equation (2.31) in the phasor domain by the transformation

$$V_{\text{output}}(t) = V_0 [1 + \alpha(t)] \cos[2\pi f_c t + \phi_n(t)] = R e[(A + \alpha(t))e^{j\phi_0 t} e^{j\phi_n(t)}].$$

(2.32)

A corresponding phasor diagram is shown in Figure 2.5 where the noise is decomposed as its phase and amplitude vector components.

In Figure 2.5, $A$ represents the amplitude of the desired signal rotating at an angular frequency $\omega_0$. Noise signal, $\alpha e^{j\phi_n}$ is superimposed on the original signal. This noise signal has its own random amplitude $\alpha$ and random phase $\phi_n$. The amplitude noise is in phase with the desired signal, whereas the phase noise is orthogonal to it. Due to limiting mechanism of the oscillator, the amplitude is going to be saturated resulting in amplitude fluctuations being rejected. A point to be made here is that the orthogonal component has a phase variation, $\phi \approx \alpha/A$, hence we can say that the phase variation is going to be related to the amplitude fluctuations. Another insight one can gain form this is that by increasing the original amplitude $A$ of the desired signal the phase noise can be reduced.

As we discussed, for an oscillator the amplitude noise is reduced and some of this amplitude noise contributes to phase noise, termed as AM-PM converted noise or AM-PM noise. An example
of this is explained using the circuit in Figure 2.6.

![Varactor tuned VCO](image)

**Figure 2.6** Varactor tuned VCO. Figure is taken from [38]

The active current source (Mbias) generates noise. High frequency noise at $2f_0$ from bias circuits are down converted to noise at $f_0$ by the transistors M1 and M2. This process is also called down conversion of noise (mixing). The low frequency noise from the power supply and the bias circuit are converted to phase noise via an AM-PM mechanism. Noise on the supply within the loop bandwidth of the voltage regulator causes the supply voltage to change. The supply is a common mode input to the two outputs and hence the differential output would reject all of this noise, which leads to a minimum AM-AM contribution. If the supply varies, it will lead to a change in the operating point which can lead to a change in the VCO amplitude. The voltage across the varactor comprises noise from the power supply at one end and a tuning voltage (controlled by the Phase locked loop) at the other end. Hence, we have a noisy voltage across the capacitor. The voltage across the varactor determines the capacitance of the varactor, and this in turn determines the oscillation frequency of the VCO. Hence a noisy voltage across capacitor leads to a time-variable capacitance and therefore causes a time-varying shift in the oscillation frequency. This is an AM-PM noise conversion process.
2.5. PHASE NOISE

2.5.1 Importance of phase noise

It is important to study phase noise and to understand its origins so that we can predict its effect on signals, mitigate the negative effects due to it, and in the end build higher performance systems.

To understand the importance of phase noise, suppose we have a transmitter as shown in Figure 2.7(a) in which an oscillator is used to upconvert the baseband signal to the desired high frequency RF signal. If the VCO used here has large sidebands due to high phase noise, this noise gets modulated onto the baseband signal. Due to this the output would contain a spectrum in which signal components appear in adjacent channels as shown in Figure 2.7(b). This leads to adjacent channel interference. To control the level of interference stringent spectral masks have been established for transmit signals. Thus oscillator phase noise must be very low.

In the case of receivers, if we are trying to detect a small desired signal with center radian frequency, $\omega_{RF}$, next to a large blocker as shown in Figure 2.8(a). After passing through the mixer, the signals are down converted to an intermediate frequency using a VCO, having bad phase noise. The phase noise gets modulated onto the signals leading to masking of the desired signal, shown in Figure 2.8(b), causing self-interference. In order to avoid this, we need to minimize the phase noise of the oscillator. One might think, that such a blocker can be removed using bandpass filters, but as these signals are extremely close to one another we would need a very high Q filter which is not realizable. Another solution might be to use a phase locked loop, to achieve some filtering within the band.

Even without the blocker signal, phase noise results in uncertainty of the phase of the modulated as well as the demodulated signal. This can be seen in the spreading of the constellation diagram of the demodulated I and Q signals, shown in Figure 2.9. Phase noise causes the demodulated signals to be seen as a ellipse instead of a constant point leading to corruption of the I and Q received baseband signals and increasing the bit error rate. An important point can be made here that the
Figure 2.8 Receiver interference.

The elliptical shape of the smeared demodulated data shows only phase noise. If instead we had a circular smearing of demodulated data, it would correspond to amplitude noise in addition to phase noise.

Figure 2.9 Constellation Diagram. Figure is taken from [1].

Hence it is of prime importance to understand phase noise and put efforts to minimize its sources.
2.5.2 Observations of Oscillator noise in the Frequency Domain

[56] presented a discussion on the observations of phase noise. The presentation in this section follows that discussion. The most puzzling noise observed with oscillators is the noise observed at a small frequency offset from the carrier (i.e. the average oscillation signal). To develop an appreciation for the breadth of observations, the signals produced by several different oscillators will be considered. First, Figure 2.10 is a plot of the phase noise observed at the output of several oscillators and amplifiers operating at 5 MHz and 10 MHz. Curve (a) is the noise floor of the noise measurement instrument and spurious tones are observed at multiples of 60 Hz, the power mains frequency. Curves (b), (c), (d), and (e) show phase noise varying in straight-line segments. Being a log-log plot, these curves show phase noise varying as \( f^{-5} \), \( f^{-4} \), \( f^{-3} \), \( f^{-1} \) and \( f^0 \). None of the phase noise plots here show a region with an \( f^{-2} \) dependence, although this is observed with other oscillators [56].

Knowing the definition and importance of phase noise, we now briefly discuss some attempts and theories aimed to explain all the observations of phase noise in electrical oscillators. The widely

![SSB Phase Noise](image-url)

**Figure 2.10** Measured phase noise of low-frequency oscillators: (a) instrument noise floor; (b) HP 5087A frequency distribution amplifier at 5 MHz (used to drive the external reference input of several test instruments using a single high-quality oscillator); (c) TADD-1 frequency distribution amplifier at 10 MHz; (d) TADD-1 frequency distribution amplifier at 5 MHz; (e) Spectracom 8140T frequency distribution amplifier at 10 MHz. Five phase noise regions are identified as \( f^{-5} \), \( f^{-4} \), \( f^{-3} \), \( f^{-1} \), and white noise. The spurious signals are related to injected harmonics of the 60 Hz power mains [56].
accepted models are: 1) Leesons model (where, flicker noise near DC is upconverted to oscillation frequency), 2) Hajmiri and Lee model (where flicker noise away from the carrier get up converted due to nonlinearities in the oscillator loop, and 3) Unified Theory of Oscillator phase noise model (where phase noise depends only on the oscillating power, injected noise, and round-trip delay). A brief summary for these theories has been provided in next sections.

### 2.5.3 Leeson Model

Leeson in his paper [35] tried to explain the relationships among four commonly used spectral descriptions of oscillators short-term stability or noise behavior. He provided a heuristic derivation, presented without formal proof, the expected spectrum of a feedback oscillator in terms of the known oscillator parameters. Leeson assumed that \( \phi(t) \) in the Equation 2.29, is a zero-mean stationery random process, describing the deviations of the phase from the ideal. Leeson stated that for a feedback oscillator in which the feedback path contains an ideal resonator with a large quality factor \( Q \), free from frequency fluctuations, the relaxation time of the resonator is given by [51]:

\[
\tau = \frac{Q I_0}{\pi} = \frac{2Q}{\omega_0}
\]  

(2.33)

According to the oscillator equation stated above, in the case of slow fluctuations of \( \phi(t) \), slower than the inverse of the relaxation time \( \tau \), the phase \( \phi(t) \) can be treated as a quasi-static perturbation. The fractional frequency offset introduced by the static phase \( \phi \) at a time \( t \) is

\[
\frac{\Delta \omega}{\omega_0} = \frac{\Delta f}{f_0} = \varphi \frac{\Delta \omega}{\omega} \ll \frac{1}{2Q}
\]  

(2.34)

This equation tells us that the oscillator would respond to the perturbation with a frequency fluctuation:

\[
\Delta f(t) = \frac{f_0 \varphi(t)}{2Q}
\]  

(2.35)

with associated power spectral density as:

\[
S_{\Delta f}(f) = \left( \frac{f_0}{2Q} \right)^2 S_{\varphi(t)}
\]  

(2.36)

Thus the instantaneous output phase is:

\[
\phi(t) = 2\pi \int \Delta f(t) \, dt
\]  

(2.37)
The time-domain integration corresponds to a multiplication by $1/(j \omega) = 1/(2j\pi f)$ in the Fourier spectrum, thus into a multiplication by $1/(2\pi f)^2$ in the spectrum. The factor $2\pi$ in above Equation (2.37) cancels with the $2\pi$ in $1/(2j\pi f)$. Consequently, the oscillators slow phase fluctuation spectrum is

$$S_\phi(f) = \frac{1}{f^2} \left( \frac{f_0}{2Q} \right)^2 S_\phi(f)$$

(2.38)

For the fast components of $\varphi(t)$, i.e. those faster than the inverse of the relaxation time $\tau$, the resonator is a flywheel that steers the signal. Loosely speaking, the resonator would not respond to fast phase fluctuations; its output signal would be a pure sinusoid. Accordingly, the fluctuation $\varphi(t)$ goes through the amplifier and shows up at its output without being fed back to the input. No noise regeneration would take place in this condition, thus

$$\varphi(t) = \phi(t) \quad \text{and} \quad S_\varphi(t) = S_\phi(t)$$

(2.39)

By adding the effect of the fast and slow fluctuations of Equations (2.39) and (2.38), we get the Leeson formula relating the oscillators output phase spectrum to the amplifiers phase fluctuations:

$$S_\phi(t) = \left(1 + \frac{1}{f^2} \left( \frac{f_0}{2Q} \right)^2 \right) S_\phi(t).$$

(2.40)

The above is the Leeson formula, in this formula, the noise contributed by the resonator is not considered. A typical plot for such a model is shown in Figure 2.11:

Another method for deriving the Leeson equation uses a Negative Resistance model (Figure 2.12). This model specifically shows that how white noise in the oscillator gets transformed to 20 dB/decade filtered noise at the output signal. The transistors in the oscillator is modeled as a negative resistance and that this transistor introduces some noise $I_nR_n$, shown in Figure 2.12. A resonator (RLC tank), shown in shunt configuration, has a positive resistance $R_p$ (corresponding to the loss in the resonator) has a thermal noise associated with it. The active circuit and the tank resistance both inject uncorrelated noise into the tank.

In order to obtain sustained oscillations, the negative resistance of the transistor should balance out the positive resistance of the resonator. Hence, we can remove the two resistances and the circuit obtained is shown in Figure 2.13. In this equivalent circuit we would have only the noise current flowing through the LC circuit.

Now, we have to figure out the voltage of the tank relate to the current itself and how does it relate to the resonance frequency of the oscillator. The admittance of the tank is given as;
2.5. PHASE NOISE

CHAPTER 2. LITERATURE REVIEW

Figure 2.11 A typical plot of the phase noise of an oscillator having low Q versus offset from the carrier. Figure is taken from [35].

Figure 2.12 Negative Resistance model. Figure is taken from [38].
2.5. PHASE NOISE

CHAPTER 2. LITERATURE REVIEW

Figure 2.13 Equivalent circuit.

\[ Y_{tank} = j \omega C - j \frac{1}{\omega L} \]  \hspace{1cm} (2.41)

Quality factor, \( Q = \omega_0 CR = R/(\omega_0 L) \), is introduced in the Equation (2.41), as \( Q \) can capture the slope of susceptance near resonance. Hence, Equation (2.41) becomes:

\[ Y_{tank} = \frac{j Q_0}{R} \left( \frac{\omega}{\omega_0} - \frac{\omega_0}{\omega} \right) \]  \hspace{1cm} (2.42)

Let us evaluate the Equation (2.42) at an angular frequency of \( \omega = \omega_0 + \Delta \omega \), where \( \Delta \omega \) is an offset from the carrier frequency. Hence,

\[ Y_{tank} = \frac{j Q_0}{R} \left( \frac{\omega_0 + \Delta \omega}{\omega_0} - \frac{\omega_0}{\omega_0 + \Delta \omega} \right) \]  \hspace{1cm} (2.43)

\[ Y_{tank} = \frac{j Q_0}{R} \left( 1 + \frac{\Delta \omega}{\omega_0} - \frac{1}{1 + \frac{\Delta \omega}{\omega_0}} \right) \]  \hspace{1cm} (2.44)

Using Taylor series we know that; \( \frac{1}{1+x} \approx 1 - x \). Hence Equation (2.44) becomes;

\[ Y_{tank} = \frac{j Q_0}{R} \left[ \left( 1 + \frac{\Delta \omega}{\omega_0} \right) - \left( 1 - \frac{\Delta \omega}{\omega_0} \right) \right] = \frac{j Q_0}{R} \frac{2 \Delta \omega}{\omega_0} \]  \hspace{1cm} (2.45)

Equation (2.45) says that for very small \( \frac{\Delta \omega}{\omega_0} \), the admittance is proportional to it. It is important to note that the approximation works only when \( \Delta \omega \) is extremely small as compared to \( \omega_0 \), the...
2.5. PHASE NOISE

As we are looking at uncorrelated noise sources, we would use mean square currents instead of instantaneous values to get the noise response from the tank circuit.

\[
\overline{V_0^2} = \frac{(i_{nR_n}^2 + i_{nR_p}^2)}{|Y_{tank}|^2} = (\overline{i_{nR_n}^2 + i_{nR_p}^2})R^2\left(\frac{\omega_0}{2Q\Delta\omega}\right)^2
\]

Equation (2.46) looks like some noise multiplied by \(\left(\frac{\omega_0}{2Q\Delta\omega}\right)^2\). We can say that the noise going in the tank is being filtered by the resonator, hence at the output we get filtered noise at around resonance. As we know the noise contributed by the resistor as opposed to the noise from the active device. Leeson simplified the equation so as to compare the noise from active device to that from the tank. He defined this ratio, by introducing \(F\) term which is the ratio of the total phase noise injected into the tank divided by the noise coming from the finite tank resistance (this is NOT noise factor, despite the nomenclature).

\[
F = 1 + \frac{i_{nR_n}^2}{i_{nR_p}^2}
\]

The thermal noise contribution is given as

\[
\overline{i_{nR_p}^2} = \frac{4k_B T}{R} \frac{1}{2} = \frac{2k_B T}{R}
\]

The factor of 1/2 appears as thermal noise from the tank is going to have amplitude and phase deviation and as we know that for an oscillator only the phase deviation remains. Thus, from the equipartition theorem one-half of the thermal noise is attributed to phase noise [34].

From Equations (2.46), (2.47) and (2.48) we get;

\[
\overline{V_0^2} = \frac{2k_B T}{R} (F)R^2\left(\frac{\omega_0}{2Q\Delta\omega}\right)^2
\]

The single-sideband phase noise is the sideband (phase) noise power divided by the total signal power. Here it is given as,

\[
\mathcal{L}(\Delta\omega) = \left(\frac{\overline{V_0^2}}{P_{sig}}\right) = \frac{2Fk_B T}{P_{sig}} \left(\frac{\omega_0}{2Q\Delta\omega}\right)^2
\]

Equation (2.50) represents the Leesons formula without the noise floor, as it relates to the noise from the active device. Thus white noise from both noise sources gets filtered by the resonator and appears as 20 dB/decade slope from the carrier offset. If we have flicker noise in the active device, it would roll off as 10 dB/decade. This flicker noise, when filtered by the resonator, the output noise
would go down by 30 dB/decade. This model explains both the slopes observed in phase noise of oscillators.

To summarize Leeson effect, initially it was observed that nearly every physical system has fluctuations that vary as $1/f$ at low frequencies. This includes electrical devices such as the amplifier in an oscillator feedback loop. This leads to equal amplitude phase and amplitude noise superimposed on the oscillation. Amplitude fluctuations are suppressed by the saturation of the active device so that the only noise observed in good designs is phase noise. Leeson determined that the oscillator phase noise has a region with $\Delta f^{-3}$ dependence that is due to low-frequency $f^{-1}$ noise, a $\Delta f^{-2}$ region due to white noise in the bandwidth of the oscillators tank circuit, and also a white noise region outside the bandwidth of the tank circuit. The development of Leesons oscillator phase noise model is shown in Figure 2.11. Mathematically, the single sideband phase noise is given as, [35],

$$L(\Delta f) = L(\Delta \omega) = \frac{2FkT}{P_0} \left[ \left( 1 + \left( \frac{f_0}{2Q \Delta f} \right)^2 \right) \left( 1 + \frac{k_3}{\Delta \omega} \right) \right],$$

(2.51)

where $Q$ is the loaded $Q$ factor of the oscillators tank circuit and $F$ is an empirical factor. $L$ has the units of radians$^2$/Hz, or in decibels,

$$L|_{\text{dB}}(\Delta f) = 10 \log \left( \frac{2FkT}{P_0} \left[ \left( 1 + \left( \frac{f_0}{2Q \Delta f} \right)^2 \right) \left( 1 + \frac{k_3}{\Delta f} \right) \right] \right),$$

(2.52)

which has the units of dB/Hz or more usually as "decibels below the carrier" of power $P_0$, or dBc/Hz. It shows that the phase noise reduces by the square of $Q$. Also there is a frequency range where the phase noise level reduces as 20 dB/decade as the carrier offset $\Delta f$ increases. It also explains the effect of the power of oscillator signal $P$ and the oscillation frequency $f_0$ on phase noise. We can observe that for the case of a high $Q$ oscillator the term $\left( \frac{f_0}{2Q \Delta f} \right)^2$ diminishes and hence the spectrum exhibits a $1/\Delta f$ slope. This explain crossover of corner frequencies such that level of $1/f^2$ slope is reduced. We call the slope of 20 dB/decade as the region where white noise dominates. In the $1/\Delta f^3$ region the slope of 30 dB/decade is taken as indicating that low frequency flicker noise in the amplifier is being upconverted to the oscillation frequency.

This empirical model corresponded to experimental measured data and provided good design insight into improving phase noise of oscillators. Leeson’s model does not give any guidance as to how to suppress noise from active devices. It did not incorporate circuit topology and circuit effects in the phase noise analysis for an oscillator. It failed to explain some other effects which were later explained by Hajimiri and Lee in their model discussed in the next section.
2.5.4 Hajimiri and Lee Model

This theory overcomes the drawbacks of the Leeson model which made restrictive assumptions, which were applicable to only a limited class of oscillators. This model was based on a linear time invariant (LTI) system assumption and suffer from not considering the complete mechanism by which electrical noise sources, such as device noise, become phase noise. In particular, it took an empirical approach in describing the upconversion of low frequency noise sources, such as $1/f$ noise, into close-in phase noise. These models were also reduced-order models and therefore were incapable of making accurate predictions about phase noise in many types of oscillators.

In Leeson’s model, we combined the noise sources from the active device as well as from the resonator. In general, the noise contributed from the active devices will consist of noise from transistors, thermal noise from parasitic resistances. The transistors are switched hard from on to off and hence its noise contribution will be a function of the operating point of the transistor. This operating point is changing as the oscillator goes through its entire cycle, generating time-dependent noise in the oscillator. Likewise, we have frequency conversions of noise, i.e. noise being modulated onto the oscillation frequency from different harmonics. These contributions are not pointed out by Leesons model.

The time variant model proposed by Hajimiri and Lee in [21] is capable of proper assessment of the effects on phase noise of both stationary and even of cyclostationary noise (time-dependent noise) sources. Incorporating the time variant and cyclostationary model, Hajimiri and Lee stated that assuming the noise sources to be impulse functions, the magnitude of phase change would be dependent on the time at which the impulse is applied. If the impulse is applied when the voltage is at its peak, there will be no phase shift and only an amplitude change will result (proportional to the magnitude of the input impulse noise). However if the impulse is applied at the zero crossing, it has maximum effect on the phase and minimum on the amplitude. An impulse thus applied at a time in between will have both amplitude and phase changes. The amplitude changes can be minimized using limiters, which would cause the oscillator to follow a closed trajectory, called a limit cycle, irrespective of the starting point, [49],[43] and [14], this is illustrated in Figure 2.14. Thus only a phase fluctuation would persist indefinitely.

This is further explained using Figure 2.15 where we have a certain oscillation cycle that exists on the oscillator shown as a sine wave. We inject an impulse of charge to the tank in the oscillator circuit. This charge impacts the voltage (due to capacitor charge build up), and hence the oscillator responds to this impulse of charge injected within the tank. We want to access the phase change because of this injection of impulse charge. As shown in Figure 2.15, am impulse injected at $t_0$, zero cross over point, we see that instantaneously the voltage has gone up by a step. This step is
going to be dependent on the amount of charge and capacitance of the tank. The oscillator settles after this impulse injection, but as we can see that the phase is shifted significantly. In case when impulse is injected at time $t_2$, right at the maximum or minimum of the oscillation cycle, the voltage is increased but there is minimal change in phase (zero crossing) of the oscillator. Hence depending upon where the charge in injected, the oscillator is more or less sensitive. Intuitively, the circuit is highly susceptible to variations at the zero crossovers, where the transistors are switching from on to off and undergoing a phase (state) change.

In order to model this we would have to come up with a impulse response of the circuit. If we know the impulse response, we can then understand the circuits response to any arbitrary signal by convolving the signal itself with the impulse response of the circuit. Noise here is thought of as the aggregated response of multiple impulses of charge, which is then convolved with the impulse response of the circuit. Impulse response of the circuit is defined as how the output phase varies with respect to an input impulse of charge. This impulse response is called an Impulse Sensitivity Function (ISF-$\Gamma(2\pi f_0 t)$). If we inject such a charge and then vary the inject times where the charge occurs, we can look on the right side of Figure 2.16, to see how the phase changes. The ISF is zero for time $t_3$ (maximum or minimum $V_{out}$) and maximum at time $t_0$ (zero crossings). It can be noticed that the ISF is periodic, with the same period as the frequency of the waveform. The magnitude
of ISF indicates where the VCO waveform is most sensitive to noise current into tank with respect to creating phase noise. We can speculate that the ISF is somewhat realted to the derivative of the output voltage ($V_{out}$). At the maximum or the minimum the derivative is zero (due to limit), whereas at the crossover points the derivative is maximum. Hence, $ISF \propto \partial V_{out}/\partial t$. Two examples are presented to understand this, Figure 2.16, on the left a sinusoidal oscillation cycle and on the right a clipped oscillation cycle. The two ISF for both the cases is presented. Example 1 looks like a sine to cosine transformation through derivation. Example 2 has impulses only during rising or falling transitions.

In practice, the ISF is derived from the simulation of the VCO.

Suppose the impulse is injected at a time $\tau$, the unit impulse response for excess phase can be expressed as:

$$h_\phi(t, \tau) = \frac{\Gamma(\omega_0, \tau)}{q_{max}} u(t - \tau).$$

where $q_{max}$ is the maximum charge displacement, $u(t)$ is the step function, $\Gamma(x)$ is the impulse sensitivity function (ISF). It is a dimensionless, frequency and amplitude independent periodic function with period $2\pi$ which describes how much phase shift results from applying a unit impulse.
at time \( t = \tau \). Thus one can calculate the excess phase noise as:

\[
\phi(t) = \frac{1}{q_{\text{max}}} \int_{-\infty}^{t} \Gamma(\omega_0 t) i(\tau) d\tau,
\]

where \( i(\tau) \) is the noise current injected in the oscillator. The ISF is periodic and is hence expanded as a Fourier series as:

\[
\Gamma(2\pi f_0 \tau) = \frac{c_0}{\sqrt{2}} + \sum_{n=1}^{\infty} c_n \cos n2\pi f_0 \tau + \theta_n
\]

where \( c_n \) are coefficients of the Fourier series. As seen above, it has got a DC component added to a number of harmonics, where the fundamental is the oscillator frequency.

The first term with the \( c_0 \) coefficient indicates noise that is up-converted from baseband, while the term in the summation is the contribution to the oscillator phase noise due to down-conversion of noise near the harmonic frequencies. Suppose that we inject a low frequency sinusoidal perturbation current \( i(t) \) into the node of interest at a frequency of \( \Delta \omega \ll \omega \):

\[
i(t) = I_0 \cos(\Delta \omega t)
\]

After integration, the arguments at frequencies higher than \( \omega_0 \) are significantly attenuated by the averaging nature of the integration, except the term arising from the first integral, which involves
Therefore, the only significant term in $\phi(t)$ will be

$$\phi(t) = \frac{I_0 c_0 \sin(\Delta \omega t)}{2q_{\text{max}} \Delta \omega}$$  \hspace{1cm} (2.58)

Thus we get two impulses at $\pm \Delta \omega$ in the PSD of $\phi(t)$.

More generally, if the applied current is $i(t) = I_n \cos((n \omega_0 + \Delta \omega) t)$, the excess phase is given as:

$$\phi(t) = \frac{I_n c_n \sin(\Delta \omega t)}{2q_{\text{max}} \Delta \omega}$$  \hspace{1cm} (2.59)

Calculating the PSD of the output voltage $S_v(\omega)$ by using linear time variant, current to phase converter (discussed above), and a non linear system that represents phase modulation (PM) (transfroms phase to voltage), the injected current at $n \omega + \Delta \omega$ would thus result in a pair of equal sidebands at $\omega_0 \pm \Delta \omega$, assuming these noise sources to be white with mean square current $\overline{i_n^2}$, then the noise spectral density is [21]:

$$S(\Delta \omega) = 10 \log \left( \frac{\overline{i_n^2} \sum_{m=0}^{\infty} c_n^2}{\Delta f 4q_{\text{max}}^2 \Delta \omega^2} \right).$$  \hspace{1cm} (2.60)

Thus we see that, the noise at higher frequencies are down converted to the ones near the oscillation frequency. As can be seen from 2.60 and the foregoing discussion, noise components located near integer multiples of the oscillation frequency are transformed to low frequency noise sidebands for $S_v(\omega)$, which in turn become close-in phase noise in the spectrum of $S_v(\omega)$, this is illustrated in Figure 2.17.

It can be seen that the total is given by the sum of phase noise contributions from device noise in the vicinity of the integer multiples of $\omega_0$, weighted by the coefficients $c_n$. Thus by using the $\Gamma(x)$ function and adjusting the weights of $c_n$ we get different slopes $1/f^3, 1/f^2, 1/f$ in the PSD. In addition to the periodically time-varying nature of the system itself, another complication is that the statistical properties of some of the random noise sources in the oscillator may change with time in a periodic manner. These sources are referred to as cyclostationary. This is incorporated in this model by modifying the ISF $\Gamma(x)$ by another $\Gamma(\text{eff})$ function, which incorporates the cyclostationary noise sources.

Briefly it can be summarized using Figures 2.18 and 2.19. Noise being injected into the circuit, in Figure 2.18, is modeled in the frequency domain. This is because we have convolved in the time domain, it relates to multiplication in the frequency domain. Hence, we multiply the frequency domain noise with the frequency domain response of the ISF. Figure 2.18 shows a DC component as well as different branches relating to different harmonics. Thus the input noise gets multiplied by the DC (average) response of the ISF and passes straight to the output phase. Likewise, if the noise
gets multiplied with the fundamental of the ISF, it mixes with the fundamental coefficient of the ISF.

Note that $\frac{i_n^2}{\Delta f}$ is the single-sided noise spectral density of $i_n(t)$. In Figure 2.19, we can see how noise in different bands are translated into the phase response. For instance, during mixing of noise with fundamental of ISF, the noise around $f_0$ gets mixed to the noise at baseband. The baseband here represents the output phase response. Similarly, noise from second harmonics, third harmonics, and so on gets mixed down to baseband. Therefore, all the contributions that occur at different frequencies gets mixed to produce a phase response at the output. In case of flicker noise, we can see that the DC component shows that how low frequency noise on the left gets transformed to the low frequency noise on the phase, i.e. flicker noise in gives flicker noise at the output. This provides a circuit designer the intuition required to reduce flicker noise at the output by reducing the value of $c_0$. $c_0$ represents the average value of the signal, and hence by balancing the signal to be symmetric on the positive and negative sides we can reduce the value of $c_0$. This shows that a balanced circuit would have a lower flicker noise as compared to that of an unbalanced circuit.

This model shows that noise located near integer multiples of the oscillation frequency contributes to the total phase noise. The model specifies the contribution of those noise components in terms of waveform properties and circuit parameters, and therefore provides important design insight by identifying and quantifying the major sources of phase noise degradation. In particular, it shows that symmetry properties of the oscillator waveform have a significant effect on the upconversion of low frequency noise.

Even though ISF is a good way of modeling phase noise, computing ISF has some practical
Figure 2.18 Convolving in Frequency domain. Figure is taken from [38].

Figure 2.19 Frequency contributions. Figure is taken from [38].
2.5. PHASE NOISE

difficulties. An impulse is a $\delta$ function of time which has to be injected into a circuit node to compute ISF. In theory, the value of the amplitude in the limit for a delta function is infinity and its duration in the limit is zero. The strength of a delta function is the area under this curve. However, practically only a current with finite amplitude and time duration can be simulated. A better approximation can be achieved with a narrow impulse with small current amplitude and short duration. This results in a small amount of charge injection which is not sufficient to cause enough phase shift to be observed or subject to large numerical errors. This greatly limits the achievable accuracy of the computed ISF. On the other side, an impulse with large strength can drive the circuit away from its normal operating state and requires longer settling time. Another practical drawback for computing ISF is the long simulation time. In order to compute the ISF for a circuit node at time $t$, fine granularity on time scale is required to ensure accuracy and longer time is needed to allow the circuit to settle to its steady state after the impulse is injected. Therefore, computing ISF at a single node and at a single time instant needs a long transient simulation. This is to compute ISF over a complete period, over many time points for a single circuit node. As the circuit complexity grows, computation for all the ISFs becomes very time consuming and tedious job. The linearization of the oscillator phase in this model introduces nonphysical artifacts of infinite spectral power at frequencies approaching the carrier. This model also failed to predict injection locking behavior of oscillator. Injection locking is a frequency effect that can occur when a harmonic oscillator is disturbed by a second oscillator operating at a nearby frequency. When the coupling is strong enough and the frequencies near enough, the second oscillator can capture the first oscillator, causing it to have essentially identical frequency as the second.

2.5.5 Unified theory of oscillator phase noise

This model was proposed by Loh, William and Yegnanarayanan, Siva and Ram, Rajeev J and Juodawlkis and, Paul W, expands the techniques previously used to analyze phase noise in lasers [22] to develop an intuitive description of phase noise in an electromagnetic oscillator [36], [37]. This approach is consistent with the use of Linear Time Variant analysis, but does not impose linearization of the oscillators phase in the steady state as is done in Hajimiri and Lee phase noise model discussed before. In this approach, instead of treating noise as a separate signal outside the oscillator, it is suggested that the noise generated must be coupled into the modes of the resonant system, which together form the basis of the oscillator. The amount of noise coupling into each mode is modified by the density of modes. A concept of noise partitioning to treat the interaction of noise with the oscillating field is introduced in this model.

An oscillator electric field $E(t)$ is represented as a phasor in the complex plane, shown in Fig-
Figure 2.20 Vector-based description of an oscillator’s field under noise perturbation. Figure is taken from [36].

Figure 2.20. Its initial magnitude is normalized such that $E_0^2 = P_0$, where $P_0$ is the power of the signal. The oscillator’s phase starts at an arbitrary angle $\phi$ with respect to the real axis.

The addition of a noise event, having magnitude $E_n$ such that $E_n^2 = P_n$ and relative phase $\theta_i$, uniformly distributed between 0 and $2\pi$ results in perturbation of both the amplitude and phase of the original oscillating field. The amplitude of the total signal vector ($E_t$) is given by $E_t^2 = P_0 + \Delta P_i$, where $\Delta P_i$ is defined as the change in oscillation power induced by a noise event having relative phase $\theta_i$. The phase perturbation is calculated as,

$$\Delta \theta_i = \frac{E_n \sin(\theta_i)}{E_0}$$

(2.61)

As the amplitude and phase shift induced by a single noise event results in only small perturbations, we can assume $E_t \approx E_0$ and $\sin(\Delta \theta_i) \approx \Delta \theta_i$. Similarly,

$$\Delta P_i = P_n + 2\sqrt{P_n P_n \cos(\theta_i)}$$

(2.62)

Using the Wiener Khinchin theorem, the phase-noise spectrum through the autocorrelation of the oscillating field is calculated as,

$$\langle E(t) * E(t + \tau) \rangle \approx E_0^2 e^{i2\pi f_0 \tau} e^{-\langle (\Delta \phi^2) \rangle/2}$$

(2.63)

where, $f_0$ is the oscillation frequency, $\tau$ is the time delay over which noise events are observed, and $-\langle (\Delta \phi^2) \rangle$ is the second moment of the phase shift incurred over $\tau$. As the phase shifts have zero
mean, $-\langle \Delta \phi^2 \rangle$ is also the variance of the total phase shift. Since the oscillating power is stabilized around its mean, we can use Equation (2.61) to define $\Delta \phi$ induced by $M$ noise events through,

$$\Delta \theta_i = \sum_{n=1}^{M} E_n \frac{\sin(\theta_i)}{E_0}$$  \hspace{1cm} (2.64)

Since each noise event is independent, the variance in phase shift is shown to be

$$\langle \Delta \phi^2 \rangle = \bar{P}_n^2 M$$ \hspace{1cm} (2.65)

Where $\bar{P}_n$ is the average strength of a noise event. The factor of $1/2$ appears due to equipartitioning of the noise such that half of the noise power is used for driving phase fluctuations [34]. The average magnitude of the noise event $P_n$ is thereby found by determining the amount of noise coupled into the oscillating mode, from the noise distributed into the individual modes of the resonant system.

Using the analogy between laser cavity resonators and electronic resonators, white noise is integrated over a span $\Delta f$ around the oscillating mode and subsequently dividing the result by the number of modes contained in $\Delta f$. The noise power per node is calculated as [36]

$$\bar{P}_n = \frac{N_{PSD}}{T}$$ \hspace{1cm} (2.66)

Where $N_{PSD}$ is the power spectral density of the white-noise perturbation and $T$ is the total round-trip time delay of the cavity.

All the noise sources encountered during a round-trip are combined through vector addition and then the result is treated as a single noise source acting on the signal of Figure 2.20. Hence, the total power ($P_{n,tot}$) of the combined noise is,

$$P_{n,tot} = \frac{N_{PSD,tot}}{T}$$ \hspace{1cm} (2.67)

Representing the addition in average power for each of the individual sources,$N_{PSD,tot}$ is the power spectral density of the combined noise source that is injected once every roundtrip. Incorporating, Equations (2.67) and (2.65), we get,

$$\langle \Delta \phi^2 \rangle = \frac{N_{PSD,tot}}{2P_0 T^2} |\tau|$$ \hspace{1cm} (2.68)

Thus, the variance of the phase shift is linearly proportional to the observation time of noise events. This diffusion in phase is analogous to the random movement of particles governed by
Brownian motion. Substituting 2.68 into 2.63 and taking the Fourier transform, we find

\[ S(f) = P_0 \frac{1}{\pi} \frac{0.5 \Delta f_{3dB}}{(f - f_0)^2 + (0.5 \Delta f_{3dB})} \]  

(2.69)

where

\[ \Delta f_{3dB} = \frac{N_{PSD, tot}}{4\pi P_0 T^2} \]  

(2.70)

Equation (2.70) reveals that the oscillator full-width-at-half-power bandwidth \( \Delta f_{3dB} \) is primarily dependent on the noise-to-signal ratio. To achieve low phase-noise, the perturbation should be made small in comparison to the signal so that each noise event only weakly disturbs the oscillators phase (Figure 2.20). Furthermore, the round-trip time \( T \) delay should be made large so that the noise partitioned into each mode and its associated rate of injection decreases relative to the oscillating power. Dividing Equation (2.69) by the corresponding signal power, \( P_0 \) the normalized phase-noise spectral density is given by

\[ L(f) = \frac{1}{\pi} \frac{0.5 \Delta f_{3dB}}{(f - f_0)^2 + (0.5 \Delta f_{3dB})} \]  

(2.71)

The above model agrees with the observations made by Leesons model, discussed before, although unlike Leesons model the derivation of Equation (2.71) makes clear the random nature of noise in its interaction with the oscillating field. A similar derivation is done incorporating flicker noise as one of the sources of noise from the various resonant modes [37].

### 2.6 Summary

In this chapter, a review of the common sources of noise in electronic circuits was presented. These are thermal, shot and flicker noise. The derivation of the form of thermal noise is based on thermodynamical principles. Shot noise was assumed to be a superposition of several random processes that follow a Poissonian statistic and its spectral density is found to be white. In the case of flicker noise, an extensive review of the various theories of the origins of flicker noise was presented and explorations on the properties of flicker noise were highlighted. Various models were described aimed at explaining the occurrence of flicker noise. The definition and importance of phase noise was discussed along with its various observations in practical oscillator systems. The way phase noise manifests itself in oscillator systems and the major problems due to it were pointed out. Some widely accepted models for describing phase noise were briefly discussed. Major drawbacks and gaps in each of the models were briefly illustrated.
CHAPTER

3

DESCRIPTION OF SIMULATION

3.1 Introduction

As we saw in the last chapter, not all phase noise models are good enough to explain all the observations of phase noise in oscillators. This suggests gaps in complete understanding of phase noise in oscillators. Here an attempt is made to overcome these gaps. Such a fundamental understanding of phase noise will assist in predicting its level, mitigate the negative effects due to it and in the process build a higher performance system.

In this research, we found that apart from the various sources described in the earlier models, phase noise has another source of flicker noise. This new source can be widely explained using the theory of nonlinear dynamics and chaos. One of the speculations for flicker noise in oscillators, is that it is produced by a time-delayed feedback around a weak non-linearity. Such a system has inherit chaotic properties.

Flicker noise in electrical circuits has been attributed to charge trapping-detrapping, generation and recombination of noise due to base current in a transistor [47]. All these sources suggest delay in the process of generating oscillations. The effect of this delay intertwined with the non-linearity of the system, can lead to generation of flicker noise.

Flicker noise is often associated with the concept of memory. White noise is termed memoryless
and the correlation plot of ideal white noise has a single impulse at the origin. This implies a complete lack of correlation between the samples of a white noise sequence and that each sample exists independent of every other sample of the process. Flicker noise ($1/f$ noise) exhibits what is termed long term memory. In this case, the correlation plot decays at a rate that is slower than exponential. $1/f$ noise process exhibits long term memory character and its correlation plot, which in general, never decays. There is an immense amount of literature on evidence of $1/f$ noise as a long term memory process in different natural phenomenon like ocean current velocity, sea levels, resistors, voice and music broadcasts and many more [53]. Its manifests in a wide spectrum of scientific situations, suggests that there had to be a deep reason for the ubiquity of this kind of power-law noises. It has been observed that it is relatively straightforward to be able to produce rich and complex dynamical behavior of generating long term memory process, from a very simple set of underlying rules that repeat in an iterative fashion. With this motivation, we speculate that flicker noise can be generated through the underlying dynamics of a weak non-linear system with time-delayed feedback.

In Section 3.2 we layout the basic principles for an electronic oscillator. The oscillator properties are modified so that delay is incorporated into the system. Section 3.3 describes a new modeling technique used to simulate the oscillator. In Section 3.4 implements this technique for the proposed oscillator model. The various factors accounted for in simulation are discussed in this section. Finally, in Section 3.5 we summarize the process and some of the results obtained.

## 3.2 Architecture

### 3.2.1 Introduction

We make an attempt to illustrate that flicker noise has one of its sources lying in the inherent dynamics of a non-linear system with time-delayed feedback. In order to prove this, we would have to eliminate all sources of noise in the oscillator and incorporate time delay in one of its components. We would then have to ensure that the numerical quantization error introduced should be minimal in the simulated system. The preliminaries needed to understand the model are explained in the next subsection.

### 3.2.2 Preliminaries

The oscillator model simulated here, has the block diagram shown in Figure 3.1.

Consider a sequence $x_n; n = 0, 1, 2, ..., N$ of numbers generated by the recursive loop. When the number $x_n$ appears on the left of the non-linearity, the number $x_{n+1}$ immediately appears on its
3.2. ARCHITECTURE

CHAPTER 3. DESCRIPTION OF SIMULATION

right. The function \( f(\cdot) \) is preselected such that it models the oscillator non-linearity. Hence we can say:

\[ x_{n+1} = f(x_n) \quad (3.1) \]

The \( x_n \) forms a gain sequence. After a preselected interval (modelled as delay in Figure 3.1) \( x_{n+1} \) is transferred to the left of the non linearity, so that \( x_{n+2} \) immediately appears on its right. Consequently, the sequence is generated at instants, each of which are separated by the aforesaid interval. \( N \) is chosen to be large enough so that the sequence reaches its steady-state values.

The major parameters that need to be carefully selected are the non-linearity function as well as the value of the delay. A Fourier transform of the output, provides the power spectrum of the oscillator. As we show in later sections, that the effect of the delay causes a change in the phase noise of the system.

3.2.3 Colpitts Oscillator

The basic oscillator has the block diagram as shown in Figure 3.2

Oscillations start from noise or from a switch-on transient. In the spectrum of random signals, only a small energy is initially present at \( f_0 \), i.e. the oscillation frequency. For the oscillation to grow to a desired amplitude, it is necessary that \(|A\beta(j\omega)| > 1\) at \( \omega = \omega_0 \), where \( A \) is the gain of the amplifier and \( \omega_0 = 2\pi f_0 \), for small signals. With \(|A\beta(j\omega)| > 1\), the oscillation rises exponentially at a frequency \( f_0 \) defined by the arg \( A\beta(j\omega) = 0 \). As the oscillation amplitude approaches the desired value an amplitude control mechanism reduces the loop gain, so that the loop reaches the stationary
condition $A\beta(\omega_0) = 1$. These two criteria is extremely important for oscillator circuits, also termed as Barkhausen criteria. Because of the saturation of the amplitude of the oscillator, the amplitude fluctuations are removed and hence oscillator noise has no contributions from amplitude variations. We start by designing a colpitts oscillator as shown in Figure 3.3.

The oscillator in Figure 3.3 has built in weak non linearity and time delayed feedback. The transistor is modelled as a current source, equivalent of its small signal model. The pi-network is in colpitts configuration with $L = 30.4 \, nH \, \, , \, C_1=C_2= 5 \, pF$. The resonant frequency of the oscillator would then be calculated as:

$$f_0 = \frac{1}{2\pi \sqrt{LC_{eq}}}$$

(3.2)

where,

$$C_{eq} = \frac{C_1C_2}{C_1+C_2}$$

(3.3)

$$f_0 = 1 \, GHz$$

(3.4)

$R = 50 \, \Omega$ acts as the load for the oscillator. It is important to note that all the components are ideal and no noise sources have been added to the system. The present value of the voltage at node B depends on the past value value of voltage at node A. This can be seen as modeled in the transconductance of the amplifier. Such a dependency is aimed to add intermittency to the system. The amount of intermittency added to the system can be modified by varying the value of the delay $\tau$. A value of 100 ps is chosen here, which is about 1/10th of the time period of oscillation. Such a value is chosen so as to exaggerate the effects occurring due to delay.
3.2. ARCHITECTURE

CHAPTER 3. DESCRIPTION OF SIMULATION

Figure 3.3 Colpitts oscillator.

node B is

\[ V_B = g_m R A(t - 100) \]  

(3.5)

The effect of delay on phase noise is studied in later chapters. It is found that an increase in delay causes a decrease in the phase noise of the oscillator which suggests the effect of increase in intermittency, decreases the phase noise level of the system.

The transconductance of the amplifier needs to be modeled such that the amplitude of the

Figure 3.4 Hyperbolic tangent function. Figure plotted in Maple 18.
oscillations gets saturated once the Barkhausen criteria is met. This is implemented by using a tanh function. Such a function has the characteristics as shown in Figure 3.4.

As we can see, the function exponentially rises up to a certain value of \( x \) and then gets saturated on both the ends. This helps to build oscillations and then limit its amplitude. The effect of such a function on oscillation build up can be seen in Figure 3.5. We can use different functions to model the same effect. In our research, we found that if we change the non-linearity from tanh function to \( x^{1/3} \), function having an inflection point, there is no change in the phase noise of the oscillator. This suggests that the type of non-linearity does not have an effect on the phase noise. Some effort is made to change the non-linearity to even functions and see its corresponding effect.

### 3.2.4 Summary

In this section, we designed a non-linear model with time-delayed feedback oscillator. The amplifier is implemented as its corresponding small signal model, with the transconductance modeled as a tanh function. The oscillator is designed to resonate at 1 GHz, by properly tuning the resonator. Time-delay has been incorporated inside the amplifier and is preselected to 100 ps.
3.3 Implementation-Intelligent Model

3.3.1 Introduction

Associated Discrete Modeling (ADM) is used to model the lumped elements. ADM is derived from fundamental circuit theory, in particular nodal analysis techniques, and stable numerical integration techniques. This formulation is purely time-domain based and apart from mapping cleanly onto the Finite Difference Time Domain (FDTD) update equations, it also minimizes the aliasing and dynamic range limitations of the Fourier transform. Such a method is chosen so that it does not require interfacing with another simulator and work as a stand-alone software code. Numerical integration is done using backward Euler form, owing to its better convergence properties. This modeling technique discussed in the next few sections is highly adapted from [30]. We discuss the theoretical foundations for ADM in the next section before implementing it for the oscillator simulation.

3.3.1.1 Modified Nodal Analysis

Nodal Analysis is one of the cornerstones of circuit simulation techniques and is a graph-theoretical formulation of a circuit. It is based primarily on Kirchoff’s Current Law (KCL) which states that the algebraic sum of currents leaving any node must be zero. It involves the creation of a Nodal Admittance Matrix (NAM) which describes the components present in a circuit as well as the connectivity information of the components. The basic idea is that each element contained in the network has an admittance description such that the network behavior can be captured by a matrix equation of the form

\[ YV = J \] (3.6)

where the admittance descriptions are entered into the matrix \( Y \), the independent current source contributions are entered into the vector \( J \) and the unknown voltages at each terminal in the circuit are in vector \( V \). If the network has \( n \) two-terminal components, then there are \( n + 1 \) terminals, one of which is the ground terminal and can be safely omitted since it does not introduce any new information. This results in a size of \( n \times n \) for matrix \( Y \). Solving this linear system of equations involves inverting \( Y \) to obtain a solution of the unknown voltages at the terminals of a circuit, written as

\[ V = Y^{-1}J. \] (3.7)
In practice, the inverse is obtained using basic Gaussian Elimination for very small systems and LU factorization for larger systems. The advantage of LU factorization is that the number of operations required is of the order of \( n^2 \) for an \( n \times n \) system whereas for standard Gaussian Elimination, this would be of the order of \( n^3 \) [12]. The other advantage of LU factorization is that the L and U factors can be reused if just the right-hand side of the system - the values of the current sources - changes from time step to time step. This means the left-hand side matrix factorization has to be performed once. A detailed theory of nodal analysis is considerably more involved and out of the scope of this paper, see [57] for this. Instead, the principle of creating a nodal admittance matrix via inspection is illustrated by an example.

![Figure 3.6 Nodal analysis of a generic circuit.](image)

Consider the circuit shown in Figure. 3.6 which consists of three non-reference terminals (labeled 1, 2 and 3), four conductances (G1 - G4) and a current source \( I_s \). In order to create the NAM, the associated discrete model (ADM) of each circuit component must be used. An ADM is nothing but a description of the constitutive relationship of a circuit element along with its position in a circuit. For instance, consider a conductance \( G \) connected between two non-reference terminals \( j \) and \( k \). This contributes a value \( G \) in the NAM at the diagonal positions \( j \), \( j \) and \( k \), \( k \) and \( -G \) in the off-diagonal positions \( j, k \) and \( k, j \) with the assumption that the current is flowing from terminals \( j \) to \( k \). Graphically, the conductance is shown in Figure. 3.7.

The NAM contribution of this component is written as
3.3. IMPLEMENTATION-INTELLIGENT MODEL  CHAPTER 3. DESCRIPTION OF SIMULATION

Figure 3.7 Associated Discrete Model for a Conductance

\[
\begin{pmatrix}
  j & k \\
  j & (G & -G) \\
  k & (-G & G)
\end{pmatrix}
\]

(3.8)

If a conductance is connected between terminal \( j \) and ground, then the only contribution to the NAM is \( G \) at location \( j, j \).

An ADM for a current source, assuming that the current flows from terminal \( j \) to terminal \( k \) contributes only to the right-hand side source vector \( \textbf{J} \) and is expressed as

\[
\begin{pmatrix}
  j & k \\
  j & (-I) \\
  k & (I)
\end{pmatrix}
\]

(3.9)

Since the circuit in Figure 3.6 has 3 non-reference terminals, the NAM will be of size \( 3 \times 3 \). Using the rules for adding a conductance and current source to the matrix as described above, the NAM for the circuit becomes

\[
\begin{pmatrix}
  1 & 2 & 3 \\
  1 & (G1 + G3 & -G1 & -G3) \\
  2 & (-G1 & G1 + G2 + G4 & -G2) \\
  3 & (-G3 & -G2 & G2 + G3)
\end{pmatrix}
\]

(3.10)

The presence of the current source causes the RHS vector \( \textbf{J} \) to be \( (I, 0, 0)^T \) and the vector of unknown voltages \( V \) is nothing but the voltages at the circuit's terminals, \( (V_1, V_2, V_3)^T \).

In general, circuits can have voltage sources instead of current sources, and since a voltage source does not have a direct admittance description, the NAM must be modified to produce a Modified NAM or MNAM. This involves adding an extra row and column to the \( Y \) matrix and an extra row to the source vector \( \textbf{J} \) for every voltage source present in the circuit. For a voltage source \( E \) connected between terminals \( j \) and \( k \) and assuming the current direction is from \( j \) to \( k \), the ADM...
for the voltage source is

\[
\begin{bmatrix}
V_j & V_k & I
\end{bmatrix}
\begin{bmatrix}
j \\
k \\
n+1
\end{bmatrix}
\begin{bmatrix}
1 \\
-1 \\
1
\end{bmatrix}
\]

(3.11)

where the MNAM now has an extra row and column. The right-hand side vector must also be modified by adding an extra row containing the value of the voltage source, which is expressed as

\[
\begin{bmatrix}
\text{source} \\
n+1
\end{bmatrix}
\begin{bmatrix}
E
\end{bmatrix}
\]

(3.12)

The key point to be derived from modified nodal analysis is that if one has an ADM for any lumped circuit element expressed as a conductance, then its MNAM can be created, and upon inversion of the MNAM, the voltages at every terminal of that circuit can be easily determined. While the conductance for a resistance \( R \) is easily determined \((G = 1/R)\), obtaining an equivalent circuit for a capacitor and inductor is not obvious. In order to derive the ADMs of a capacitor and inductor, a focused treatment of the numerical integration of ordinary differential equations is presented in the next subsection.

### 3.3.2 Numerical Integration of Ordinary Differential Equations

Numerical integration is required for finding a solution to differential equations. Only ordinary differential equations (ODEs) are considered here. Consider an ODE written as

\[
\dot{x} = f(x)
\]

(3.13)

where \( \dot{x} \) is the time derivative of \( x \), and the time dependence of \( f(.) \) is assumed implicitly. The equivalent integral solution is

\[
x = \int_{a}^{b} f(x) dt + x(a).
\]

(3.14)

If the approximate solution \( x_n \) at discrete time point \( n \) is known, then the solution \( x_{n+1} \) can be determined by taking a time step \( \Delta t \). Starting with \( n = 0 \) and with initial value \( x_0 = x(a) \), \( \dot{x}_0 \) can be determined by (3.13) and the solution \( x_1 \) can be determined by assuming that a tangent line with
3.3 IMPLEMENTATION-INTELLIGENT MODEL CHAPTER 3. DESCRIPTION OF SIMULATION

slope \( \dot{x}_0 \) connects point \( x_0 \) with \( x_1 \). Expressed mathematically we obtain

\[
\dot{x}_0 = \frac{x_1 - x_0}{\Delta t} \tag{3.15}
\]

or

\[
x_1 = x_0 + \Delta t \dot{x}_0 = x_0 + \Delta t f(x, t) \tag{3.16}
\]

which is known as the Forward Euler integration formula. This is an explicit formula for time-stepping, and is the easiest to understand and implement in software. However, this method is known to result in inaccurate results and the error between the approximate and actual solution increases significantly with increasing \( \Delta t \).

An alternative is to make the assumption that \( x_1 \) is determined by the value at \( x_0 \) and the derivative at point \( x_1 \). While \( \Delta t \) can still be chosen explicitly, \( \dot{x}_1 \) cannot be found directly from (3.13) because \( x_1 \) is unknown. The typical approach in this case is to assume a value of \( x_1 \) from the Forward Euler formula, calculate \( \dot{x}_1 = f(x_1) \) to obtain the following form

\[
x_1 = x_0 + \Delta \dot{x}_1. \tag{3.17}
\]

This is known as the Backward Euler (BE) form of integration and is an implicit formula that requires a Forward Euler iteration step at each time step. In this way, the solution at steps \( n = 2 \ldots N \) can be determined. The BE form has better convergence properties that the Forward Euler form. It is the BE form that will be used to derive ADM descriptions for the capacitor and inductor in the next section.

3.3.3 Associated Discrete Modelling using the Backward Euler form

This sections presents the ADMs of a capacitor and inductor, all of which assume Backward Euler integration.

3.3.3.1 Capacitor

For a capacitor \( C \), its charge is linearly proportional to the voltage across it so that \( q = C v \), where \( q \) denotes capacitor charge. This gives

\[
i(t) = \frac{dq}{dt} = C \frac{dv}{dt} = C \dot{v} \tag{3.18}
\]

or

\[
\dot{v}_{n+1} = \frac{1}{C} i_{n+1}. \tag{3.19}
\]
Substituting this into (3.17) gives

\[ v_{n+1} = v_n + \Delta t \, f(v_{n+1}) \]
\[ v_{n+1} = v_n + \Delta t \, \dot{v}_{n+1} \]
\[ v_{n+1} = v_n + \frac{\Delta t}{C} \, i_{n+1}. \] (3.20)

Re-arranging the above equation leads to

\[ i_{n+1} = \frac{C}{\Delta t} \, v_{n+1} - \frac{C}{\Delta t} \, v_n \]
\[ i_{n+1} = g_{eq} \, v_{n+1} + i_{eq}. \] (3.21)

In words, this means that a capacitor can be modeled by a constant conductance \( g_{eq} = C/\Delta t \) in parallel with a current source \( i_{eq} = -(C/\Delta t) \, v_n \) whose value depends on the previous time step. This is shown in Figure 3.8.

![Figure 3.8 Associated discrete model for a linear capacitor based on the BE form.](image)

### 3.3.3.2 Inductor

For an inductor \( L \), its flux is linearly proportional to the current flowing through it, \( \phi = L \, i \), where \( \phi \) denotes inductor flux. This gives

\[ v(t) = \frac{d\phi}{dt} = L \frac{di}{dt} = L \dot{i} \] (3.22)

or

\[ i_{n+1} = \frac{1}{L} \, v_{n+1}. \] (3.23)
Substituting this into (3.17) and rearranging leads to the discretized BE form for the inductor which is expressed as

\[
\begin{align*}
    i_{n+1} &= \frac{\Delta t}{L} v_{n+1} + i_n \\
    i_{n+1} &= g_{eq} v_{n+1} + i_{eq}.
\end{align*}
\] (3.24)

In words, this means that a linear inductor can be modeled by a constant conductance \( g_{eq} = \Delta t / L \) in parallel with a current source \( i_{eq} = i_n \) whose value depends on the previous time step. This is shown in Figure 3.9.

![Figure 3.9](image)

**Figure 3.9** Associated discrete model for a linear inductor based on the BE form.

### 3.3.4 Summary

Here we laid the theoretical basis for modeling and simulating of discrete elements using ADM. Usage of backward euler form, helps to get better convergence of numerical integration. The error induced in Fourier transforms of time-series is highly reduced using ADM. In the next section we apply this novel modeling technique for the designed colpitts oscillator (Figure 3.3). Most of the material from this section is taken from [30].

### 3.4 Oscillator modeling

#### 3.4.1 Introduction

After laying down the theoretical foundations for modeling and simulating a circuit using ADM we apply it for the oscillator proposed in Section 3.2. The pseudo code used for simulation is described briefly. Transient plot of the oscillator is discussed.
3.4. OSCILLATOR MODELING

3.4.2 Discretization of oscillator

The first step in simulating the oscillator designed in Section 3.2 is to choose the time step of simulation. Here we chose a time step of 2.5 ps which is $1/400$th of the time period of the oscillator. Such a value helps to reduce error due to improper sampling of the wave. It avoids aliasing errors, by satisfying the Nyquist rate, i.e. sampling frequency $f_{\text{sampling}} > 2f_{\text{oscillator}}$. As no ambient noise is present in the circuit, an initial start-up of 1 mV is provided to the circuit by setting an impulse at time $= 0$ s. This assists in kick starting the oscillator circuit.

Quad precision and Discrete Fourier Transform is used for simulation. A detailed discussion on this tools is found in Chapter 4.

3.4.2.1 Pseudo Code

For the oscillator we designed in Figure 3.3, we discretize using the model described earlier in Figure 3.10.

The corresponding matrix developed using modified nodal analysis is:

$$A = \begin{bmatrix} gc1 + gl & -gl \\ -gl & gl + gc2 + gload \end{bmatrix}.$$
3.4. OSCILLATOR MODELING

\[ B = \begin{bmatrix} V1 \\ V2 \end{bmatrix}. \]

\[ C = \begin{bmatrix} -ic1 \\ -ic2 - g_m V1[0] \end{bmatrix}. \]

Solving the Matrix equation \( B = A^{-1} C \), we obtain the corresponding voltage \( V_2 \).

The pseudo code developed for simulation is listed below:

Initialize variables;
Define time step of FDTD simulation;
Declare vector size for voltages at nodes;
Define parameters for resonator;
Modify parameters according to ADM;
Define Matrices as listed above;
initialize oscillations;
for given number of windows;
  
  for \( (t = 0; t < \text{samplesize}; t++) \)
    
    Define delay modified parameters;
    get \( V_{output} \) from inverse matrix calculations;
    update new matrices;
  
for given window size
  
  get DFT for specified frequency component;
  store result to file;
  delete calculated data;
repeat process;
end;

A detailed code used to simulate is given in Appendix A. After running the simulation for around
25 million cycles, we get rid of the transient effects and get a stable oscillation. We can see the initial transient effects at the output of the oscillator in Figure 3.11. The oscillation starts up and increases exponentially and gets saturated to an amplitude of around 3.4 V. The voltage spike at a time of around 4.5 ns can be explained by the exponential value of a small deviation from saturated amplitude, which may be caused by a numerical error.

3.4.3 Summary

In this section, we modeled the proposed oscillator using ADM. We discretized its lumped components and simulated using modified nodal analysis. The pseudo code used for simulation in C++ is presented, with the time step set to 2.5 ps.

3.5 Summary

In this chapter, we hypothesize a new source of flicker noise, generated from the dynamics of a weak non-linear system with time delay feedback. An oscillator with inbuilt non-linearity and delay is proposed. Its non-linearity is modeled as a tanh function. The delay has been incorporated with the transconductance itself.

FDTD is used for simulation, along with modeling of the lumped components using ADM.
3.5. SUMMARY

CHAPTER 3. DESCRIPTION OF SIMULATION

Modified nodal analysis is used along with backward euler form for simulating the oscillator. Pseudo code for the discretized oscillator is presented. The proposed model of oscillator is shown to illustrate long term memory and hence intermittency in its system.
4.1 Introduction

In this chapter the oscillator simulation characteristics are explored and the simulation validated. The conditions for start-up of the oscillation simulation are explored. Then the autocorrelation characteristics of the oscillation are presented for various conditions. Finally the phase noise of the oscillator are presented. It is seen that it is essential to use quad precision in simulation as well as providing sufficient time for oscillation start-up transients to become insignificant. The oscillator characteristics are explored with and without delay in the feedback path. Initially the results are presented with the simulations performed using quad precision. Later in the chapter the oscillation characteristics are compared for simulations undertaken using double precision and quad precision.

4.2 Oscillation start-up

Oscillations are initiated by noise or by a switch-on transient. In the spectrum of random signals, only a small energy is initially present at $f_0$, i.e. the oscillation frequency. For the oscillation to grow to a desired amplitude, it is necessary that $|A\beta(j\omega)| > 1$ at $\omega = \omega_0$, where $A$ is the gain of the amplifier.
4.2. OSCILLATION START-UP

and $\omega_0 = 2\pi f_0$, for small signals. With $|A\beta(j\omega)| > 1$, the amplitude of the oscillation increases rapidly at an approximate frequency $f_0$ defined by the arg $A\beta(j\omega) = 0$. As the oscillation amplitude approaches its final value, an amplitude limiting mechanism reduces the loop gain, so that the loop reaches the stationary condition $A\beta(\omega_0) = 1$. These two criteria are extremely important for oscillator circuits, also termed the Barkhausen criteria. A hyperbolic tangent function is implemented here to saturate the amplitude. At first, this function causes the amplitude to increase exponentially, until it finally reaches a saturated value (Figure 3.4) to obtain sustained oscillations. The insight behind implementing a tanh function for oscillation build up was to mimic the behavior of typical transistor based amplifiers [27].

As no ambient noise is present in our oscillator system, an initial transient is required for oscillations to start up. This initial transient is provided by applying a 1 mV at time $t=0$ at node A in Figure 3.3. This helps the oscillations to build up and saturate to a final value of around 3.4 V as shown in Figure 4.1.

In this chapter, we consider two oscillator conditions, both with a nominal oscillating frequency of 1 GHz. The first oscillator has a delay of 100 ps and the second, has zero delay (i.e. no delay). For reasons explained later in this chapter, we omit the first 5 million cycles (5 ms) so that the artifacts due to initial transient effects are completely eliminated. We compare the properties of the oscillator for these two cases, in order to understand the impact of delay on the system.
4.3 Autocorrelation

As described in Wikipedia [2], autocorrelation is the cross-correlation of a signal with itself. Informally, it is a measure of the similarity of two copies of a function shifted in time. In statistics, the autocorrelation of a random process describes the correlation between values of the process at different times, as a function of the time lag [40]. Let $X$ be some repeatable process, and $i$ be some point in time after the start of that process. $i$ may be an integer for a discrete-time process or a real number for a continuous-time process. Then $X_i$ is the value (or realization) produced by a given run of the process at time $i$. Suppose that the process is further known to have defined values for mean $\mu_i$ and variance $\sigma_i^2$ for all times $i$. Then the definition of the autocorrelation between times $s$ and $t$ is

$$R(s, t) = E[(X_t)(X_s)]$$ (4.1)

where $E$ is the expected value operator. In signal processing, the above definition is often used without the normalization, that is, without subtracting the mean and dividing by the variance.

Given a signal $f(t)$, the continuous autocorrelation $R_{ff}(\tau)$ is most often defined as the continuous cross-correlation integral of $f(t)$ with itself, at lag $\tau$.

$$R_{ff}(\tau) = \int_{-\infty}^{\infty} f(u)\bar{f}(u-\tau)du$$ (4.2)

where $\bar{f}$ represents the complex conjugate of function $f(t)$. Similarly for a discrete function, the discrete autocorrelation $R$ at lag $l$ for a discrete signal $y(n)$ is calculated as [40],

$$R_{yy}(l) = \sum_{n \in \mathbb{Z}} y(n)\bar{y}(n-l)$$ (4.3)

Equation 4.3 is used to calculate the autocorrelation for the modeled oscillator.

White noise is generated using an inbuilt function in C++ which, uses random seed generator to generate normally distributed white noise. The autocorrelation of a normally distributed white noise ([15]) is calculated using equation. 4.3. The corresponding autocorrelation plot obtained is shown in Figure 4.2.

Figure 4.2 shows that white noise has an autocorrelation of 1 with itself (for lag=0) and 0 for all other lags. This illustrates that white noise has zero correlation with its past values. Hence white noise is also called as a memoryless system.

We compare the autocorrelation for the two cases discussed before. For the first case when
4.4 Phase Noise characteristics

4.4.1 Introduction

Having discussed the time domain properties of the oscillator output. We now observe the characteristics of the oscillator in frequency domain.

In this section, we first discuss the Fourier transform implemented here to obtain the spectrum.
4.4. PHASE NOISE CHARACTERISTICS

CHAPTER 4. PHASE NOISE SIMULATION

Figure 4.3 Autocorrelation of the oscillator output for delay, $\tau = 100\,\text{ps}$ (time step is set to 2.5 ps).

Figure 4.4 Auto correlation of the oscillator output for zero delay, $\tau = 0$ (time step is set to 2.5 ps).
of the oscillator. We see that elimination of the transient is necessary so as to obtain an error free spectrum. Single sided phase noise results for delay of 100 ps are discussed using quad and double precision.

### 4.4.2 Fourier Transform Implementation

The principle of Fourier Analysis is to test for the presence of each frequency component by multiplying the waveform, \( V_{\text{out}}(t) \), by a sine and cosine waveform of the same test frequency and average the results over one or more cycles of the test frequency. The Fourier transform decomposes a function of time (a signal) into its frequency components. The Fourier transform of a function of time is a complex-valued function of frequency, whose absolute value represents the amount of that frequency present in the original function, and whose argument is the phase offset from the basic sinusoid at that frequency [8]. The Fourier transform is called the frequency domain representation of the original signal.

For a discrete signal, the two major ways of implementing Fourier transform are Discrete Fourier transform (DFT) and Fast Fourier transform (FFT). The DFT is implemented as a summation of exponential series at each frequency point. This enables reduction of the memory needed as the results are calculated on the fly. The FFT is implemented as calculating harmonics not one at a time, but as a group, using a special algorithm and requires more memory than does a DFT. However the FFT requires much less processing power than does a DFT for the same number of frequency components. An FFT however, requires that the number of samples being analyzed be a binary number e.g. a power of two. The DFT has a higher accuracy than does the FFT, with the error in results being less than 1 percent. The only advantage the FFT has over the DFT is speed. For a waveform with \( N = 1024 \) samples, it takes \( N^2 \) computations to calculate the frequency components, while for a FFT it takes \( N \log_2(N) \) computations. So, for the DFT it takes 1,048,576 computations and for the FFT it takes 10,240 computations. The FFT is over 100 times faster. In addition, the FFT yields the complex amplitude of the full set of frequency components. Thus with 1024 samples, the FFT yields the complex amplitude of 512 frequency components. With our implementation of the DFT, the Fourier transform of selected frequency components can be evaluated. Owing to these reasons, DFT has been chosen over FFT for our simulations.

For a discrete function \( x[n] \), its corresponding DFT is calculated as [8],

\[
X_k = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}
\]  

(4.4)

\( X_k \) is a complex number that encodes both amplitude and phase of a sinusoidal component of...
function $x[n]$. The sinusoid’s frequency is $k$ cycles per $N$ samples. Its amplitude and phase are:

$$|X_k|/N = \sqrt{\Re(X_k)^2 + \Im(X_k)^2} / N \quad (4.5)$$

$$\arg(X_k) = \arctan \frac{\Im(X_k)}{\Re(X_k)} \quad (4.6)$$

For our oscillator, the real and imaginary parts are separately computed on the fly using inbuilt sine and cosine functions in C++.

The process followed to evaluate DFT starts with generating around 25 million time periods of voltage at the output of the oscillator. This data is then multiplied by cosine and sine of the angle term. The angle term is evaluated by multiplying the time point, the frequency component (whose DFT value is to be evaluated) and $-2\pi$, which is then divided by a sample size $N$. Evaluating a for loop and summing iteratively for all the time points, we obtain the real and imaginary parts of the Fourier transform value at the desired frequency component. An iteration of this calculation over the desired frequency components using a for loop is executed. This is then divided by the maximum Fourier transform value at the carrier frequency to obtain the single sided phase noise response of the oscillator. The code that implements the DFT is given in Appendix A.

### 4.4.3 Effect of initial transients

At the start of oscillations, a small impulse kick starts the oscillation. This transient eventually fades out to a stable steady state. It is important to remove the transient state from DFT calculations, so that an accurate spectrum is obtained.

#### 4.4.3.1 Phase noise with and without initial transient

However, the results presented in this and latter chapters require approximately, $N = 2 \times 10^9$ samples and computation time is dominated by memory accesses and the need to store and retrieve a large amount of data.

In order to see the error due to the initial transient, the spectrum of the oscillator is calculated using the DFT, without skipping the initial transient effects. The corresponding single sided phase noise obtained is shown in Figure 4.5.

The DFT simulation is normalized to dB/Hz, referenced to the phase noise at 100 Hz deviation from the carrier. As we can see in Figure 4.5, no stable value is achieved throughout the plot. The phase noise wiggles around some value for each frequency iteration. No clear slopes are seen in this figure. Due to these inaccuracies in the phase noise plot of the oscillator, it becomes extremely
important to remove the transient artifacts as much as possible. This is achieved by eliminating the first 5 million cycles, i.e. 5 ms in the time domain. DFT is then calculated over the next 25 million cycles. This process minimizes the error due to addition of transients.

To get an idea of the number of cycles that needs to be skipped at the start, a study is conducted to obtain the phase noise at 1 MHz for different number of cycles skipped. It is found that the phase noise at 1 MHz wiggles around some value for smaller number of cycles skipped. As the number of cycles skipped increases, the phase noise settles down to a final steady state value of around 9.8 dB. This is shown in Figure 4.6. The final level of the phase noise is around 130 dB below the level of the carrier.

At around 5 million cycles the phase noise settles to a steady value. Hence skipping 5 million cycles at the start reduces the error due to transient effects of oscillator.

After eliminating the artifacts from transient state of the oscillator, DFT is calculated on the next 25 million cycles. The corresponding single sided phase noise obtained is shown in Figure 4.7.
4.4. Phase Noise Characteristics

**Figure 4.6** Number of cycles skipped to minimize error due to transient.

**Figure 4.7** Phase noise of Colpitts oscillator with $Q = 15$ noiseless circuit elements ($L = 28$ nH, $C_1 = 9.3$ pF and $C_2 = 1$ pF). 0 db corresponds to -9.8 dBc.
4.4.4 Phase noise with delay

We can see from figure. 4.7, the phase noise plot is more reliable than figure. 4.5, to get an idea of the phase noise of the oscillator. The Quality factor of the resonator is tuned to 15, namely $L = 28$ nH, $C_1 = 9.3$ pF and $C_2 = 1$ pF. Simulation run times are excessive if simultaneously the phase noise at 1 hertz from the carrier is desired. In order to minimize the time required for simulation, phase noise is calculated in four slots with different frequency offsets (iterations). The first slot has phase noise calculated at multiples of 4 Hz up to around 10 Hz. The second slot calculates phase noise values at multiples of 10 Hz up to around 120 Hz. The third slot calculates values at multiples of 400 Hz up to around 4 kHz. The final slot calculates phase noise values at multiples of 40 kHz up to around 10 MHz.

This curve (figure. 4.7) is the result of a single simulation run. In this simulation there is no white noise and the noise observed is solely due to the chaotic process. Clearly defined $1/\Delta f$ and $1/\Delta f^3$ regions are observed with abrupt transition between the two regions. These straight line regions are remarkable and it is tempting to think that there is a straight line region joining points at 1 Hz and 300 Hz. However the discrete points at which the Fourier transform was evaluated are indicated from 1 Hz for 350 Hz. This curve has an exact $1/\Delta f^3$ dependence. Above 350 Hz up to 80 kHz small aliasing-like effects are seen. These are reduced and eventually disappear if a longer initial transient is discarded.

4.5 Impact of Simulation Precision

During numerical simulation, quantization error occurs due to rounding off the values to some unit of precision. This precision is governed by the number of bits that is used to define the different variables. Because quantization is a many-to-few mapping, it is an inherently non-linear and irreversible process (i.e., the same output value is shared by multiple input values, it is impossible in general to recover the exact input value when given only the output value). The quantization that occurs in this simulation, is called as Scalar quantization, which is attributed to rounding of high-precision numbers to the nearest multiple of some other unit of precision (for instance, rounding a large monetary amount to the nearest thousand dollars).

Another type of error that commonly occurs in numerical simulation is overflow errors. In a computer, the condition that occurs when a calculation produces a result that is greater in magnitude than that which a given register or storage location can store or represent. In such cases, the value gets resetted to default value and leads to loss of information. It is found in [6], that such errors leads to less accurate simulations and affects the corresponding spectrum. This distortion is minimum in
4.5. IMPACT OF SIMULATION PRECISION

CHAPTER 4. PHASE NOISE SIMULATION

case of a 7 digit precision, i.e. 128 bit precision. Hence, efforts are put to use Quad precision in our simulation.

Quad precision must be emulated, so there is a significant increase in run time over simulation performed in normal double precision. Oscillation initiates and simulations are typically run for 25 million cycles with the first 5 million discarded to remove transient effects. With so many time points, efforts are required to minimize the use of memory and ensure that the program remains in on-chip cache. One of the efforts made was to limit the memory used by implementing a rotation cyclical program. Such a program, calculates a window set of observed data, processes it, stores the result, deletes the previous data and overwrites the new data on the same space. The simulation being single threaded uses only one of the cores of the server.

It is necessary to use quad precision to minimize accumulated numerical error. Typical noise levels in a 1 hertz bandwidth at 10 MHz offset can be 200 dB below the carrier level and the extended simulation runs required to evaluate noise levels at 1 hertz offset mean that the number of time points in a simulation can exceed $10^9$, in the evaluation of phase noise which can extract noise power levels which can be $10^{-10}$ of that of the main oscillating signal. Another factor of $10^{12}$, equal to the number of time points, is required for fidelity of the Fourier transform. A simple analysis indicates that the voltage resolution required from the simulation must be considerably better one part in $10^{12} \times 10^9 \times \sqrt{10^{10}} = 10^{26}$. This far exceeds the capability of double precision which is approximately one part in $10^{15}$.

Figure 4.7 was obtained from defining all the variables and parameters of the oscillator with quad precision. Implementation of quad precision takes twice the amount of time required for a double precision simulation. A typical simulation run using quad precision takes up to 5-6 days to complete.

In order to ensure that implementation of quad precision is advantageous over implementation of double precision, a similar run is done to obtain the phase noise plot using double precision. The result obtained is shown in Figure 4.8. We can see from Figure 4.8, due to usage of double precision ripples are generated in the phase noise plot of the oscillator. This ripples can be seen dominantly at values close to the carrier. At frequency points away from the carrier these artifacts are removed.

Hence it is necessary to implement quad precision in our simulation. Quad precision would ensure that the results obtained are accurate and have minimal error from numerical simulation of the oscillator.
4.6 Conclusion

In this chapter the conditions required to obtain reliable simulations of oscillation were explored. It was seen that it is essential to use quad precision. This is not surprising given the number of time points used in simulation and the worst case accumulation of numerical error.
5.1 Introduction

After building the model of a nonlinear system with time delayed feedback, we characterize its phase noise dependence on its various parameters. We have implemented a quad precision simulator so that we have high accuracy of the observations and minimize quantization errors. As shown in the previous chapter, we discretized the oscillator using associate discrete modeling and implemented a standalone code, using backward euler form in C++ (gcc 4.7 version).

In this chapter, we examine the effect of quality factor of the resonator and the effect of delay of the transconductance on the phase noise of the oscillator. It is seen that delay adds long term memory to the system. Varying the delay has an immense impact on the phase noise of the oscillator. Increasing the delay increases the intermittency of the system. The Quality factor ($Q$) of the resonator, impacts the bandwidth of noise filtered and hence a high $Q$ would amount to a smaller bandwidth, leading to less noise added to the system.

In Section 5.2 we first discuss the effect of time delay on phase noise of oscillator. Here we first discuss the observation for a 100 ps time delay and then vary to 25 ps and 200 ps to see its corresponding effect on noise in the system. Section 5.3 discusses the derivation of the $Q$ factor of the oscillator, we then discuss the effect $Q$ has on phase noise of the oscillator. Finally in Section 5.4
we summarize our findings.

5.2 Effect of Delay

The non-linear model explained in the previous sections, have a delay associated with it. This delay is modeled inside the transconductance of the amplifier and its accredited to the random delays in the arrival of electrons or holes in the transistor. Due to this delay, the present value of the voltage at the output, depends upon a past value of voltage at the input. Hence instead of having an instantaneous response, the circuit would exhibit a delayed response to a current input.

In our simulation we vary the delay and see the effect it has on the phase noise of the oscillator. For the set up as explained in the previous chapter was simulated on a Linux server. A typical simulation of a single run takes around 5-6 days owing to the usage of quad precision. If we resort to using double precision the time taken is around 2-3 days, hence there is a significant increase in the time required for simulation. During the simulation, the first 5 million cycles were omitted so that all the transient effects are removed from the signal at the output of the oscillator. This omission helps to minimize errors due to aliasing, which might cause ripples to be propagated in the spectrum of the signal. Spectrum magnitude was calculated in three sets so as to reduce the time for simulation. The first set, phase noise as close as 0.1 Hz from the carrier upto 120 Hz has been accurately calculated. The second set calculate in multiples of 400 Hz upto around 80 kHz after which the third set calculates spectrum values in multiples of 4 kHz upto 10 MHz. It is important to note that there are no noise sources modeled in the oscillator system.

For the set up, discussed in the previous chapter, the corresponding phase noise plot is shown in Figure 5.1

Figure 5.1 clearly shows two slopes existing, $1/\Delta f^3$ and $1/\Delta f$. In this simulation there is no white noise and the noise observed is solely due to the chaotic process. Clearly defined $1/\Delta f$ and $1/\Delta f^3$ regions are observed with abrupt transition between the two regions. These straight line regions are remarkable and it is tempting to think that there is a straight line region joining points at 1 Hz and 300 Hz. However the discrete points at which the Fourier transform was evaluated are indicated from 1 Hz for 350 Hz. This curve has an exact $1/\Delta f^3$ dependence. Near the carrier, at around 0.1 Hz deviation, we have a $1/\Delta f^3$ slope, or 30 dB/decade, which extends upto around 350 Hz. After 350 Hz, phase noise reduces as 10 dB/decade, seen here as $1/\Delta f$ slope. 350 Hz is termed as the corner frequency. The noise would continue to roll off as 10 dB/decade as there is no noise floor currently present in the oscillator. The relatively thick lines from 10 kHz to around 80.3 kHz has some ripples associated with it. This is due to aliasing errors occurring due to the improper sampling of the signal at low deviations from the carrier. This aliasing error is observed in all the
5.2 EFFECT OF DELAY

Figure 5.1 Phase noise of oscillator with delay 100 ps and $Q = 400$ ($L = 25$ nH, $C_1 = 1$ pF, $C_2 = 1.26$ nF). 0 dB corresponds to -9.8 dBC.

plots to be discussed and can be reduced and eventually disappear if a longer initial transient is discarded along with proper sampling of the signal.

These slopes show that even though there are no noise sources in the system, flicker noise is being added to the system, seen in the phase noise plot of the oscillator. We can therefore say that, the dynamics of a nonlinear system with time-delayed feedback gives rise to flicker noise in the system. This flicker noise shows up in the oscillators spectrum as side bands.

Now, we consider several delays and see what effect delay has on the phase noise of the oscillator. The delay here was set to 25 ps, 100 ps and 200 ps. These values correspond to $1/40$th, $1/10$th and $1/5$th respectively, of the time period of the oscillator. The Q of the resonator and $P_{\text{sig}}$, power of the oscillator was kept constant in these simulations.

The corresponding phase noise plots for these different delays are shown in Figure 5.2. As we can see in the Figure 5.2, we observe that the phase noise level decreases as the delay increases. The phase noise level is highest for the delay of 25 ps and is minimum for a delay of 200 ps. This suggests that an increase in intermittency and hence long term memory of the system causes the noise to decrease. At 100 kHz, phase noise is around $-49$ dB/Hz for a system with 25 ps delay, whereas for a system with 200 ps delay the phase noise is around $-60.6$ dB/Hz.
5.2. EFFECT OF DELAY

Figure 5.2 Effect of Delay on phase noise, $Q = 400$ ($L = 25$ nH, $C_1 = 1$ pF, $C_2 = 1.26$ nF).

Figure 5.3 Autocorrelation of the oscillator output with a delay of 200 ps (time step = 2.5 ps).
5.3. Effect of Q

Quality factor ($Q$) is a dimensionless parameter that describes how underdamped an oscillator or resonator is. It characterizes a resonator’s bandwidth relative to its center frequency. Higher $Q$ indicates a lower rate of energy loss relative to the stored energy of the resonator. In such a case, the oscillations die out more slowly. $Q$ factor is quantified as the ratio of $f_0/\Delta f$, where $f_0$ is the oscillation frequency and $\Delta f$ is the 3-dB bandwidth of the resonator.

5.3.1 Derivation of Quality factor

We first derive the $Q$ factor for our model and then vary it to see its effect on phase noise. If the only ohmic component resides across the oscillators output terminals (Figure 3.3) the equivalent diagram becomes the one shown in Figure 5.4.

The conductances are

$$g_a = \frac{1}{R_a} = \frac{R_{load} + R_{o1}}{R_{load}R_{o1}}$$  (5.1)
where,
\[ R_{o1} = \frac{1}{g_m}. \]  
(5.2)

In order to get sustained oscillations, the real-part of the impedance should be zero and the imaginary part would equal
\[ \frac{G_m}{\omega_0 L} = \frac{\omega_0 C_2}{R_a} - \frac{1}{\omega_0 L R_a}. \]  
(5.3)

\[ G_m = \frac{1}{R_a} [\omega_0^2 LC_2 - 1] \]  
(5.4)

Equation (5.4) is an expression for the size of the transconductance, in order to get sustained oscillations up to a certain amplitude. The oscillation frequency for the oscillator is given as,
\[ \omega_0 = \sqrt{\frac{C_1 + C_2}{LC_1 C_2}}. \]  
(5.5)

Substituting the oscillation frequency, from Equation (5.5), into Equation (5.4), we get,
\[ G_m = \frac{1}{R_a} \left[ \frac{C_1 + C_2}{C_1} - 1 \right] = \frac{1}{R_a} \frac{C_2}{C_1} \]  
(5.6)

\( Q \) factor is defined as the derivative of the phase of the loop gain with respect to radian frequency, at \( \omega_0 \), as
\[ Q = \frac{\omega_0}{2} \left| \frac{d \angle A_{01}}{d \omega} \right| \]  
(5.7)

where, \( A_{01} \) represents the transfer function, given as
\[ A_{01} = \frac{V_1'}{V_1} = -\frac{G_m Y_a}{1 + Y_C/Y_a + Y_C/Y_b} = \frac{-G_m Y_C}{Y_a Y_b + Y_C (Y_a + Y_b)} \]  
(5.8)

and
\[ Y_a = g_a + j b_1 \]  
(5.9)

represents the parallel network of \( R_{load}, 1/g_m \) and \( C_1 \)
\[ Y_C = j b_L \]  
(5.10)

is the susceptor of inductor and
\[ Y_b = j b_{C_2} \]  
(5.11)
is the susceptance of \( C_2 \).

Hence,

\[
A_{01}(j\omega) = \frac{-G_m/j\omega L}{j\omega C_2(1/R_a + j\omega C_1) + [1/R_a + j\omega(C_1 + C_2)]/\omega L}
\]  
\[
= \frac{-G_m}{(1 - \omega_0^2LC_2)/R_a + j\omega(C_2 + C_1(1 - \omega_0^2LC_2))}
\]  
(5.12)

Its angle can be calculated as the inverse tangent of the ratio of the imaginary to the real part as

\[
\angle A_{01} = -\arctan R_a\omega \left[ \frac{C_2}{1 - \omega_0^2LC_2} + C_1 \right]
\]  
(5.13)

At the oscillation frequency, \( \omega = \omega_0 \), this angle should vanish and direct substitution from Equation 5.5 reveals that the expression in brackets becomes zero. The derivative with respect to angular frequency at \( \omega_0 \) now develops as,

\[
\frac{d}{d\omega} \arctan x = \frac{1}{1 + x^2}
\]  
(5.14)

Where at \( \omega = \omega_0 \), \( x = 0 \)

Thus,

\[
\frac{d}{d\omega} \angle A_{01} = -R_a\omega_0C_2 \frac{2\omega_0LC_2}{(1 - \omega_0^2LC_2)^2} = -2R_a(C_1 + C_2) \frac{C_1}{C_2}
\]  
(5.15)

And the corresponding Q-factor becomes

\[
Q = R_a\omega_0(C_1 + C_2) \frac{C_1}{C_2}
\]  
(5.16)

Equation (5.17) gives the Q factor for the designed Colpitts oscillator. For Figure 5.1 a Q of 400 was used. We vary Q and observe the corresponding effect on the phase noise of the oscillator. The resonator was tuned so that the oscillation frequency remains at 1 GHz and the Q’s of 15, 17.8 and 20 were considered. The delay was set to 100 ps in all the cases. Care was taken to ensure that \( P_{sig} \), power of the signal remains the same in all the cases. The corresponding values of the components of the resonator for these three cases are:

1. For \( Q = 15 \), \( C_1 = 9.34\ \text{pF} \), \( C_2 = 1\ \text{pF} \) and \( L = 28\ \text{nH} \).
2. For \( Q = 17.8 \), \( C_1 = 10.15\ \text{pF} \), \( C_2 = 1\ \text{pF} \) and \( L = 27.85\ \text{nH} \).
3. For \( Q = 20 \), \( C_1 = 10.8\ \text{pF} \), \( C_2 = 1\ \text{pF} \) and \( L = 27.6\ \text{nH} \).

The effects observed is shown in Figure 5.5.

We see that the noise level decreases as Q increases. At around 100 kHz away from the carrier the noise level is around \(-145\ \text{dBc/Hz} \) for Q of 15, while it is \(-157.75\ \text{dBc/Hz} \) for Q of 20. This suggests
5.4 Summary

In this section, we have determined the effect delay and quality factor of the resonator, has on the phase noise of the oscillator. An increase in delay causes a reduction in the phase noise. The phase noise also decreases if the quality factor of the resonator is increased. We also derived the quality factor of the designed oscillator. An increase in delay causes intermittency to increase leading to an increase in the long term memory of the process.

From these observations we can say that, phase noise of the oscillator is inversely proportional
to the round trip delay around the loop and the quality factor of the oscillator.
6.1 Introduction

In the previous chapter, we saw the effect of $Q$ and delay on the phase noise of the oscillator. We found that increasing $Q$ and delay, reduces the phase noise of the oscillator. The corresponding phase noise plot showed $1/\Delta f^3$ and $1/\Delta f$ as dominant slopes in the spectrum. There was no white noise being added to the system. It is speculated that $1/\Delta f^2$ slope in the phase noise is contributed by addition of white noise to the system [35]. Leesons model stated that $1/\Delta f^2$ slope has it sources from upconversion of white noise, filtered by the resonator, to the oscillation frequency.

In this chapter, we understand the effect of the addition of baseband white noise to the system. In order to generate baseband white noise, normally distributed white noise is passed through a RC low pass filter, having 31.83 MHz as its cutoff frequency. This filter adds white noise from baseband to the oscillator system modeled in Chapter 3. The noise added is normally distributed with mean of 0 and variance of 1.

In this chapter we first start with discussion of the statistical properties of white noise in Section 6.2. A baseband RC low pass filter is designed in Section 6.3. We discuss the various properties of the RC filter. In Section 6.4, we implement this RC filter using ADM technique discussed in chapter 3. We discuss the characteristics of added white noise. In Section 6.5, the various results obtained from
6.2 White Noise Properties

In this section, we discuss the statistical properties of white noise. Central Limit theorem proves that thermal noise follows a Gaussian distribution, with zero mean [15]. A Gaussian distribution also called as normal distribution has statistical properties as shown in Figure 6.1. Such a function tells the probability that any real observation will fall between any two real limits or real numbers, as the curve approaches zero on either side.

A normal distribution is given as,

\[ f(x, \mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \]  

(6.1)

where, \( \mu \) is the mean or expectation of the distribution, \( \sigma \) is the standard deviation, \( \sigma^2 \) is the variance of the distribution.

White noise is a random signal having constant Power spectral density [15]. According to Wiener-
Khinchin theorem [63], for a stationary process \( x \), if its autocorrelation function (also called autocovariance) exists and is finite at every lag \( \tau \), then its Power Spectral density (PSD) is related to it as;

\[
S(f) = \int_{-\infty}^{\infty} r_{xx}(\tau) e^{-2\pi i \tau f} d\tau
\]

(6.2)

where, autocorrelation of the process is defined as,

\[
r_{xx}(\tau) = E[x(t)x^*(t-\tau)]
\]

(6.3)

Hence, the autocorrelation function and the PSD form a fourier transform pair. Knowing the PSD we can estimate the autocorrelation, thus for a white noise process the autocorrelation is calculated as,

\[
S(f) = \sigma^2;
\]

(6.4)

\[
r_{xx}(\tau) = \int_{-\infty}^{\infty} \sigma^2 e^{2\pi i \tau f} df;
\]

(6.5)

\[
r_{xx}(\tau) = \sigma^2 \delta(\tau)
\]

(6.6)

This can be seen in Figure 6.2

Hence, white noise has zero correlation with its past. It possesses no memory, and hence is also termed a memoryless process.
6.3 Baseband RC Filter

In this section, a first-order lowpass RC filter is designed. Normally distributed white noise is passed through this filter. Due to filtering, white noise is confined to the baseband and is suppressed at higher frequencies. The aim here is to determine whether the baseband white noise is upconverted by the non-linear dynamics of the system and to check whether its addition changes the dynamics of chaos.

A lowpass RC filter is shown in Figure 6.3. At low frequencies, the capacitor acts as an open circuit, providing high impedance to ground whereas at high frequencies, the capacitor acts as an open circuit providing low impedance for the signal to ground. Hence, such a filter attenuates high frequencies and allows only low frequency components to pass through.

The transfer function for this filter is given as,

\[
H(s) = \frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{1 + RCs}
\]  

Where, \(s = j \omega\).

For our simulation we have fixed \(R = 5\ \text{K\Omega}\) and \(C = 1\ \text{pF}\). Such a filter configuration has a cutoff frequency, \(f_C = 31.83\ \text{MHz}\), or equivalently \(\omega_C = 2 \times 10^8\ \text{radians/second}\). The filter’s frequency analysis is done using Bode plot and its pole zero plot as shown in Figure 6.4.

We can see from the pole-zero plot that we have a pole at around \(-31.831\ \text{MHz}\). A first order filter provides an attenuation of about \(-30\ \text{dB}\) at 1 GHz, which can be improved by using higher order filters.

In our simulation, we generate normally distributed white noise and then pass it through the
low pass RC filter, so that baseband white noise is obtained at the output, see Figure 6.5

In Figure 6.5, $v_n$ is normally distributed white noise source with mean 0 and standard deviation of 1. This source is implemented in simulation using inbuilt normally distributed function in C++. The following code is used to generate the white noise:

```cpp
std::random_device rd; \define seed
std::mt19937 gen(rd()); \generate seed
std::normal_distribution<> d(0,1); \define mean=0, variance=1, noise voltage
```

The output voltage $v_B$ is given as,
6.4 Implementation

After generating white noise at baseband we need to add this noise to the oscillator system. This is done by adding the baseband white noise to the current source in Figure 3.3 so that the current becomes

\[ i = g_m \tanh[V_A(t - \tau)] + g_n v_B \]  

(6.9)

where, \( g_n = 1A/V \). This equation adds the white noise at baseband to the designed oscillator.

The baseband RC filter is modeled using the same discretization process, discussed in Chapter 3. Using ADM, the discretized circuit is shown in Figure 6.6, where, \( g_R = 1/R_B \), \( g_C = C_B/\Delta t \), \( \Delta t \) is the time step set to 2.5 ps and \( i_C = -(C_B/\Delta t)v_B \).

After declaring these parameters, we implement a matrix in order to solve using FDTD method. The matrices that are solved in order to obtain the filtered noise voltage \( v_B \) at every time step of simulation are as follows.

\[ A_{noise} = \begin{vmatrix} g_R & -g_R \\ -g_R & g_R + g_C \end{vmatrix}. \]

\[ B = \begin{vmatrix} v_n \\ v_B \end{vmatrix}. \]
6.5 Observations

The simulation is run following the same process as discussed in Chapter 4. The power of the oscillation signal is kept constant as before. The resonator is tuned so that a $Q$ of 3.7 is obtained. The phase noise plot is run in three sets, each set pertaining to a subset of discrete frequencies where the spectrum values are calculated. The reason being, the time taken for simulation for a frequency near the carrier is large as compared to the one far away from the carrier. Single sideband phase noise values are accurately determined at 0.1 Hz deviation from the carrier. The phase noise observed is shown in Figure 6.7.

We observe two dominant slopes $1/\Delta f^3$ and $1/\Delta f^2$. The region with a 30 dB/decade slope
extends from 0.1 Hz up to around 80 Hz. It is interesting to note that we have around 10 points on
this slope, which are connected through one line.

The most interesting observation here is the presence of $1/\Delta f^2$, 20 dB/decade slope, extending
from around 80 Hz upto around 10 MHz. This slope was previously not encountered, and is attributed
to the addition of baseband white noise to the system. This slope is due to filtering of white noise by
the resonator of the oscillator at the carrier frequency. This is termed upconversion of baseband noise
to the oscillation frequency. There is also experimental evidence of this slope in actual oscillators.
A region extending from around 350 Hz upto 80 kHz shows some aliasing errors due to improper
sampling of the signal. This can be reduced by increasing the sample size and discarding a larger
initial transient. Such a slope is not observed in the earlier simulations, where no white noise was
added to the system. We can therefore say, that white noise modifies the dynamics of the chaos.

It is important to note that the magnitude of noise added, quantified by the carrier-to-noise
ratio, was set to 75 dB in this case. Such a ratio (in dB) is calculated as

$$\text{SNR} = 20 \log \left( \frac{V_{\text{peak}}}{V_{\text{noise}}} \right)$$

(6.10)

Where, $V_{\text{peak}}$ is the peak amplitude of the oscillation signal at the output, and $V_{\text{noise}}$ is the amplitude
of the added white noise ($v_n$).

It is observed that a decrease in the carrier to noise ratio causes an increase in the level of phase
noise of the oscillator, i.e. the level of the region with $1/\Delta f^2$ slope increases. Guidelines can be
created to design low phase noise oscillator by reducing the level of baseband white noise being
added to the system.

In order to confirm that the $1/\Delta f^2$ slope region has its origins from baseband white noise we
compare the phase noise of the system for the case when $Q=17.8$, with no white noise being added
and the case when $Q=17.8$, with white noise (SNR=88 dB) being added to the system. Running the
same simulation, the observed phase noise is shown in Figure 6.8.

We can clearly see that due to the addition of white noise a slope of $1/\Delta f^2$ masks the slopes
of $1/\Delta f^3$ and $1/\Delta f$. This creates two abrupt transitions as seen in Figure 6.8. Thus, addition of
baseband white noise gives rise to $1/\Delta f^2$ slope in the phase noise of the system.

It is tempting to think that by varying the $Q$ and the carrier to noise ratio a condition can be
obtained, where we can see the presence of all the three slopes namely $1/\Delta f^3$, $1/\Delta f^2$ and $1/\Delta f$.
After some iterations of tuning $Q$ and the white noise level (SNR), it is found that for the condition
when $Q = 17.8$ and the carrier-to-noise ratio (SNR) is equal to 88 dB, three slopes exists in the single
sideband phase noise of the oscillator, shown in Figure 6.9.

We can see three slopes namely $1/\Delta f^3$, $1/\Delta f^2$ and $1/\Delta f$. We have two abrupt transitions be-
Figure 6.8 Phase noise with $Q = 17.8$, with and without white noise (SNR = 88 dB). 0 dB is considered as $-9.8$ dBc.

Figure 6.9 Phase noise with $Q = 17.8$ and SNR = 88 dB. 0 dB is considered as $-9.8$ dBc.
6.6 Summary

In this chapter, we observed the effect of addition of baseband white noise, on the phase noise of the oscillator. A baseband RC filter is modeled using ADM and simulated in C++ using FDTD with time
step set to 2.5 ps. The noise added is normally distributed with mean of 0 and standard deviation of 1. This noise is generated using standard inbuilt function in C++ (gnu 4.7.3).

It is found that $1/\Delta f^2$ slope is observed due to addition of white noise. This noise is a result of upconversion process of baseband white noise to the oscillation frequency. The level of this slope is controlled by $Q$ factor and carrier to noise ratio of the oscillator. An increase in the $Q$ causes the level of phase noise to decrease, whereas a decrease in the carrier to noise ratio, causes an increase in level of phase noise. Addition of white noise changes the dynamics of the nonlinear system with time delayed feedback.

It can be extrapolated that the phase noise of the system is inversely proportional to the carrier to noise ratio and $Q$ of the oscillator system.
Phase noise is one of the limiting factors for performance in microwave oscillators. There are gaps in the complete understanding of phase noise. It is important to build a complete understanding of phase noise, mitigate the negative effects due to it and in the process improve the performance of the system.

In a typical single sided phase noise plot of an electronic oscillator, distinct $1/\Delta f^3$, $1/\Delta f^2$, $1/\Delta f$ slopes are often observed. Flicker noise is suspected to cause $1/\Delta f^3$ and $1/\Delta f$ slopes in the spectrum of the oscillator. Various models have been proposed to explain the observations but none of them describe the existence of $\Delta f^{-1}$ or $\Delta f^{-n}$, where $n > 3$, slopes observed in oscillators.

In this work, it is shown that flicker noise derives from the dynamics of a nonlinear process with delayed feedback. Such a system generates a long term memory process, showing regular laminar sections separated by intermittent bursts of seemingly random behavior—hallmark of chaos. There is some literature hinting on such a system showing chaotic behavior [26].

This is shown by implementing a very simple microwave oscillator. A small signal model of the transistor is implemented using a nonlinear tanh function. The time delay is incorporated in the transistor. The circuit is modeled using associate discrete modeling technique with differential equations solved using backward Euler form. Such an implementation has the advantage to simulate as a standalone C++ code, instead of incorporating with existing circuit simulators. A Discrete
Fourier transform helps to remove the redundancy caused by Fast fourier transform. Quad precision provides a higher accuracy of observations over double precision. Transient effects are minimized in the observations by eliminating the first 5 million cycles.

The single sided phase noise shows two slopes $1/\Delta f^3$ and $1/\Delta f$ existing in the spectrum. As there is no source of noise present in the system, these two slopes are caused by the dynamics of the nonlinear system with delayed feedback. Thus, it is illustrated that flicker noise has one of its sources lying in the dynamics of a nonlinear system with time delayed feedback.

It is also found that the phase noise of the oscillator is inversely proportional to the delay. An increase in delay causes the intermittency (long term memory) to increase and leads to a corresponding decrease in the level of phase noise. The phase noise is also inversely proportional to the quality factor of the resonator. Increase in Q from 15 to 20 causes the phase noise to decrease by around 12 dB at an offset frequency of 1 MHz.

Baseband white noise is generated by passing normally distributed white noise through a first order RC baseband (low pass) filter. This filtered noise is then added to the current source. It is found that addition of baseband noise gives rise to a $1/\Delta f^2$ slope in the spectrum of the oscillator. Hence, addition of baseband white noise causes white noise to be upconverted to the oscillation frequency. The level of $1/\Delta f^2$ slope can be adjusted by changing the carrier-to-noise ratio or by increasing the amplitude of white noise added to the system. It is found that addition of such a noise causes a change in the dynamics of chaos.

Integrating all the effects observed, it can be concluded that flicker noise as one of its sources in the dynamics of a nonlinear system with time delayed feedback. Such a system is chaotic in nature. The phase noise of an oscillator is a result of chaos. Phase noise is found to be inversely proportional to delay, quality factor of the oscillator tank circuit, and carrier-to-noise ratio of the added baseband white noise.

In order to build an oscillator with low phase noise, a circuit designer should increase the delay incorporated in the system. Increase the Q of the oscillator tank circuit and make efforts to minimize the contribution of thermal noise to the nonlinear system.
BIBLIOGRAPHY


[38] M. Perrot, MIT OCW, Lecture 17, 6.776.


A.1 Source code for oscillator simulation

In this Section, the code used to simulate colpitts oscillator is presented. The oscillators is simulated using Finite Difference Time Domain method and its components are modeled using Associate discrete modeling. DFT is implemented as a running summation, to get the frequency domain characteristics of the oscillator. Normally distributed white noise source is generated using random library here. This white noise is later added to the circuit after passing through a RC filter. All variables are defined as quad precision variables.

In order to manage memory and speed, a cache friendly code is implemented here. As we know, data is always retrieved through the memory hierarchy (from fastest(L1) to slowest memory). A cache hit/miss usually refers to a hit/miss in the highest level of cache in the CPU. The cache hit rate is crucial for performance, since every cache miss results in fetching data from RAM (or worse) which takes a lot of time (hundreds of cycles for RAM and tens of millions of cycles for Hard Drive). In comparison, reading data from the (highest level) cache typically takes only a handful of cycles.

A very important aspect of cache friendly code is all about the principle of locality, the goal of which is to place related data close in memory to allow efficient caching. With this in mind, the following optimizations are incorporated in our code. Elements are declared as std::vector, helps to
store it in contiguous memory locations, and as such accessing them is much more cache-friendly than accessing elements in a std::list, which stores its content all over the place. Unpredictable branches are avoided as modern CPU architectures feature pipelines and compilers which are very good at reordering code to minimize delays due to memory access. If code contains (unpredictable) branches, it is hard or impossible to prefetch data. This will indirectly lead to more cache misses. Due to this, care was taken to implement a straight flow code, rather than calling routines to pass control from one function to another (causes more cache misses). In the context of C++, virtual methods represent a controversial issue with regard to cache misses (a general consensus exists that they should be avoided when possible in terms of performance). Virtual functions can induce cache misses during look up, but this only happens if the specific function is not called often. Hence, here we do not use functions and keep all the data to be processed sequentially, assisting in proper caching of data (gives more hits). As the server used for simulation have multiple processors sharing the same cache, there might be a problem when each individual processor is attempting to use data in another memory region and attempts to store it in the same cache line. This causes the cache line, which contains data another processor can use, to be overwritten again and again. This is avoided here by implementing a single threaded code. With these trade-offs in mind, the following code is written and simulated in C++.

A.1.1 Source Code

```c++
// define the header files
#include <iostream>
#define _USE_MATH_DEFINES
#include <map>
#include <random>
#include <cmath>
#include <iomanip>
#include <string>
#include <strstream>
#include <cstring>
#include <vector>
#include <complex>
extern "C" { 
#include "quadmath.h" //quad precision from C++ gnu library 
}
```


#include <fstream>

using namespace std;
typedef complex<double> dcomp;

int main()
{
    // defining double precision variables
    double c0; // size of simulation (number of points)
    double dx; // half of the observation time in seconds
    double dt; // time step
    double C1; // capacitor at node 1
    double C2; // capacitor at node 2
    double L; // inductor between the nodes 1 and 2
    double fs; // sample frequency (not used)
    double Ceq; // equivalent capacitor for C1 and C2
    // shunt configuration
    double freq; // frequency of oscillation
    double tau; // time period of oscillation
    double Rload; // load at the output
    double gload; // conductance form of the load at the output
    double gm; // transconductance of current source
    double gc1; // ADM parameter for C1 (conductance)
    double gc2; // ADM parameter for C2 (conductance)
    double ic1; // ADM parameter for C1 (current source)
    double ic2; // ADM parameter for C2 (current source)
    double il_init; // ADM parameter for L with
    // initial startup (current source)
    double il; // ADM parameter for L (current source)
    double gl; // ADM parameter for L (conductance)

    // Size of simulation
    int size_org = 2e9;

    // define time step for simulation using Associate discrete modeling
c0=2000000000; //size of simulation for removing aliasing error
dx=0.01; //half the real observation time
dt=dx/(2 * c0); //calculating time step of simulation
//cout<<dt<"\n"; //printing time step value

//Declaring vectors for voltages at node 1 and 2
size_t size=size_org; //defining size of vectors
std::vector<_float128> V1(size); //quad precision vectors
//to calculate voltage at node 1
std::vector<_float128> V2 (size); //quad precision vectors
//to calculate voltage at node 2

//Defining parameters
//Providing initial transient to kick start the oscillator
V1[0]=1e-3;
C1=1e-12; //capacitor at node 1
C2=5e-12; //capacitor at node 2
L=30.4e-9; //inductor between nodes 1 and 2
fs=4e12; //sampling frequency (not implemented here)

Ceq=(C1 * C2)/(C1+C2); //calculating total capacitance
//(C1 and C2 in parallel)
//cout<<Ceq<"\n"; //testing Ceq print

//calculating the frequency of oscillation
freq=1/(2 * M_PI * (sqrt(L*Ceq)));
//cout<<freq<"\n"; //testing freq print

tau=1/freq; //calculating time period
//cout<<tau<"\n"; //testing time period print

Rload=50; //resistor at node 2 (output)

gm=0; //initial transconductance value at time=0
gcl=(C1)/dt; //defining gcl param for ADM (conductance)
    //cout<<gcl<<"\n";  //testing gcl print

    //defining ic1 param for ADM with initial startup
    ic1=-(gcl)*V1[0];  //current source of C1
    cout<<ic1<<"\n"; //testing ic1 print

gc2=(C2)/dt;  //defining gc2 param for ADM (conductance)
    //cout<<gc2<<"\n";  //testing gc2 print

    //defining ic2 param for ADM with initial startup
    ic2=-(gc2)*V2[0];  //current source of C2
    cout<<ic2<<"\n"; //testing ic2 print

gl=dt/(L);    //defining gl param for ADM (conductance)
    //cout<<gl<<"\n";  //testing gl print

    //defining il param for ADM with initial startup
    il_init=gl*(V2[0]-V1[0]);  //current source of L
    cout<<il_init<<"\n"; //testing il print

gload=1/Rload;  //defining gload param for ADM
    //cout<<gload<<"\n";  //testing gload print

    //defining A B C
    __float128 s1 = gcl+gl;  //combining conductances at node 1
    __float128 s2= gl+gc2+gload;  //combining conductances at node 2

    //defining quad precision conductance matrix
    __float128 A[2][2] = { {s1,-gl}, {-gl,s2} };

    //defining quad precision initial node voltage matrix
    __float128 B[2][1] = {{ V1[0]},{V2[0]}};

    //defining quad precision initial node current matrix
A.1. SOURCE CODE FOR OSCILLATOR SIMULATION

```c
__float128 C[2][1] = {{-ic1+il_init},{-ic2-il_init-(gm*V[0])}};

/*
//display C- initial node current matrix
for (int nRow = 0; nRow < 2; nRow++)
{
    for (int nCol = 0; nCol < 1; nCol++)
        cout << C[nRow][nCol] << "\t";

    cout << endl;
}
*/

/*
//taking inverse of matrix A- conductance node matrix
__float128 det;
__float128 inv_A[2][2]= {{0,0},{0,0}};
det=A[0][0]*A[1][1]-A[0][1]*A[1][0];
inv_A[0][0]=A[1][1]/det;
inv_A[0][1]=-A[0][1]/det;
inv_A[1][0]=-A[1][0]/det;
inv_A[1][1]=A[0][0]/det;

/*
//display inv_A- inverse of conductance matrix
for (int nRow = 0; nRow < 2; nRow++)
{
    for (int nCol = 0; nCol < 2; nCol++)
        cout << inv_A[nRow][nCol] << "\t";

    cout << endl;
}
*/

/*starting simulation to calculate the node voltage at each iteration, size_org set here to 2e9,
*/
starting after 40*dt or 100 ps to account for vector association with negative indexes
*

for (int t = 40; t < size_org; t++)
{
  /*Multiplying inverse with the current matrix to get node voltage at each time step*/

  //At node 1
  B[0][0]=(inv_A[0][0]*C[0][0])+(inv_A[0][1]*C[1][0]);
  //At node 2
  B[1][0]=(inv_A[1][0]*C[0][0])+(inv_A[1][1]*C[1][0]);

  //Updating the vectors V1 and V2 –the node voltages
  V1[t+1]=B[0][0];
  //cout<<"V1 is"<<V1[t+1]<<"\n";
  V2[t+1]=B[1][0];
  //cout<<"t= "<<t<<" V2 ="<<V2[t+1]<<"\n";

  //Updating current at nodes–for each iteration

  //for current source at node 2
  ic2=-(C2)/dt*V2[t+1];
  //for current source at node 1
  ic1=-(C1)/dt*V1[t+1];
  //cout<<"ic1 is "<<ic1<<"\n";
  //current source for L
  il=il_init+(gL*(V2[t+1]-V1[t+1]));

  //ADDING white noise (normal distribution)
  //generating normally distributed white noise
  std::random_device rd; //random number generator that
  //produces non–deterministic random numbers.
/
generating random numbers using 32-bit Mersenne Twister
algorithm sets the current state of the engine, and
acts like a seed
*/
std::mt19937 gen(rd());

//Generating random numbers according to the
Normal (or Gaussian) random number distribution
Acts as source of noise voltage
*/
std::normal_distribution<> d(0,1);

//testing generation of white noise
//std::cout<<(d(gen))<<"\n";

//Designing the baseband RC filter
//RC filter params
double R_w=5e3; //series resistor
double C_w=1e-12; //shunted capacitor
double g_rw=1/R_w;//converting R_w to conductance for A\*M
//generating conductance parameter for C_w
double g_cw=(C_w)/dt;
//generating current source parameter for C_w
//generated using noisy voltage
double i_cw=-(C_w)/dt*(d(gen));
double V_whiteout; //defining output of RC filter

//define conductance matrix for RC filter
__float128 A_white[2][2]= {{g_rw,-g_rw},{-g_rw,(g_rw+g_cw)}};

//define voltage matrix for RC filter
__float128 B_white[2][1] = {{d(gen)},{V_whiteout}};
//define current matrix for RC filter
__float128 C_white[2][1] = {{0},{-i_cw}};

//taking inverse of the RC filter conductance matrix
__float128 det_white;
__float128 inv_A_white[2][2]= {{0,0},{0,0}};
det_white=A_white[0][0]*A_white[1][1]
    -A_white[0][1]*A_white[1][0];
inv_A_white[0][0]=A_white[1][1]/det_white;
inv_A_white[0][1]=-A_white[0][1]/det_white;
inv_A_white[1][0]=-A_white[1][0]/det_white;
inv_A_white[1][1]=A_white[0][0]/det_white;

/*multiplying the inverse matrix with the current matrix
to generate baseband white noise at each time step*/
B_white[0][0]=(inv_A_white[0][0]*C_white[0][0]) +
    (inv_A_white[0][1]*C_white[1][0]);
B_white[1][0]=(inv_A_white[1][0]*C_white[0][0]) +
    (inv_A_white[1][1]*C_white[1][0]);

//saving the baseband white noise to output node voltage
V_whiteout=B_white[1][0];

//printing scaled white noise to be added to current source
cout<<"t="<<t<<" Noise added= "<<(0.0012*V_whiteout)<<"\n";

//defining VCCS--current source
__float128 o; //temp parameter
//applying quadprecision tanh function
gm=(0.1)*tanhq(V1[t-39]);
//adding baseband white noise scaled for SNR
o=gm+(0.0012*V_whiteout);

//Updating the node current matrix at end of iteration
// updating current at node 1
C[0][0] = (-ic1+il),
// updating current at node 2
C[1][0] = (-ic2-il-o);
// updating current value of current to previous value for ADM
il_init=il;

//end of sample
int end_dft = 2e9;

//final updated parameters print

//final L current source value
cout<<"\n"<<il_new<<il<<"\n";
//final transconductance value
cout<<"\n"<<gm_is<<gm<<"\n";

//Writing voltage at node 2 to file
to check intermittency data
ofstream myfile;
myfile.open ("V2_withdelay_400period.txt");
for (int i = 0; i < size_org; i++)
{
    myfile << V2[i]<<"\n";
}
myfile.close();
*/

/*
Implementing Discrete Fourier transform
instead of usage of routine, the process is repeated
to avoid memory accesses (cache friendly)
DFT is implemented as a running summation
//original size of iterations
int size_dft=size_org;

//number of frequency points to be calculated
int size_dfty=2e5;

//used to calculate DFT in three slots
int increment=0;

//start of DFT window
int initial_size_dft=0;

//declaring parameters for DFT
__float128 pi2 = -2.0 * M_PI; // declaring 2*pi constant
__float128 angleTerm; // used to represent frequency point

// and to calculate real and imaginary parts
__float128 cosineA; // used to calculate real part of DFT
__float128 sineA; // used to calculate imaginary part of DFT

// as Parsing of data via loops shows error
__float128 temp_real; // used to swap real values
__float128 temp_imag; // used to swap imaginary values
int N_dft= 1e3; // used to divide the angle term

double y_dft_deeper=0; // used for scaling from FDID

// to real frequency axis
__float128 invs = 1.0 / N_dft;

// declare quad precise DFT vector
std::vector< std::complex<__float128> > output_seq(size_dft);

// Calculating DFT values for each frequency point
for( int y = initial_size_dft;y < size_dfty;y++)
{
    output_seq[y] = 0;
    /*
      scaling from FDID to real frequency axis
starting from 973 MHz in 10 Hz steps
*/
y_dft_deeper = 2.4316321 + (0.0000001 * y);

// for first slot multiples of 10Hz
if (y_dft_deeper < 2.4318321)
{
    // skip value for first slot
    int first_1 = 0;
    increment = first_1;
    // save real value to temp
    temp_real = std::real(output_seq[y]);
    // save imaginary value to temp
    temp_imag = std::imag(output_seq[y]);

    // first slot running summation for a freq point
    for (unsigned int x = 0; x < end_dft; x++)
    {
        /*
         * calculating angle of real and imaginary parts for a freq point
         */
        angleTerm = pi2 * y_dft_deeper * x * invs;
        // Real part cosine term
        cosineA = cosq(angleTerm);
        // Imaginary part sine term
        sineA = sinq(angleTerm);
        // real part of DFT as a running summation
        temp_real = temp_real + (V2[x] * cosineA);
        // imaginary part of DFT as a
        // running summation
        temp_imag = temp_imag + (V2[x] * sineA);
    }
    // saving the temp values to vector DFT


```cpp
output_seq[y]= (temp_real, temp_imag);
output_seq[y] *= invs;
// printing the iteration value
cout << "iteration = " << y;
// incrementing the frequency to next DFT point
y = y + increment;
}

else
{
    // save real value to temp
    temp_real = std::real(output_seq[y]);
    // save imaginary value to temp
    temp_imag = std::imag(output_seq[y]);

    // for second slot multiples of 400Hz
    if (y_dft_deeper < 2.4336321)
    {
        // skip value for second slot
        int first_2 = 9;
        increment = first_2;

        // second slot multiples of 400Hz
        for (unsigned int x = 0; x < end_dft; x++)
        {
            /*
              calculating angle of real and imaginary parts for a freq point
              */
            angleTerm = pi2 * y_dft_deeper * x * invs;
            // Real part cosine term
            cosineA = cosq(angleTerm);
            // Imaginary part sine term
            sineA = sinq(angleTerm);
            // real part of DFT
```
// as a running summation
temp_real = temp_real + (V2[x] * cosineA);
// imaginary part of DFT
// as a running summation
temp_imag = temp_imag + (V2[x] * sineA);

// saving the temp values to vector DFT
output_seq[y] = (temp_real, temp_imag);
output_seq[y] *= invs;
// printing the iteration value
cout << "iteration = " << y;
// incrementing the frequency to next DFT point
y = y + increment;

else {
  // save real value to temp
  temp_real = std::real(output_seq[y]);
  // save imaginary value to temp
  temp_imag = std::imag(output_seq[y]);
  // skip value for third slot
  int first_3 = 99;
increment = first_3;

  // third slot multiples of 4 KHz
  for (unsigned int x = 0; x < end_dft; x++) {
    /*
    calculating angle of real and imaginary parts for a freq point
    */
    angleTerm = pi2 * y_dft_deeper * x * invs;
    // Real part cosine term
    cosineA = cosq(angleTerm);
// Imaginary part sine term
sineA = sinq(angleTerm);

// real part of DFT
// as a running summation
temp_real = temp_real + (V2[x] * cosineA);

// imaginary part of DFT
// as a running summation
temp_imag = temp_imag + (V2[x] * sineA);

// saving the temp values to vector DFT
output_seq[y] = (temp_real, temp_imag);
output_seq[y] *= invs;

// printing the iteration value
cout << "iteration = " << y;

// incrementing the frequency to next DFT point
y = y + increment;

// Writing DFT to file
ofstream myfile_dft;
char buf[128];
myfile_dft.open("jan_white_noiseadded_DFTdeep.txt");

// saving the different frequency points with same skip values
for (int i = initial_size_dft; i < size_dft; i++)
{
    if (i < 2000) // first slot save
    {
        increment = 0; // skip value for first slot
        // save quad DFT value to buf
        // with same precision settings
quadmath_snprintf(buf, sizeof buf,"%+-#46.*Qe",128,output_seq[i]);
//setting precision to 128 and printing
myfile_dft << buf<<"\n"; //saving buf to file
i=i+increment; //incrementing to next DFT iteration
}
else
{
    if(i<20000) //second slot save
    {
        increment=9; //skip value for second slot
        //save quad DFT value to buffer with
        //same precision settings
        quadmath_snprintf(buf, sizeof buf,"%+-#46.*Qe",128,output_seq[i]);
        myfile_dft << buf<<"\n"; //saving buf to file
        //incrementing to next DFT iteration
        i=i+increment;
    }
    else //third slot save
    {
        increment=99; //skip value for third slot
        //save quad DFT value to buf
        //with same precision settings
        quadmath_snprintf(buf, sizeof buf,"%+-#46.*Qe",128,output_seq[i]);
        myfile_dft << buf<<"\n"; //saving buf to file
        //incrementing to next DFT iteration
        i=i+increment;
    }
}
myfile_dft.close();
return (0);
B.1 Data for figures

In this section the various data sets for different figures are listed as files.

<table>
<thead>
<tr>
<th>Figure</th>
<th>Filename</th>
<th>Software used</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.11</td>
<td>transient.txt</td>
<td>Matlab</td>
</tr>
<tr>
<td>4.3</td>
<td>intermittency_100ps.txt</td>
<td>Matlab</td>
</tr>
<tr>
<td>4.4</td>
<td>intermittency_0ps.txt</td>
<td>Matlab</td>
</tr>
<tr>
<td>4.2</td>
<td>whitenoise_autocorr.txt</td>
<td>C++/Matlab</td>
</tr>
<tr>
<td>4.5</td>
<td>transient_effect_pn.txt</td>
<td>C++</td>
</tr>
</tbody>
</table>
Table B.2 Data files-Part2.

<table>
<thead>
<tr>
<th>Figure</th>
<th>Filename</th>
<th>Software used</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.6</td>
<td>window_final_scales.txt</td>
<td>C++</td>
</tr>
<tr>
<td>4.7</td>
<td>PN_Q15_nowhite.txt</td>
<td>C++</td>
</tr>
<tr>
<td>4.8</td>
<td>double_precision_data2.txt</td>
<td>C++</td>
</tr>
<tr>
<td>5.1</td>
<td>Pn_final_org_Q400_nowhite.txt</td>
<td>C++</td>
</tr>
<tr>
<td>5.2</td>
<td>diff_slopes_diffdelay.txt</td>
<td>C++</td>
</tr>
<tr>
<td>5.3</td>
<td>intermittencydeep_morepoints_200ps.txt</td>
<td>Matlab</td>
</tr>
<tr>
<td>5.5</td>
<td>PN_Q15_nowhite.txt</td>
<td>C++</td>
</tr>
<tr>
<td>5.5</td>
<td>PN_Q17_8_nowhite.txt</td>
<td>C++</td>
</tr>
<tr>
<td>5.5</td>
<td>PN_Q20_nowhite.txt</td>
<td>C++</td>
</tr>
<tr>
<td>6.7</td>
<td>PN_Q3_7_white_SNR100.txt</td>
<td>C++</td>
</tr>
<tr>
<td>6.8</td>
<td>PN_Q17_8_white_SNR88.txt</td>
<td>C++</td>
</tr>
<tr>
<td>6.8</td>
<td>PN_Q17_8_nowhite.txt</td>
<td>C++</td>
</tr>
<tr>
<td>6.9</td>
<td>PN_Q17_8_white_SNR88.txt</td>
<td>C++</td>
</tr>
<tr>
<td>6.10</td>
<td>PN_Q15_white_SNR88.txt</td>
<td>C++</td>
</tr>
<tr>
<td>6.10</td>
<td>PN_Q17_8_white_SNR88.txt</td>
<td>C++</td>
</tr>
<tr>
<td>6.10</td>
<td>PN_Q20_white_SNR88.txt</td>
<td>C++</td>
</tr>
</tbody>
</table>